Computer Engineering and Applications Vol. 4, No. 2, June 2015

Network Attacks Detection by Hierarchical Neural Network

Mohammad Masoud Javidi, Mohammad Hassan Nattaj

Department of Computer Science, Shahid Bahonar University of Kerman
javidi@uk.ac.ir

ABSTRACT

Intrusion detection is an emerging area of research in the computer security and net-works with the growing usage of internet in everyday life. Most intrusion detection systems (IDSs) mostly use a single classifier algorithm to classify the network traffic data as normal behavior or anomalous. However, these single classifier systems fail to provide the best possible attack detection rate with low false alarm rate. In this paper, we propose to use a hybrid intelligent approach using a combination of classifiers in order to make the decision intelligently, so that the overall performance of the resul-tant model is enhanced. The general procedure in this is to follow the supervised or un-supervised data filtering with classifier or cluster first on the whole training dataset and then the output are applied to another classifier to classify the data. In this research, we applied Neural Network with Supervised and Unsupervised Learning in order to implement the intrusion detection system. Moreover, in this project, we used the method of Parallelization with real time application of the system processors to detect the systems intrusions. Using this method enhanced the speed of the intrusion detection. In order to train and test the neural network, NSLKDD database was used. Creating some different intrusion detection systems, each of which considered as a single agent, we precisely proceeded with the signature-based intrusion detection of the network.In the proposed design, the attacks have been classified into 4 groups and each group is detected by an Agent equipped with intrusion detection system (IDS).

Keywords: Intrusion Detection, Multi-layer Perceptron, False Positives, Signature-based intrusion detection, Decision tree, Nave Bayes Classifier

1. INTRODUCTION

With the development of the internet and its wide application in all domains of ev-erybodys life, intrusion detection is becoming a critical process in computer network security. Intrusion detection systems (IDS) attempts to recognize and notify the users activity as either normal or anomaly (or intrusion) by comparing the network con-nection records to the known intrusion patterns obtained from the human experts. As traditional methods cannot detect the unknown intrusion patterns efficiently because of the problems faced by a human analyst during analyzing a complex and faster net-work, we concentrate on data mining based intelligent decision technology to make effective decisions to this effect.

Vasiliadis et al. [1] have also presented their idea about Parallelization of IDSs, in which they have applied all system CPUs to speed up the detection process of all intrusions of the network.

Detection of anomaly outside-in traffics of the network and reporting it to the network administrator, or preventing suspicious contacts is the other feature of IDS [2]. IDS tool is capable of detecting attacks caused by both internal and external users.

Wang et al [3] have declared their ideas about intrusion detection of the neural network and have explored different methods of the Neural Network. There are many different intelligent techniques for designing Intrusion Detection systems, such as Ma-chine Learning, Data Mininsg, and Fuzzy sets which are divided into two groups of Fuzzy misuse detection and Fuzzy anomaly detection, to mention some. Neural net-work algorithms are also divided into two groups of Supervised Learning and Unsu-pervised Learning.

In this paper, in order to detect the network intrusions, we have experienced neural network using Supervised Learning in a way that it uses the intrusion detection system CPU as parallel. In this project, NSLKDD database has been used in order to train and test the neural network. Creating some IDS layers, each of which is considered as a single agent, we start to detect the network intrusionsprecisely which belong to one of the four categories of DOS, Probing, U2R or R2L. In our proposed design, we have classified the attacks into 4 classes[4] and detect each class by using one agent equipped with IDS. These agents act independently and report the intrusion or non-intrusion in the system; the agents achievements will be studied in the Final Analyst and at last the analyst will report whether there has been an intrusion in the system or not.

In section 2, we have presented the related works which have previously done on Intrusion Detection in the Network. In section 3, the proposed implementation of the intrusion detection system, and also the database used in the project have been presented. In section 4, we have briefly discussed about the results evaluation and also the evaluation methods. In section 6, the evaluation results of the proposed method along with the works which have done in this domain have been presented. In section 7, we have presented the intrusion detection system as parallel. And finally, in section 8, you will be provided with the general conclusion of the article.

2. RELATED WORKS

Most of unsupervised systems have applied SOM networks and only few of them have used other methods. SOM, also known as Kohonen, is among the feedforward single layer networks, the output of which has been clustered in low dimensional space (2-dimensional or 3-dimensional) [5].

IDSs are still experiencing difficulties in detecting intrusive activity on their networks since novel attacks are consistently being encountered.

In [6] author shows that the accuracy and performance of an IDS can be improved through obtaining good training parameters and selecting right feature to design any Artificial Neural Network (ANN).

Zhang et al. in [7] applied Support Vector Machine for implementation of the proposed intrusion detection system, and also in [8] K-means was used for separating network intrusions from one another.

In [9], the author used PCA to project features space to principal feature space and select features corresponding to the highest Eigen values using Genetic

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

Algorithm. In [10], the author proposes an automatic feature selection procedure based on Correla-tion based Feature Selection (CFS).

In [11] Ben Amor tried to compare the two methods of Decision Tree and Nave Bayes which was shown in this article; the result: Nave Bayes method has a higher intrusion detection percentage than Decision Tree method, however the error rate per-centage of this method is also higher than the Decision Tree method.

In [12] author investigate the performance of two feature selection algorithm in-volving Bayesian network (BN) and Classification & Regression Tee (CART), and an ensemble of BN and CART and finally propose a hybrid architecture for combining different feature selection algorithms for intrusion detection.

Also, neural networks either have multiple output neurons [13] or get multiple binary neural network classifiers together [14]. Usually, when faced with a new intru-sion, the latter is more flexible than the former for the network and detect the intrusions more precisely [15]. Beghdad in [16] set to design an intrusion detection system with MLP which had applied KDD99 database and learning algorithm in the article had been chosen as a Resilient Back Propagation.

In [17], the author proposes two phase approach in intrusion detection design. In the first phase, develop a correlation-based feature selection algorithm to remove the worthless information from the original high dimensional database. Next phase de-signs an intrusion detection method to solve the problems of uncertainty caused by limited and ambiguous information.

Due to the capabilities of neural networks, they are able to act with deficient and incomplete data. Furthermore, there are Machine Learning techniques that can learn the templates they have not learnt during the training phase. Most ML algorithms have been proposed to recognize attacks; regardless of the minimum cost of error. These errors can lead to false alarms. The cost of a false alarm is more expensive than non-detection [18].

Carrying out different experiments on network packets, Rhodes et al. [19] declared that every network protocol has a single structure and function, so malicious activities in a specific protocol should be unique. It is somehow unrealistic to create a single SOM to undertake all these activities. Sarasamma et al. [20] achieved similar results, that is, different subsets of traits are suitable for attack detection.

3. IMPLEMENTATION

As already mentioned, the network attacks can be divided into4 groups of DOS, R2L, U2R, and Probe. In our designed IDS, the proposed intrusion detection system is able to well detect all kinds of attacks belonging to the 4 mentioned groups with high percentages. In fact, this IDS detects the 4 kinds of attacks separately and receives the responses of the 4 Agents by a single analyst and then returns one final response.

In order to design the proposed IDS, we first detected the attacks of each group and then designed a separate IDS for each specific attack. In general, regarding the designed IDSs, if there is a connection with each one of the 4 mentioned groups of attacks, the designed system will detect it, otherwise the connection is considered as a normal connection.

In order to train a neural network and have a more qualified process of neural network, we made some changes in the database which do not effect on the totality of the database;they are just done for improving the function of neural networks.

3.1 NSLKDD DATABASE

The utilized database consists of information about standard connection records which in their own turn includes a set of simulated attacks and intrusions in a military net-work.

A connection is a series of packets with TCP, UDP or ICMP protocols which start and finish at specific times and runs under a defined contract between those data from the origin IP address to the destination IP address and vice versa. Each connection is labeled as normal or an attack, and in the case that it is an attack, its type is exactly defined. The record of each connection consists of about one hundred bytes.

The attacks shown in this data set fall in four principle groups of DOS, R2L, U2R, and Probe, which are shown in Table 1 [21]. As you see in the table, approximately 74 percent of the whole database is composed of DOS attacks which indicate the importance of creating an IDS to detect this group of attacks.

TABLE 1.
Classification and Dispersion Percentage of attacks in database

Connection type	KDD Testing Set
Normal	19.48%
DOS	73.90%
U2R	1.34%
R2L	5.2%
Probe	0.07%

Each record from NSLKDD dataset, describes one connection in the form of 41 features. To implement the neural network algorithm, we applied the MATLAB soft-ware. In order to implement this algorithm, we should firstly train the designed neural network using training data, and then we should analyze the efficiency of the network using the experimental data.

To do so, The NSLKDD database which is the optimal version of KDD99, is itself divided into two parts of the Test and Train. We also used this verydatabase.Before using this database, it is necessary to mention that this database has a high capacity and we used only 10% of the records for testing and training the designed network. Of course, this 10% has been chosen in such a way to contain different types of attacks and include different modes of the network [21].

3.1.1. FEATURES REDUCTION

Some attacks are easily recognizable due to a set of specific features and there is no need to use the whole database for detection of this type of attacks.Based on this, researchers attempt to specify appropriate features for each category. The advantage of our proposed method compared to other methods is that, in this method there is an intrusion detection system for each category and based on this we could select the features which were necessary for each category separately; this process increases the intrusion detection accuracy.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

Sarsamma et al. in [22] recognized different subsets of features as suitable for detecting some specific type of attacks.Kayacik et al. in [20] studied a series of subse-quent experiments on KDD99 data.They found out that 6 basic features were enough for detecting an expanded range of DOS attacks while 41 features were necessary for minimizing the FP rate. Among these 6 main features, Protocol and Service are the most significant ones and the other features according to table 2 are: C, D, E, F.

In article [23] some reduction algorithms are implemented and compared to each other; we also used these very algorithms to improve the accomplishment of the pro-posed intrusion detection system. In this article, we applied 3 methods of Bayesian, Classification & Regression Trees (CART), and the combination of these 2 methods.

According to table 2, features of 17, 12, and 19 factors which have been used in this paper are as follows:

12 features achieved from feature reduction are: C, E, F, L, W, X, Y, AB, AE, AF, AG, AI.

17 features are: A, B, C, E, G, H, K, L, N, Q, V, W, X, Y, Z, AD, AF.

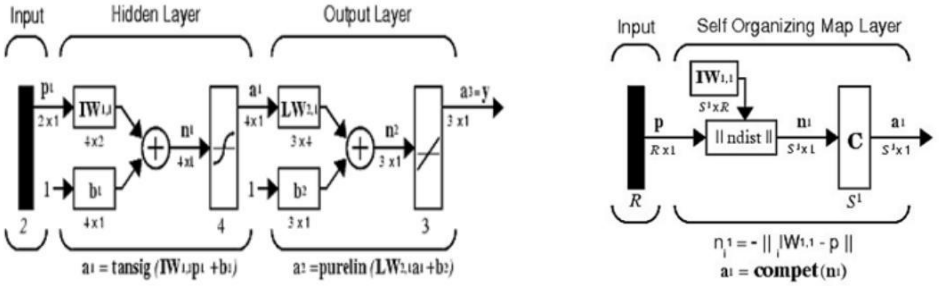
19 features achieved from feature reduction are: A, B, E, F, H, K, L, Q, S, T, V, W, X, Y, AB, AD, AF, AG, AI.

TABLE 2.
Features of KSLKDD database

Label	Network Features	Label	Network Features
A	protocol_type	V	count
B	service	W	srv_count
C	flag	X	serror_rate
D	src_bytes	Y	srv_serror_rate
E	dst_bytes	Z	rerror_rate
F	land	AA	srv_rerror_rate
G	wrong_fragment	AB	same_srv_rate
H	urgent	AC	diff_srv_rate
I	hot	AD	srv_diff_host_rate
J	num_failed_logins	AE	dst_host_count
K	logged_in	AF	dst_host_srv_count
L	num_compromised	AG	dst_host_same_srv_rate
M	root_shell	AH	dst_host_diff_srv_rate
N	su_attempted	AI	dst_host_same_src_port_rate
O	num_root	AJ	dst_host_srv_diff_host_rate
P	num_file_creations	AK	dst_host_serror_rate
Q	num_shells	AL	dst_host_srv_serror_rate
R	num_access_file	AM	dst_host_rerror_rate
S	num_outbound_cmds	AN	dst_host_srv_rerror_rate
T	is_host_login	AO	Sta
U	is_guest_login		

3.1 STEPS OF DESIGNING THE PROPOSED INTRUSION DETECTION SYSTEM USING NEURAL NETWORK

a) Optimizing the number of the middle layers: Of course, the number of the middle layers neurons may be increased during the process of designing neural network. Therefore, in this research, performing a number of various experiments, we considered the most appropriate state (regarding the speed and accuracy of the performance) in such a way that the number of both neurons and layers be at minimum. (Figure 1 shows a MLP with a SOM.)
b) Methods of network training: There is not any significant difference between Batch Training and /or Incremental Training. In this research, we applied batch mode for training.



(a) (b)

FIGURE 1. Two Figures side by side

In this method, firstly, the initial random values for synapses (input values) should be selected. In this research, we have selected these values in the interval of [0 1] (training inputs have been turned into standard data). Then using these values, the process of training for the total data of training and testing has been done and the total classification error for training and testing data has been calculated. Weights trimming treatment of synapses have been done using the method of back-propagation. By iterating the process above (each phase of an iteration is called an Epoch), the Weights of synapses are trimmed in such a way for the total classification error to be at minimum. Well later present necessary description about the halting time of the training.

c) MLP and SOM network architectures: MLP and SOM networks should be determined with the number of layers and the neurons of each layer.In the proposed intrusion detection system, we applied MLP to detect intrusions belonged to DOS and R2L categories and SOM to detect intrusions belonged to U2R and Probing group.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

TABLE 3.
Number of layers and neurons in the proposed neural network

	IDS1	IDS2	IDS3	IDS4
Number of Layers	1	2	2	1
Number of neurons in first layer	12	4	18	15
Number of neurons in second layer	0	7	6	0

d) Optimizing the learning rate of the optimizing algorithm: Evidently, there is a direct connection between optimizing the learning rate of the optimizing algorithm and optimizing the number of the middle layer neurons, or even optimizing other components of the network. So, it seems necessary to mention that this step of the network optimization has been processed simultaneously with the optimization process of the middle layer neurons. Therefore, if here, we have considered the number of the middle layer neurons of the neural network as you see in the Table 3, it is regarding the general performance of the network. However, the optimization process of the number of neural network neurons will be explained in the following steps.In this table, IDS1, IDS2, IDS3 and IDS4 represent intrusion detection systems belonged to DOS, R2L, U2R and Probing respectively.
e) Optimizing the number of the middle layer neurons: In order to optimize the number of the neurons of the middle layer, we performed the Training and Testing operations with neurons of different numbers. Then we selected the number of neurons of the state in which we observed the minimum testing data classification error as the optimal number of neurons. So, the optimal number of neurons will be based on the Table 3. We have performed the Training and Testing operation with neurons of different numbers over 50 times before selecting this optimal number of neurons. Its worth mentioning that the rate of the variability (uncertainty) of the neural network classification is in the interval of its two sequential performances. Therefore, we could say that its so important to know the efficient parameters in the neural network performance and to confront them expertly
f) A number of the neurons of the output layer: The number of the output layer neurons has been considered as equal to the number of the classes (here, 2 classes for each intrusion detection system). In this way, the control coding of the network operation has been simply done by defining a code of 0as correct answers and a code of 1as wrong answers for each class. Another advantage of this output coding method, as well as its simplicity, is enhancing the accuracy of the classification.
g) Type of the activation function of neurons:We need to select an appropriate activation function for the classification process. Regarding many times of trial and errors done in the network training algorithm, we have considered the same appropriate activation function for the neurons of the middle and output layers in the MLP network as Sigmoid Function and for SOM network training, we set

training function as ¨trainrp¨. We also set Topology function and Distance function as ¨tritopand¨ mandist¨, respectively.

h) Stopping conditions: Different conditions may be determined to stop the algorithm: Stopping after iteration of determining times Stopping when error is less than a given value Stopping when error follows a specific rule in the samples of the validation sets.

Generally, there has been no need to more than 400 Epochs for the network training because error reduction of the following training has been invisible.

The general flowchart of the detection process is shown in Figure 2.

The designed IDS acts in binary mode, i.e. the output of this IDS is either 1 or 0, in a way that 0 indicates our specific attack and 1 indicates the non-existence of that attack, that is, if the designed neural network in IDS 1 returns the value of 1, it means that the specific attack in this example has been detected an attack of DOS type intruding the network and if it returns the value of 0, it means that no DOS attack has been detected in inputted data.

3.2 EVALUATION CRITERIA

To measure and detect the efficiency of the designed IDSs or the exact degree of their assurance and correctness, the following criteria can be used:

True negative = correctly detection of the normal data

True positive = correctly detection of the attacks

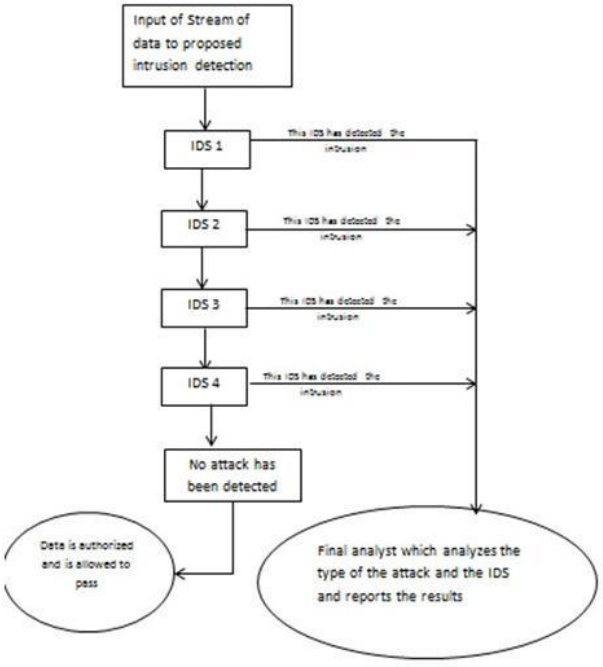


FIGURE 2. Method of intrusion detection using the proposed system

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

TABLE 4.
Confusion Matrix

Expectations

True	Not True		
TN	FP	Normal	Activity
TP	FN	attack	

False Positive = distinguishing normal events as attacks

False negative = distinguishing the attack incidents as normal

TNR = TN / (TN + FP) = the ratio of the total number of normal incidents that are correctly detected to the total number of normal incidents

TPR = TP / (TP + FN) = the ratio of the number of attack incidents that are correctly detected to the total number of attack incidents

FNR = FN / (FN + TP) = the ratio of the number of attack incidents that are detected as normalto the total number of normal incidents

FPR = FP / (FP + TN) = the ratio of the number of normal incidents which are detected as attack to the total number of normal incidents

In this implementation process, we applied TPR and FPR criteria. To implement the neural network algorithm, we applied the MATLAB software (version 7.12.0.635 32bit and March 18, 2011). In order to implement this algorithm, we should firstly train the designed neural network using training data, and then we should analyze the efficiency of the network using the experimentaldata.In addition, in order to design the proposed intrusion detection system we applied MLP neural network to detect DOS and R2L attacks, and SOM neural network to detect U2R and Probing attacks.The computer used for implementation was a 7-Core GHz1 CPU with Ram G2.

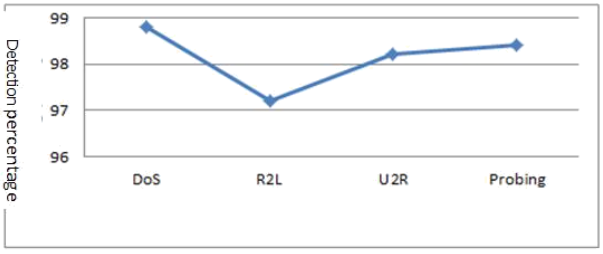


FIGURE 3. Detection Percentage of errors and Warning percentage of errors

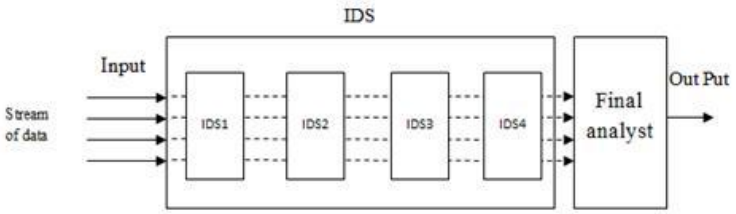


FIGURE 4. Intrusion detection system without using parallelization

4. PERCENTAGE OF INTRUSION DETECTION IN THE PROPOSED SYSTEM

Performing a variety of experiments, the most appropriate algorithm for training was found. This algorithm can detect a high percentage of the attacks. In graph [25], this detection percentage has been presented regardless of the parallelization operation:

TABLE 5.
Detection Percentage of errors and Warning percentage of errors

	TPR	FPR
IDS1	0.985%	1.88%
IDS2	0.972%	2.1%
IDS3	0.982%	1.9%
IDS4	0.973	1.83%
Final IDS	0.984%	1.85%

5. PARALLELIZATION

In order to distribute the performance of the program process, the parallelization tech-nique needs some subprograms and also parallel performance of the subprograms on the system processors to increase the speed of the process performance.

5.1 METHOD OF SEQUENTIAL AND PARALLEL IMPLEMENTATION OF IDSS

Figure 4 shown designing intrusion detection systems, sub-projects (IDSs related to each group) are applied sequentially. In Figure 5, the proposed intrusion detection system using parallelization is shown.

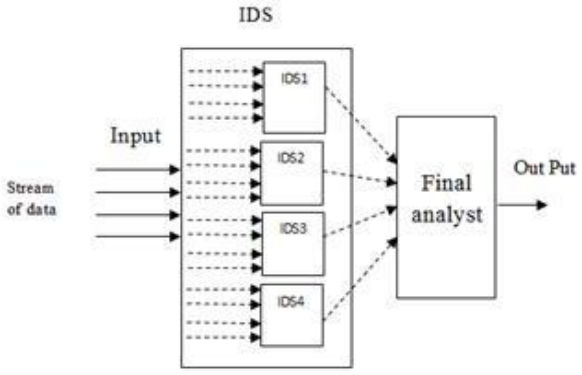


FIGURE 5. Intrusion detection system using parallelization technology

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

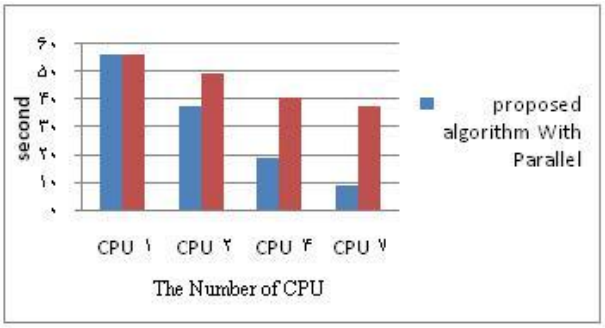


FIGURE 6. Comparison between performance time

As you see in Figure 5, this system has a final analyst which analyzes the results of the IDSs and reports the output in binary mode to the output.

Although, the proposed algorithm has an accepting speed even without paralleliza-tion, but it will reach a much higher speed when using parallelization on the pro-cessors. The Figure 6,shows the improvement rate of the algorithm. In Table 6, the improvement percentage of using multi-processors has been shown.

TABLE 6.
Speed improvement percentage using parallelization technique

Number of whole CPUs used	2 CPU	4 CPU	7 CPU
Speed rate of improvement	27%	65%	80%

6. COMPARISON WITH OTHER SYSTEMS

Several Multi Layout algorithms have been presented for the detection of attacks that are available in articles [7, 8, 11, 16]. Table 7 shows a comparison between several IDSs which can detect attacks.

In addition to the improvement in the intrusion detection percentage, and reduction of the false positive rate in this method, the procedure speed has also been enhanced. The rate of the efficiency improvement in this algorithm depends on the number of the processors available on the system that is arranged to perform IDS operation. Since the algorithm for the proposed intrusion detection is designed in such a way to detect each attack individually, and each designed IDS is located on one separate agent, therefore we apply these agents in a parallel mode on the available processors on the desired system which leads to improve the speed of the procedure.

TABLE 7.
Comparison between designed Intrusion Detection Systems

False positive rate	True positive rate	Classifier
15.74	84.25	Multilayer perceptron (MLP) Resilient_Back Propagation [16]
6.2	97.8	Naive Bayes [11]
5.6	94.9	Decision Tree (C4.5) [11]
8.3	90.3	Support Vector Machine (SVM) [7]
6.21	92	K Meanse Clustering [8]
1.85	98.4	The proposed intrusion detection system

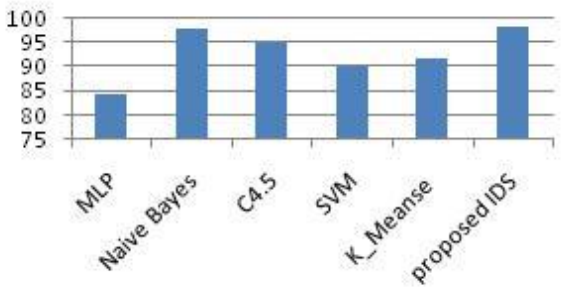


FIGURE 7. Comparison intrusion detection percentage with other methods

7. CONCLUSIONS

There are different algorithms for implementation and network intrusion detection to detect intrusions in computer networks. Depending on what kind of attack and in what degree of accuracy is going to be detected, we choose the appropriate algorithm and the best method of implementation for it.

In the proposed IDS, we used the Supervised Neural Network to detect DOS, R2L, U2R, Probing intrusions in NSLKDD database and we also improved the speed of implementation using Feature Reduction and Parallelization Technologies.

In the proposed IDS, we used misuse-based, also known as signature-based tech-nique. In the used database, it is defined for each record that it is a special kind of attack or a normal connection is done. Based on this, we designed an IDS using the neural network that can detect the attacks, and an agent is considered for each cat-egory of attacks, while the appropriate IDS of the detection of the attack is located on that agent. In other words, the proposed intrusion detection system with 4 Agents performs in parallel and detects the attacks.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

REFERENCES

[1] G. Vasiliadis, and M. Polychronakis ,and S. Ioannidis, ”MIDeA: a multi-parallel intrusion detection architecture,” in Proceeding of the 18th ACM conference on Computer and communications security, New York, 2011, pp. 297-308.

[2] X. D. Hoang,and J. Hu , and P. Bertok , ”A Multi-layer Model for Anomaly Intrusion Detection Using Program Sequences of System Calls”, in Proceeding of The 11th IEEE International Conference ontheICON2003, 2003, pp. 531 536.

[3] G. Wang, J. Hao, L. Huang, ”A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering”,Expert Systems with Applications, vol. 37, (9), pp. 6225-6232, 2010.

[4] The KDD99 Dataset, [online] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html (Accessed January 26, 2008).

[5] S. X. Wu, and W. Banzhaf, ”The use of computational intelligence in intrusion detection systems”, Applied Soft Computing, vol. 10, (1), pp. 135, 2010.

[6] S. M. Abdulla, and N. B. Al-Dabagh, O. Zakaria, ”Identify Features and Param-eters to Devise an Accurate Intrusion Detection System Using Artificial Neural Network”, in Proceeding of the World Academy of Science, Engineering and Technology, 2010.

[7] X. Zhang, and G. Chun-hua, and L. Jia-jin, ”Intrusion Detection System Based on Feature Selection and Support Vector Machine”, Communications and Networking in China,pp: 1-5, 2006.

[8] K. M. Faraoun, and A. Boukelif, ”Neural networks learning improvement using the K-means clustering algorithm to detect network intrusions”, Computational Intelligence, vol. 3, (2), pp. 1618, 2006.

[9] S. Zaman, and F. Karray, ”Features selection for intrusion detection systems based on support vector machinesCCNC’09”, in Proceedings of the 6th IEEE Conference on Consumer Communications and Networking Conference 2009.

[10] H. Nguyen , and K. Franke, and S. Petrovic, ”Improving Effectiveness of Intrusion Detection by Correlation Feature Selection”, in Proceeding of theInternational Conference on Availability, Reliability and Security,IEEE, 2010, pp:17-24.

[11] M. Moradi, and M. Zulkernine, ”A neural network based system for intrusion detection and classification of attacks”, in Proceedings of the IEEE International Conference on Advances in Intelligent SystemsTheory and Applications, Lux-embourg, Kirchberg, 2004.

[12] S. Chebrolu, and A. Abraham , and J. P. Thomas,”Feature deduction and ensem-ble design of intrusion detection systems”, Computers & Security, vol. 24, (4), pp. 295-307, 2005.

[13] B. Amor, ”Naive Bayes vs decision trees in intrusion detection systems”, in ACM symposium on applied computing, Cyprus, 2004, p. 420424.

[14] B. Zhang, ”Internet intrusion detection by auto associative neural network”, in Proceeding of the International Symposium on Information & Communications Technologies, Malaysia, 2005.

[15] A. K. Ghosh,and C. Michael, and M. Schatz, ”A real-time intrusion detection system based on learning program behavior”, in Proceedings of the 3rd International Workshop on Recent Advances in Intrusion Detection (RAID00) , Springer, Toulouse, France, 2000, pp. 93109.

[16] R. Beghdad, ”Critical study of neural networks in detecting intrusions”, Com-puters & Security, vol. 27, (56), pp. 168175, 2008.

[17] T. S. Chou, and K. K. Yen, and J. Luo,”Network Intrusion Detection Design Using Feature Selection of Soft Computing Paradigms”. Computational Intelli-gence, vol. 4, (3), 2008.

[18] D. Parikh , and T. Chen, ”Data fusion and cost minimization for intrusionde-tection”, IEEE Transactions on Information Forensics and Security, vol.3, (6), pp.381389, 2008.

[19] E. Corchado, and L. Herrero, ”Neural visualization of network traffic data for intrusion detection”, Applied Soft Computing, vol 11, (2), pp. 20422056, 2011.

[20] S. T. Sarasamma, and Q. A. Zhu , and J. Huff, ”Hierarchical kohonenen net for anomaly detection in network security”, IEEE Transactions on Systems, Man and Cybernetics- Part B, vol. 35, (2), pp. 302312, 2005.

[21] The KDD99 Dataset, [online] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, (Accessed January 26, 2008).

[22] H. G. Kayacik, and A. N. Zincir-Heywood, and M. I. Heywood, ”A hierarchical SOM-based intrusion detection system”, Engineering Applications of Artificial Intelligence, vol. 20,(4), pp. 439451, 2007.

[23] S. Chebrolu, and A. Abraham, J. P. Thomas, ”Feature deduction and ensemble design of intrusion detection systems”, Computers & Security, vol. 24, (4), pp. 295307, 2005.

[24] Neural Network Toolbox User’s Guide, ”MATLAB User Manual”, Math Works Inc., 2011.

[25] E. Lundin, and E. Jonsson, ”Anomaly-based intrusion detection: privacy con-cerns and other problems”, Computer Networks, vol. 3, (4), pp. 623-640, 2000.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)