Propose HMNIDS Hybrid Multilevel Network Intrusion Detection System

Dr. Saad K. Majeed[1] Dr. Soukaena H. Hashem[1] Ikhlas K. Gbashi[1]

[1]Computer Sciences/ University of Technology/ Baghdad-Iraq

Abstract

This research present a proposal Hybrid Multilevel Network Intrusion Detection System (HMNIDS) which is a "hybrid multilevel IDS", is hybrid because use misuse and anomaly techniques in intrusion detection, and is multilevel since it apply the two detection techniques hierarchal in two levels. First level applies anomaly ID technique using Support Vector Machine (SVM) for detecting the traffics either normal or intrusions, if normal then passes it else the system input the intrusion traffic to the second level to detect the class of intrusion where this level apply Misuse ID technique using Artificial Neural Networks (ANN).

The proposal depend on Data mining is a DM-based HMNIDS since mining provide iterative process so if results are not satisfied with optimal solution, the mining steps will continue to be carried out until mining results are corresponding intention results. For training and testing of MHNIDS in our experiment, we used NSL-KDD data set. It has solved some of the inherent problems of the KDD'99. NSL-KDD similar to KDD99 their connections contains 41 features and is labeled as either normal or attack type, many of these features are irrelative in classification process. In our proposal we used Principle Component Analysis (PCA) as feature extraction to reduce no. of features to avoid time consuming in training and real-time detecting. PCA introduce 8 features as subset of correlated intrinsic features present the basic point in classification. The sets of features that have been resulted from PCA and the all features set will be the feeding of HMNIDS.

The results obtained from HMNIDS showing that accuracy rate of SVM and ANN classifiers separately are both high but they are higher with PCA (8) features than all (41) features. Confusion matrix of HMNIDS gives high detection rates and less false alarm rate, also they are higher with (8) PCA than all (41).

Keywords
NIDS, PCA, SVM, ANN and Confusion Matrix

1. Introduction

An intrusion can be defined as "any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource". Also intrusion is any set of deliberate, unauthorized inappropriate, and/or illegal activity by perpetrators either inside or outside a system, which can be deemed a system penetration, that attempt to compromise the integrity, confidentiality or availability of a resource [1]. An intrusion detection system is a system that can analyze in real time or delayed events from a computer system. It detects overflows, among other rights and prevents visible signs of attacks against information systems. It's sort of a device to monitor the activity of a machine or network to detect intrusion attempts and generate alerts for possible against reactions and procedures. The detection of intrusions is an important component of infrastructure protection mechanisms and it analyzes the occurring events in the aim to identify intrusive behavior and establish a response plan [2, 3].

Intrusion detection systems have been widely applied to prevent and reduce damage to information systems. The criteria of an intrusion system are described as follows. Robustness, because IDSs are susceptible to attack, survivability is the essential ability such as redundancy, health checking, mobility, and dynamic recovery. Efficiency, the IDS modules can execute the sniffing and analysis software in time to detect the anomaly behaviors. Adaptability, the intrusion

detection model does not depend on any special system, application environments, system vulnerabilities, or sorts of intrusions. Scalability, IDS can scale sizes for different traffics growing to avoid IDS services bottleneck. Feedback, the feedback measures the system whether to detect an intrusion and take certain actions automatically [4]. IDS can be classified according to IDS's environment as: a network-based IDS (NIDS) that is a dedicated computer, or special hardware platform, with detection software installed that captures packets in a promiscuous mode, or as a host-based IDS (HIDS) that monitors the resource usage of the operating system (OS) and the network. HIDS can only monitor the resource usage of the applications and not the applications themselves. Intrusion detection techniques are classified into two broad categories: misuse detection and anomaly detection. Misuse detection works by searching for the traces or patterns of well- known attacks. Clearly, only known attacks that leave characteristic traces can be detected that way. Anomaly detection, on the other hand, uses a model of normal user or system behavior and ages significant deviations from this model as potentially malicious. This model of normal user or system behavior is commonly known as the user or system profile. Strength of anomaly detection is its ability to detect previously unknown attacks [2, 3]. In order to overcome the limitations of traditional intrusion detection system, a systematic and higher automation method should be employed in the design of intrusion detection system. Data mining is the kind of effective method. Data mining concentrates on analyzing from a lot of noise, fuzzy and random data and extracts meaningful, potential useful information and knowledge. Mining is a repeated execution of the following three steps: Collect the data need for mining, Do the mining operation for the data and Get the corresponding result and then express the result with a certain way [5].

2. Related works

In [6] Ahmad I. et al, propose instead of using traditional approach of selecting features with the highest eigenvalues such as PCA, they applied a Genetic Algorithm (GA) to search the principal feature space for genetic eigenvectors that offers a subset of features with optimal sensitivity and the highest discriminatory power. Therefore, in this research, a mechanism for optimal features subset selection is proposed to overcome performance issues using PCA, GA and Multilayer Perceptron (MLP) is used for classification purpose. Consequently, this method provides optimal intrusion detection mechanism which is capable to minimize amount of features and maximize the detection rates. In [7] Reddy E. K. et al, they see network security technology has become crucial in protecting government and industry computing infrastructure. Modern intrusion detection applications facing complex problems. These applications has to be require reliable, extensible, easy to manage, and have low maintenance cost. In recent years, data mining-based intrusion detection systems (IDSs) have demonstrated high accuracy, good generalization to novel types of intrusion, and robust behavior in a changing environment. Still, significant challenges exist in the design and implementation of production quality IDSs. Instrumenting components such as of data transformations, model deployment, cooperative distributed detection and complex engineering endeavor. In [8] Suebsing A. et al, see in the previous researches on feature selection, the criteria and way about how to select the features in the raw data are mostly difficult to implement. Therefore, this work presents the easy and novel method, for feature selection, which can be used to separate correctly between normal and attack patterns of computer network connections. The goal in their work is to effectively apply Euclidean Distance for selecting a subset of robust features using smaller storage space and getting higher Intrusion detection performance. Experimental results show that the proposed approach based on the Euclidean Distance can improve the performance of a true positive intrusion detection rate especially for detecting known attack patterns. In [9] Bensefia H. et al., propose a new approach for IDS adaptability by integrating a Simple Connectionist Evolving System (SECOS) and a Winner-Takes-All (WTA) hierarchy of XCS (eXtended Classifier System). This integration puts in relief an adaptive hybrid intrusion detection core that plants the adaptability as an intrinsic and native functionality in the IDS. In [10] Vaarandi R., proposed a novel unsupervised DM based approach for IDS alert classification. With this strategy, knowledge is mined from IDS logs and processed in an automated way, in order to build an caution classifier. The classifier is then used in real-time for discerning important IDS warns from frequently occurring false positives and events of low significance. In [11] Vaarandi R. et al., extended their previous work (Risto, 2009) on IDS alert classification, and present a novel unsupervised real time alert classification method which is based on frequent itemset mining and data clustering techniques. Their method first applies a frequent itemset mining algorithm to past IDS alert logs, in order to discover patterns that describe redundant alerts. After that, data clustering methods are used for finding detailed subpatterns for each detected pattern. Finally, the detected knowledge is explained and used

for real time classification of IDS alerts, in order to characterize critical alerts from irrelevant ones. In [12] Mohammad M. N. et al., introduced an improved approach for IDS based on combining DM and expert system that is presented and implemented in WEKA (Waikato Environment for Knowledge Analysis). They aimed to design and develop intelligent DM IDS and its core part a composite detection engine with anomaly detection and misuse detection features and the two detection engines work serially to detect the user's activity in turn. The system collects the data of DB audit system in real time, analyzes the audit data, judges that it is a normal behavior, abnormal behavior or aggressive behavior and responds to the result obtained by the operation behavior and finally reports the result to the manager in a comprehensible form. In [13] Guojun Z. et al., presented a cooperative IDS based on IPv6 to address this challenge. Such a system consists of four parts: data flow tracking and analysis, capturing packets and rules matching, disaster recovery, and blocking. The technique of cooperative ID is introduced into the system for realizing the coordination control among parts. The system has a perfect detection rating. In [14] Al-Janabi S. T. and Saeed H. A. proposed an anomaly based IDS that can promptly detect and classify various attacks. Anomaly-based IDSs need to be able to learn the unstable behavior of users or systems. The proposed IDS experimenting with packet behavior as parameters in anomaly ID. There are several methods to assist IDSs to learn system's behavior, the proposed IDS uses a back propagation artificial neural network (ANN) to learn system's behavior and uses the KDD CUP'99 dataset in its experiments and the obtained results satisfy the work objective. In [15] Haldar N. et al presented IDS which employs usage of classification methods to model the usage patterns of authenticated users and uses it to detect intrusions in wireless networks. The key idea behind the proposed IDS is the identification of discriminative features from user's activity data and using them to identify intrusions in wireless networks. The detection module uses statistical methods to accumulate interested statistical variables and compares them with the thresholds derived from users activities data. When the variables exceed the predestined thresholds, an alarm is put forward to alert about a sensible intrusion in the network.

3. Analysis for Critical Points in Current NIDS

From our survey of traditional and current NIDS there are number of significant drawbacks, we see these critical issues must be taken in consideration to construct our proposed model, these drawbacks are:

Traditional dataset KDDCUP'99 is the mostly widely used data set for the intrusion detection. But this dataset has huge no. of records and high no. of redundant records. That make selecting subsets of records for training and testing fatiguing for researchers and affect on performance of IDSs.

NIDS are usually tuned to detect known service level network attacks. This leaves them vulnerable to original and novel malicious attacks.

That amount of data needs to look at seems to be growing rapidly. Depending on the intrusion detection tools employed by a company and its size there is the possibility for logs to reach millions of records per day.

False positives, a common complaint is the amount of false positives an IDS will generate. A false positive occurs when normal attack is mistakenly classified as malicious and treated accordingly.

False negatives, this is the case where IDS does not generate an alert when an intrusion is actually taking place.

In anomaly detection the subject's normal behavior is modeled on the basis of the (audit) data collected over a period of normal operation. If undiscovered intrusive activities occur during this period, they will be considered normal activities. Because of some technical reasons, the current anomaly detection approaches usually suffer from a high false-alarm rate.

Another difficult problem in building anomaly detection models is how to decide the features to be used as the input of the models. It is not guaranteed that all and only the features related to intrusion detection will be selected as input parameters. Although missing important intrusion-related features makes it difficult to distinguish attacks from normal activities, having non-intrusion-related features could introduce "noise" into the models and thus affect the detection performance.

Current misuse detection systems usually work better than anomaly detection systems for known attacks. That is, misuse detection systems detect patterns of known attacks more accurately and generate much fewer false alarms. This better performance occurs because misuse detection systems take advantage of explicit knowledge of the attacks.

Limitation of misuse detection is that it cannot detect novel or unknown attacks. As a result, the computer systems protected solely by misuse detection systems face the risk of being comprised without detecting the attacks.

4. The Proposed Model of HMNIDS

The proposed HMNIDS, see figure (1), is a "hybrid multilevel IDS", is hybrid because it trend to detect intrusions with both techniques misuse and anomaly, and is multilevel because it trend to detect intrusions

with two levels for more accurate intrusions types detections. First level of proposed HMNIDS will apply anomaly ID technique should first learn the characteristics of normal activities and abnormal activities of the system, and then the HMNIDS detect traffic that deviate from normal activities. So, the result of first level is detecting the traffics either normal or intrusions, if normal then passes it else enter the intrusion traffic to the second level to detect the class of intrusion. Second level of proposed HMNIDS will apply Misuse ID technique is able to automatically retrain ID models on different input data that include new types of attacks, as long as they have been labeled appropriately. The results of this level are detecting the type of intrusions.

The proposal is a DM-based HMNIDS in which both the misuse and anomaly detection techniques depended in the detection of intrusion, where each instance in a dataset is labeled as "normal" or "intrusion (specify intrusion type)" and a learning algorithms is trained over the labeled data. If the mining results are not satisfied with what we want to, the mining steps will continue to be carried out until mining results are corresponding with our intention results. For training and testing of the proposed MHNIDS in our experiment, we used NSL-KDD data set. It has solved some of the inherent problems of the KDD'99. It is considered as standard benchmark for intrusion detection evaluation. The training dataset of NSL-KDD similar to KDD99 consist of approximately 4,900,000 single connection vectors each of which contains 41 features and is labeled as either normal or attack type ,with exactly one specific attack type. We see the demand for the number of samples for the training the classifier grows exponentially with the dimension of the feature space. This limitation is called the 'curse of dimensionality'. In our proposed HMNIDS model we indicate that feature reduction technique is capable of reducing the size of dataset. The time and space complexities of most classifiers used are exponential function of their input vector size.

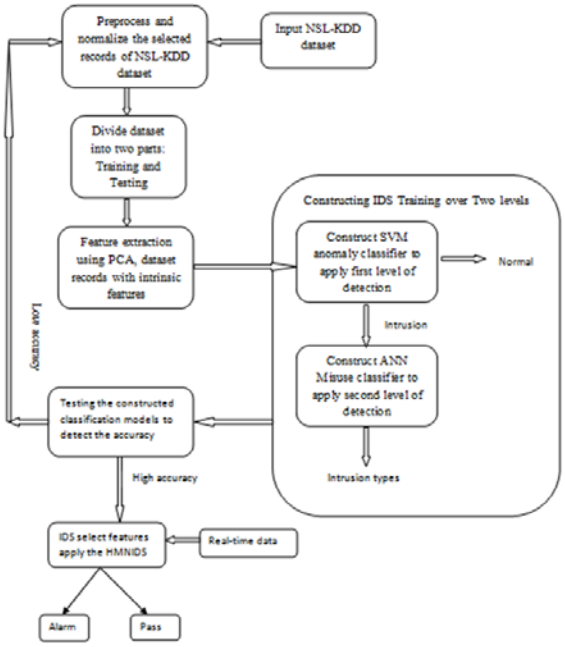

Fig.1. Detailed Proposed HMNIDS

4.1 Description of Dataset

KDD'99 training dataset consists of approximately 5 million connection records (a connection is a sequence of TCP packets starting and ending at some well-defined times, between which data flows to and from a source IP address to a destination IP address under some well-defined protocol) each of which contains 41 features and is labeled as either normal or an attack, with exactly one specific attack type. These features are grouped into four categories: basic features, content features, timebased traffic features and host-based traffic features. The simulated attacks fall in one of the following four categories: Denial of Service Attack (DoS), User to Root Attack (U2R), Remote to Local Attack (R2L), and Probing Attack. A total of 22 training known attack types and additional 17 unknown types are summarized.

The KDD dataset consists of three components: "Whole KDD", "Corrected KDD" and "10% KDD", KDDCUP'99 is the mostly widely used data set for the intrusion detection. But researchers conducted a statistical Analysis on this data set and found important issues which highly affect the performance of evaluated systems, and results in a very poor evaluation of anomaly detection approaches. To solve these issues, they have proposed a new data set, NSL-KDD, which consists of selected records of the complete KDD data set. NSL-KDD has many advantages over the original KDD dataset; It does not include redundant records in the train set, so the classifiers will not be biased

towards more frequent records. The number of selected records from each difficulty level group is inversely proportional to the percentage of records in the original KDD data set. As a result, the classification rates of distinct machine learning methods vary in a wider range, which makes it more efficient to have an accurate evaluation of different learning techniques. The numbers of records in the train and test sets are reasonable, which makes it affordable to run the experiments on the complete set without the need to randomly select a small portion. Consequently, evaluation results of different research works will be consistent and comparable.

4.2. Preprocessing on NSL-KDD Dataset

The following processes have been applied to the "NSL-KDD dataset" before it being used in design of the proposed system:

Converting the original NSL-KDD dataset from a text file to SQL server.

Since type of some of NSL-KDD dataset's features is continuous, thus a process for normalization these features have been done in order to become of categorical type so it becomes more convenient with the used DM classification algorithms. Normalization is used for data preprocessing, where the attribute data are scaled so as to fall within a small specified range such as -1.0 to 1.0 or 0.0 to 1.0. Since using neural network back propagation algorithm for classification, normalizing the input values for each attribute measured in the training samples will help speed up the learning phase.

The resulted dataset from process two will be split into two distinct datasets by using, one for classifiers' training which nearly equal 2⁄3 of resulted dataset from process two and the other for classifiers testing which nearly equal 1⁄3 of the same resulted dataset.

4.2 Process of Feature Reduction

Feature selection is intended to suggest which features are more important for the prediction, to find out and get rid of irrelevant features that reduce classification accuracy, discover relations between features and throw out highly correlated features which are redundant for prediction. An earlier general task in data mining is to extract outstanding features for the prediction. This function can be broken into two groups; feature extraction or feature transformation, and feature selection. Feature extraction refers to the process of creating a new set of combined features (which are combinations of the original features). On the other hand, feature selection is different from feature extraction because it does not produce new variables. Feature selection also known as variable selection, feature reduction, attribute selection, feature ranking, or variable subset selection, is a widely used dimensionality reduction technique, which has been the focus of much research in machine learning and data mining.

In our proposal we used Principle Component Analysis (PCA) as feature extraction rather than using any techniques of feature selection. That because we have 41 features, like this no. of features will cause time consuming in training and real-time detecting, so we need to transform these set of 41 features into small subset of correlated intrinsic features present the basic point in classification. Principle Component Analysis (PCA) is a useful statistical technique that has found application in fields such as face recognition and image compression, and is a common technique for finding patterns in data of high dimension. The complete subject of PCA statistics is based on the idea that you have this huge set of data, and you want to analyze that set expressions of the relationships between the single points in that set. PCA is applied to the training dataset to find the intrinsic features, see algorithm (1), the resulted set of features in addition to the original set present all features in the training dataset will be used in design (learning) of the classifiers.

Algorithm (1): Suggested-PCA

Input: NSL-KDD training dataset.

Output: PCA set of most frequent and related features.

Steps:

Obtain training NSL-KDD'99 transactions.

Represent every transaction Ii as a vector xi.

Compute the average transaction $\Psi=1/M \sum_{i=1}^{M} xi$ ……..…...(1)

Subtract the mean transaction $\varphi i=xi-\Psi i$……..…. (2)

Compute the covariance matrix $C=1/M \; [\varphi_n \varphi] \_n^T=AAT$……..……(3)

From C Compute eigenvectors ui of AAT:

Consider matrix AAT as a M×M matrix.

Compute the eigenvectors vi of AAT such that:

ATAvi →μ iVi → AATAVi = μ iAvi → Cui = μ iui where μ i =Avi……..(4)

Compute the μ best eigenvectors of AAT: μi= Avi ……..…(5)

Keep only K eigenvectors, (K features with their values).

End.

4.3. Suggested Hybrid Multilevel NIDS

After the intrinsic features had been selected, the two popular DM classification algorithms: Support Vector Machine SVM from statistical field and Artificial Neural Networks from soft computing field, these two

classifiers will used in the design of the suggested HMNIDS in shape of two levels, So both of SVM and ANN classifiers will be trained 2 times, one training with dataset of all 41 features and other with dataset of PCA features to design the suggested HMNIDS classifier, these levels are:

First Level (SVM-Anomaly Detection), Support vector machines (SVMs) are a set of related supervised learning methods used for classification and regression. They belong to a family of generalized linear classifiers. SVMs attempt to separate data into multiple classes (two in the basic case) though the use of a hyper-plane. Here will use in more conventional SVM approach, we used one SVM, as anomaly detection techniques to identify normal traffic from intrusion traffics. Algorithm (2) explains SVM algorithm with Anomaly Intrusion detection learned on NSL-KDD dataset.

Algorithm (2): Suggested-SVM

Input: NSL-KDD for training and testing.

Output: Results of Anomaly detection on NSL-KDD testing dataset using SVM.

Steps:

1.Initialize all points in training dataset as (Xi, Yj) where X is a vector of data x1, ……, xn and Y is vector of classes.

2.Initialize vector of weight W.

3.Distribute all points (x, y) and extract the hyper plane separator.

4.If the hyper plane give optimal separation then depend hyper plane as classifier model to classify testing dataset and go End

5.Else must do the following steps

6.Maximize the hyper plan using equation of Getting Maximum Margin = MM = 2 / ||w||. ……. (6)

7.For minimum using equation same as maximizing $\Phi(w) = \frac{1}{2} w^T w$ ……. (7)

8.Initialize Lagrange multiplier αi vector α1…αn using equation

Q(α) =Σαi - ½ΣΣαiαjyiyjxiTxj ……. (8)

9. Apply classification function using equation f(x) = ΣαiyixiTx + b ……. (9)

10. Determine the support vectors xi with non-zero αi (support vectors are the points determine the area of hyper plan)

11. Depend the hyper plan resulted after determining support vectors as the classifier model to classify testing dataset

12. End

Second Level (ANN-Misuse Detection), an artificial neural network is a system simulating work of the neurons in the human brain. The neuron consists of some inputs emulating dendrites of the biological neuron, a summation module, an activation function and one output emulating an axon of the biological neuron. The importance of a particular input can be intensified by the weights that simulate biological neuron's synapses. Then, the input signals are multiplied by the values of weights and next the results are added in the summation block. The sum is sent to the activation block where it is processed by the activation function. Thus, we obtain neuron's answer to the input signals "x". Here will use in more conventional MLP (Multi Layer Perceptron) approach as misuse detection techniques to identify class of intrusion traffics (these are detected intrusion in the first level classifier and sent to the second level to determine it is class). Algorithm (3) explains ANN algorithm with Misuse Intrusion detection learned on NSL-KDD dataset.

Algorithm (3): Suggested-ANN

Input: NSL-KDD for training and testing.

Output: Results of Misuse detection on NSL-KDD testing dataset using ANN.

Steps:

Main Assumption for the Training Process of MLP:

Learning method: Quasi Newton BFGS and Levenberg-Marquardt

Number of Epochs: 1000.

MSE (Mean Square error): 0.01.

Learning rate: 0,9.

Activation function: log-signoid.

Number of neurons in the Input layer: (41 or according no. of PCA set).

Number of neurons in the hidden layer: (21 for 41 input neurons and with no. of PCA set equal to half of this no.).

Number of neurons in the output layer: (4 cause no. of intrusions classes are 4).

Update of weights – batch mode (after presentation of the entire training data set).

Train and Test on NSL-KDD to construct final ANN misuse NIDS.

End.

5. Discussion and Experimental works

The proposal had been implemented on the following platform: Windows 7 Ultimate Service Pack1 and 32-bit OS, 16GB RAM, and Intel® Core (TM) 3 Duo CPU with 2.00 GHz; and by using Visual Basic.Net and SQL server.

Training results of the proposed MHNIDS will presented with two stages; first one will introduce training results of anomaly classifier (SVM) and misuse classifier (ANN) separately. Second level will introduce training results of the complete proposed HMNIDS. That partitioning in presenting the results of training to discover the strength and weak points in the proposed system in all it is parts. Before explaining the

results will introduce the no. of records taken from NSL-KDD as samples for training and testing, see table (1), then present the set of intrinsic features obtained by applying PCA on training dataset, this subset is {Protocol_type, Service, Flag, count, srv_count, same_srv_rate, dst_host_srv_count, dst_host_same_srv_rate, dst_host_same_src_port_rate}.

No. of Records/Type of Records	Training	Testing
DOS	Normal	23,000
Probe	500	1,000
R2L	150	2,000
U2R	100	500
Normal	10,000	7,000

Table (1): No. of Records selected from NSL-KDD for

First Training (SVM and ANN separately)

In this study, as in table (1) Training data set in the paper contained 50,750 records, which were randomly generated from the NSL-KDD for training data set that consists of 10,000 normal patterns, 40, 000 known DoS patterns, 500 known Probe patterns, 150 known R2L patterns and 100 known U2R patterns. Testing data set in the paper contained 33,500 records, which were randomly generated (with omitting the records of training) from the NSL-KDD for testing data set that consists of 7,000 normal patterns, 23, 000 known and unknown DoS patterns, 1,000 known and unknown Probe patterns, 2000 known and unknown R2L patterns and 500 known and unknown U2R patterns.

Training which consist of two hierarchy classifiers (SVM and ANN) on Training dataset has been done with two sets of features (All_Features, PCA_Features), so the proposed system has been experimented (i.e., trained and tested) for two times to assess the accuracy of the classifiers. We performed three different experiments and selected a subset of eight features that indicates better performance as compared to others. Our aim is to select minimum features that produce optimal results in accuracy. This definitely impact on overall performance of the system. The features are reduced to 8 from the 41 raw features set. The above experiments show that optimal features increased accuracy, reduced training and computational overheads and simplified the architecture of intrusion analysis engine.

Results of three conducted experiments (Exp1, Exp2, Exp3), which producing the most accurate results, have been presented in this section. Four classification models have been constructed in each of these three experiments. Next these models have been applied on the same Testing dataset, which has been constructed during Exp1, to assess the validation and accuracy of these constructed models on the same testing dataset. The classification results of testing are either TP (intrusion), TN (normal), false positive (FP) (misclassified as intrusion), false negative (FN) (misclassified as normal), Unknown (new user behavior or new attack). From classification results we calculate the detection rate (DR) of IDS is the ratio between the number of TP and the total number of intrusion patterns presented in the testing dataset. It has been calculated using

$$DR = TP/(TP+FN+Unknown2) *10 \quad (10)$$

,and the false alarm rate (FAR) of an IDS is the ratio between number of "normal" patterns classified as attacks (FP) and the total number of "normal" patterns presented in the testing dataset. It has been computed using

$$FAR=FP/(TN+FP+Unknown1)*100\% \quad (11)$$

Values for both of DR and FAR for each classifier in the three experiments have been illustrated in table (2).

Table (2): DRs and FARs of both of them SVM and ANN

SVM-Anomaly Classifier				ANN-Misuse Classifier			
Feature Selection Measure	Experiment No.	DR	FAR	Feature Selection Measure	Experiment No.	DR	FAR
PCA	1	1	0	PCA	1	0.999	0
	2	0.998	0		2	0.999	0
	3	0.999	0		3	0.999	0
ALL	1	0.995	0.02	ALL	1	0.994	0.03
	2	0.996	0.03		2	0.988	0.03
	3	0.995	0.07		3	0.995	0.03

DR are higher with SVM anomaly classifiers and also with ANN misuse classifiers and FAR often ranging between (0 - 0.07) with SVM classifiers and ANN classifiers. It is very clear from these that SVM classifiers are better with anomaly detection and ANN classifiers are better with misuse detection. Selection of the best classification model would be done significantly according to its classification accuracy, which is introduced as the ratio between the number of the correctly classified patterns (TP, TN) and the total number of patterns of the testing dataset. The accuracy (Accu) of each classifier has been calculated using

Accu = (TP+TN)/(TP+FP+TN+FN+unknown) *100% ……. (12)

Table (3) summarizes Accu of both SVM and ANN classifiers with PCA_F and ALL_F in the three experiments. According to these results, the classifiers SVM and ANN were more accurate with PCA_F Accu.

Table (3): Accuracy of SVM and ANN Classifiers

Classifier	Experiment No.	PCA_F	ALL_F
SVM-Anomaly	1	1	0.995
	2	1	0.996
	3	0.999	0.997
ANN-Misuse	1	1	0.994
	2	0.999	0.997
	3	0.999	0.995

Second training (Full HMNIDS)

Here will explain the overall accuracy of the proposed HMNIDS according confusion matrix calculations, we will display the confusion matrix for two cases; all 41 features and 8 features, see table (4) and table (5). We see the confusion of classification with 8 features less than confusion of classification with 41 features.

Table (4): confusion matrix of HMNIDS with 41 features

Confusion Matrix	DOS	Probe	R2L	U2R	Normal
DOS	22,981	3	6	0	10
Probe	10	985	1	0	5
R2L	6	1	1984	0	9
U2R	1	0	0	448	1
Normal	6	9	3	0	6992

Table (5): confusion matrix of HMNIDS with 8 features

Confusion Matrix	DOS	Probe	R2L	U2R	Normal
DOS	22,990	2	5	0	3
Probe	8	988	1	0	3
R2L	5	1	1989	0	5
U2R	1	0	50	448	1
Normal	1	3	1	0	6995

6. Conclusion

Data mining are introduced for helping IDS to detect intrusions correctly, and accordingly IDSs have shown to be successful in detecting known attacks. Feature selection is an important task of Network Intrusion application. Using PCA feature selection approach, intrusions are detected with less error rate and high accuracy. Usage of ANN for misuse intrusion detection and SVM for anomaly detection with the input data from the NSL-KDD gives good performance of HMNIDS as comparison with depended related works [6-15]. Tables (2 and 3) present results of detection for SVM and ANN separately where notice the higher rates of detection and very less rates of false alarms especially with PCA set of features. The same in tables (4 and 5) present results of detection for HMNIDS where notice from confusion matrices the higher rates of detection and very less rates of false alarms especially with PCA set of features.

References

1.Lalli and Palanisamy, “Modernized Intrusion Detection Using Enhanced Apriori Algorithm”, International Journal of Wireless & Mobile Networks (IJWMN) Vol. 5, No. 2, April 2013, [IVSL].

2.Saidi Ben Boubaker Ourida , “Implementation of an Intrusion Detection System”, International Journal of Computer Science Issues(IJCSI) ,Vol. 9, Issue 3,No. 1, May 2012, ISSN (Online): 1694-0814, www.IJCSI.org.

3.Garuba M., Liu C., Fraites D., "Intrusion Techniques: Comparative Study of Network Intrusion Detection Systems", IEEE Computer Society, Fifth International Conference on Information Technology, pp. 592-598, 2008.

4.Huang L. and Hwang M. “Study of Intrusion Detection Systems”, Journal Of Electronic Science And Technology, Vol. 10, No. 3, September 2012.

5.Zhou Q. and Zhao Y. , “ The Design and Implementation of Intrusion Detection System based on Data Mining Technology”, Journal of Applied Sciences, Engineering and Technology 5(14): 3824-3829, 2013 ISSN: 2040-7459; e-ISSN: 2040-7467 © Maxwell Scientific Organization.

6.Ahmad I., Abdulah A. B, Alghamdi A. S, Alnfajan K. and Hussain M., “Feature Subset Selection for Network Intrusion Detection Mechanism Using Genetic Eigen Vectors”, 2011 International Conference on Telecommunication Technology and Applications, Proc .of CSIT vol.5 (2011) © (2011) IACSIT Press, Singapore.

7.Reddy E. K. , Reddy V. N., Rajulu P. G., “A Study of Intrusion Detection in Data Mining”, Proceedings of the World Congress on Engineering 2011 Vol III WCE 2011, July 6 - 8, 2011, London, U.K.

8.Suebsing A., Hiransakolwong N. , "Euclidean-based Feature Selection for Network Intrusion Detection", 2009 International Conference on Machine Learning and Computing IPCSIT vol.3 (2011) © (2011) IACSIT Press, Singapore.

9.Bensefia H. and Ghoualmi N., "A New Approach for Adaptive Intrusion Detection", 2011 Seventh International Conference on Computational Intelligence and Security, 2011.

10.Vaarandi R., "Real-Time Classification of IDS Alerts with Data Mining Techniques", MILCOM'09 Proceedings of the 28th IEEE conference on Military communications pp.1786-1792.

11.Vaarandi R., and Podinš K., "Network IDS Alert Classification with Frequent Itemset Mining and Data Clustering", The 2010 International IEEE Conference on Network and Service Management, pp. 451-456.

12.Mohammad M. N., Sulaiman N. and Muhsin O. A., "A Novel Intrusion Detection System by using Intelligent Data Mining in Weka Environment", Published by Elsevier Ltd. Procedia Computer Science 3, pp. 1237–1242.

13.Guojun Z., Liping C. and Weitao H., "The Design of Cooperative Intrusion Detection System", IEEE Computer Society, 2011 Seventh International Conference on Computational Intelligence and Security, pp. 764-766.

14.Al-Janabi S. T., and Saeed H. A., "A Neural Network Based Anomaly Intrusion Detection System", IEEE Computer Society, 2011 Developments in E-systems Engineering, pp. 221-226.

15.Haldar N. A., Abulaish M. and Pasha S. A., "An Activity Pattern Based Wireless Intrusion Detection System", IEEE Computer Society, 2012 Ninth International Conference on Information Technology-New Generations, pp. 846-847.