ISSN: 0975-766X
CODEN: IJPTFI
Research Article

Available Online through
www.ijptonline.com

AN ENHANCED MULTILAYER PERCEPTRON BASED APPROACH FOR EFFICIENT INTRUSION DETECTION SYSTEM

R. Bala Krishnan[1]*, N.R. Raajan[2]

[1]Srinivasa Ramanujan Centre ; SASTRA University ; Kumbakonam – 612001, Tamilnadu, India

[2]SASTRA University ; Thanjavur, Tamilnadu, India.

Email: balakrishnan@sastra.ac.in

Received on: 20.10.2016 *Accepted on: 25.11.2016*

Abstract:

The growth of Internet applications makes the intrusions on the networking system at a high level. In such a situation, it is essential to offer security to the network systems by an efficient intrusion detection and avoidance mechanisms (IDS and IPS). This can be accomplished by formulating an efficient intrusion detecting systems that works with effective algorithms which can distinguish the abnormal and normal activities and in the network and protects the networked environment and the resources from illegitimate penetrations. Many intrusion detection system applications have been proposed in the past and the systems are having limitations in terms of detection accuracy. To overcome these limitations, the proposed research works offers a new IDS architecture by incorporating Multilayer Perceptron Based approach (MLP) for reducing the rate of false positive alerts and to increase the detection accuracy. The experiments are conducted using KDD CUP'99 dataset and the observation shows that the proposed approach reduces false positive alarms, detects intrusion sensitively, precisely and accurately. Hence the overall accuracy detection has been improved.

Keywords: Intrusion Detection System, Neural Network, Multilayer Perceptron, Anomaly Detection, Misuse Detection.

1. Introduction

The network based services and information over the networks attained a tremendous growth and the services over the network become more significant than ever before. The Intrusion detection mechanisms are the preceding stripe of fortifications against computer attacks in the wake of secure network architecture design and firewalls. In spite of

the embarrassment of available intrusion prevention mechanisms, attacks beside computer systems are still unbeaten. Thus, the intrusion detection systems (IDSs) participate a imperative part in network security [1].

This paper attempts to address the issue of predictive and decision making of false positive attacks in network systems and the attacks are classified as external, which is attempted from various sources [2]. The main purpose of the IDS application is to identify the attack by using the existing knowledge in the format of patterns. The Intrusive attacks identification mechanism monitors the occurrence of events in the network environment and it checks the signs for the occurrence of intrusive behaviours [3 - 6]. The data mining approach is used to check the pattern of the arrived signatures from various sources and from the available knowledge the decision is taken that is either allow or deny the packet. Data mining approaches are used to find the regularities and irregularities in large amount of arriving data patterns. The role of data mining in the intrusion detection is used to spot the user actions and extract it from the collected knowledge, so that it is easy for the analysts to focus on real attacks. The approaches of data mining such as signature based and anomaly based detection are used to identify the intrusive attacks, which are sufficient to identify the false positive attacks and it is one of the major issues. Intrusions are attacks to compromise the system integrity, availability and confidentiality of a computer or network or to bypass its security mechanisms. They are engendered by attackers get into a system via Internet, by authorized users of the systems who attempt to gain added rights for which they are not approved, and by authorized users who use their given prerogatives wrongly. The System alert alarm of the IDS application intimates that there is a try for an illegal penetration attack on our network to access the offered services or to damage the system resources [2]. IDS incorporate the two popular methods for detection processes namely misuse detection and anomaly detection.

The approach Anomaly detection has some merits when compared with the misuse detection such as its capacity to detect attacks with known or unknown signatures. Anomaly detection scheme follows a strategy in its methodology which is finding out the divergence of normal and abnormal user behaviors [3–5]. Misuse detection deals with identifying the user activities on the basis of its access rights and from which it identifies the legal and illegal activities. it is simple in its training, as it is only skilled about the rule that distinguish between authorized and unauthorized users or attacks, but it has some issues. One of the most significant issues is false positives (FPs) [5].

The outcome from the IDS application is the FP, which informs about a user that it is unauthorized penetration or attack although it is authorized. This problem reduces the benefiting of authorized users from the network which gives them the authority or permittivity to use it [6] and it leads to the overload issue in IDS. Hence it would the limiting factor for IDS performance. This issue also causes load on the network management group such as IT personnel and administrator [7].

This research paper discusses this problem and proposed an efficient strategy for designing powerful IDS platform. This strategy improves the efficiency of the existing IDS [2] by incorporating a data mining based knowledge and makes it an automatically adaptive system with the ability of reducing the rate of false positive alerts. The detector part of the proposed IDS is based on Multilayer Perceptron feedforward artificial neural network which used as intelligence for detector and classifier for network based attacks.

The rest of the paper is summarized as follows: Section 2 presents intrusion detection systems; Section 3 states the proposed intrusion detection system (IDS). The experimental observations of the proposed application are stated in Section 4 and the conclusion of the proposed work is stated in Section 5.

2. Related Works

The Intrusion detection system is implemented with several approaches such as machine learning, statistical and data mining. These approaches proceed with the content from the existing knowledge, which is collected from various domains. The conceived knowledge holds the details about the arrival of patterns and its corresponding action.

Major payback of Intrusion Detection System is:

- ✓ Documenting the existing threat from inside and outside a system, permitting security management to realistically assess risk and adapt its security strategy in response.
- ✓ Observing attacks and other security ravishments, which have not been precluded by elementary protection techniques [18].
- ✓ Behaving as quality control for security design and implementation.
- ✓ Keeping problem-deportments by increasing the comprehended risk of uncovering and penalization for those who attacks the system.

A large amount work has been prepared to employ these characteristics [3]. To assist appraisal of these Solutions, IDS research project of Purdue University put a catalogue of features for high-quality systems:

- ✓ The system must run continuously without any human intervention.
- ✓ Needs to hold fault tolerant nature i.e., it must endure a system failure and not lose its knowledge-base at restart.
- ✓ It must be consistent to permit it to work in the background of the system being adverted.
- ✓ It must examine deflections from conventional behaviour.
- ✓ The system would observe itself to guarantee that it has not been subverted.
- ✓ The IDS application must entail negligible transparency on the underlying platform.
- ✓ It should not make a processor to a crawl slowly.
- ✓ It must deal with changing system behaviour over time as new software applications are being included. The profile of the system will change over time; it would be adaptable.
- ✓ Needs capability to adapt easily to the underlying system environment.

Note: All the systems hold a different usage and defence mechanisms signatures and the system have to easily acclimatize to these patterns.

2.1 Overview of IDS Approaches

This session describes some of the famous approaches which are given below that are cusps in the development of intrusion detection system applications.

Data Mining (DM): It holds a group of proficiencies that use the practices of evoking formerly unknown but likely practicable data from large amount of data stores. In other words, this technique allows finding regularities and irregularities in huge input data sets. However, they are memory exhaustive and have need of double storage: one for the normal IDS data and another for the data mining [8-10]. The part of data mining for intrusion detection is to categorize typical user activity and extort it from the composed data so that it is painless for the analysts to spotlight on real attacks, to discover false alarm generators and terrible sensor signatures and also to locate inconsistent activity that uncovers a genuine attack [19].

Data mining often involves four course of assignment:

- ✓ **Association Rule Learning:** It searches for kinships between variables.
- ✓ **Classification:** It formats and maps a data item into one of several pre-defined categories. Popular procedures include Nearest Neighbour mechanism, Naive Bayesian Classifier with or without Neural Networks.
- ✓ **Clustering:** The process is similar to classification but here the data groups are not predefined and hence the clustering procedure attempts to group similar items in to clusters.
- ✓ **Regression:** It seeks to locate a task which models the data set with the slightest variation.

Machine Learning: This approach allows computers to learn from data, such as from sensor data or system databases [11]. A major spotlight of machine learning research is to robotically learn to identify complex patterns and make intellectual decisions on the basis of the data.

Pattern Recognition: A pattern matching technique principally looks for a definite attack pattern [12], which may be given in the record that is audited previously. A human knowledge is mandatory to spot and extort inconsistent elements or patterns from input data [13]. This approach is effectual in reducing the need of maintaining large amount of audit data [14]. This approach is also unable to detect new attack signatures.

Expert System: It is a software system or a hybrid application with software and hardware collaboration capable of performing specific task usually performed by a human expert. Traditional IDSs with Expert systems holds the statistical contents such as the application users, workstations and programs and by regular monitoring of user activities, it identifies the abnormal behavior and it results in the detection of intruders. All the security related features are incorporated in the Expert System based IDS through the audit trail rule and is implemented in terms of if–then-else pattern. The major drawback of the Expert Systems based IDS is that it needs frequent updates by a System Administrator.

Radial Basis Function: The Neural networks have the ability of inferring from input data that is incomplete or limited in terms of its size or parameters [15-16]. Because of this property, they are commonly used in the intrusion detection systems. The neural network followed radial basis function (RBF) has a transcendency of high learning speed and an ability of strong non-linear function approach and mode-classification [17]. RBF neural network is

composed of three layers. Those are input layer, hidden layer and output layer. The structure of RBF neural network is shown in Figure 1.

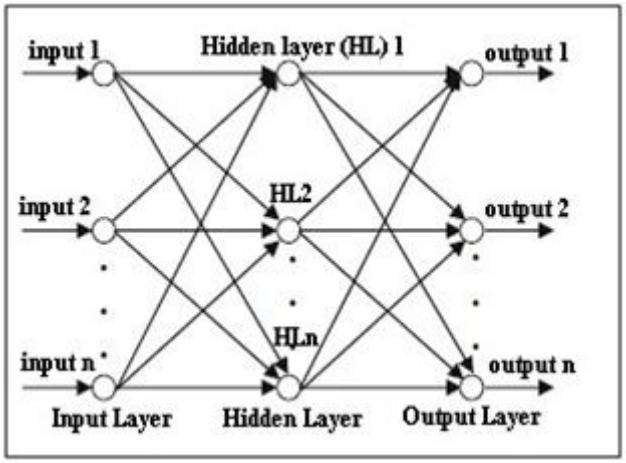

Figure 1. Structured view of RBF Neural Network.

The input layer nodes channelize the input signal to reach hidden layer, hidden layers nodes are described by the Gauss kernel function and output layer nodes are described by the linear function. The kernel function of hidden layer nodes responds to the input signal in local. The property of the analyzing the incomplete data in the RBF neural network results in a wrong prediction in the IDS application, which would be the major limitation of this approach.

Multilayer Perceptron (MLP): It is a feedforward artificial neural network model. It has the ability of processing the network data to classify the normal and abnormal data and hence it suits for IDS applications. It is having the feature of quick learning tendency with strong non-linear activation function. It works with the base of Supervised Learning and follows the backpropagation to train the classifier. In simple words the MLP is a revision of the standard linear Perceptron.

MLP consists of three or more layers (a single layer for input and a single layer for output and the remaining layers are in the intermediate level) of nonlinearly activating layers and is termed as a deep neural network. The structure is similar to the structure of RBF (stated in Figure 1) and the basic activation function of the MLP is defined as follows:

$$y(v_i) = \tanh(v_i) \qquad (1)$$

and $$y(v_i) = (1 + e^{-v_i})^{-1} \qquad (2)$$

the above equ. (1) is a hyperbolic tangent which ranges from -1 to 1, and the equ. (2) is a logistic function, which ranges from 0 to 1. The parameter y_i is the output of the ith node and v_i is the weighted sum of the input layer.

3. Proposed System

The architecture of the proposed IDS is presented in the Figure 2. It works with the KDD Cup'99 Dataset and it is classified into training and testing data. The training data is subjected to Construction Phase, in which the classifier is being build and is updated in the Knowledge Base. The classifier is built using the pre-processed data with features [5], in which the proposed MLP feature is imported to detect the normal request and attacks.

The newly created MLP classifier is then acts as evaluator in the Execution Phase, in which the testing data is imported on to the MLP classifier and it results the type of attacks and the possible attack classes of the IDS are stated in the Table 1.

The Major Classes of Attacks and its action are stated below:

- ✓ *Probing* (*Probe*): Port scanning process, in which the attacker scans the port number of a computer network to extract information.
- ✓ *Denial of Service* (*DoS*): Denying Services to valid user requests.
- ✓ *User to Root* (*U2R)*: Trying unauthorized access to local root privileges, attacker tries to perform illegal penetration into the system.
- ✓ *Remote to Local (R2L)*: attacker trying to perform unauthorized access from the remote system to extract information or to gain access to the system.

Table 1. **Types of Possible Known and Unknown IDS attacks [5].**

Attack Type	Class	Attack	
		Known	**Unknown**
1	Probe	ipsweep, satan and nmap	saint and mscan
2	DoS	neptune, smurf and land	udpstorm
3	U2R	loadmodule, rootkit and buffer_overflow	xterm, ps and sqlattack,
4	R2L	ftp_write, warezmaster and guess_passwd	snoop, worm, xlock and named
5	Normal	-	-

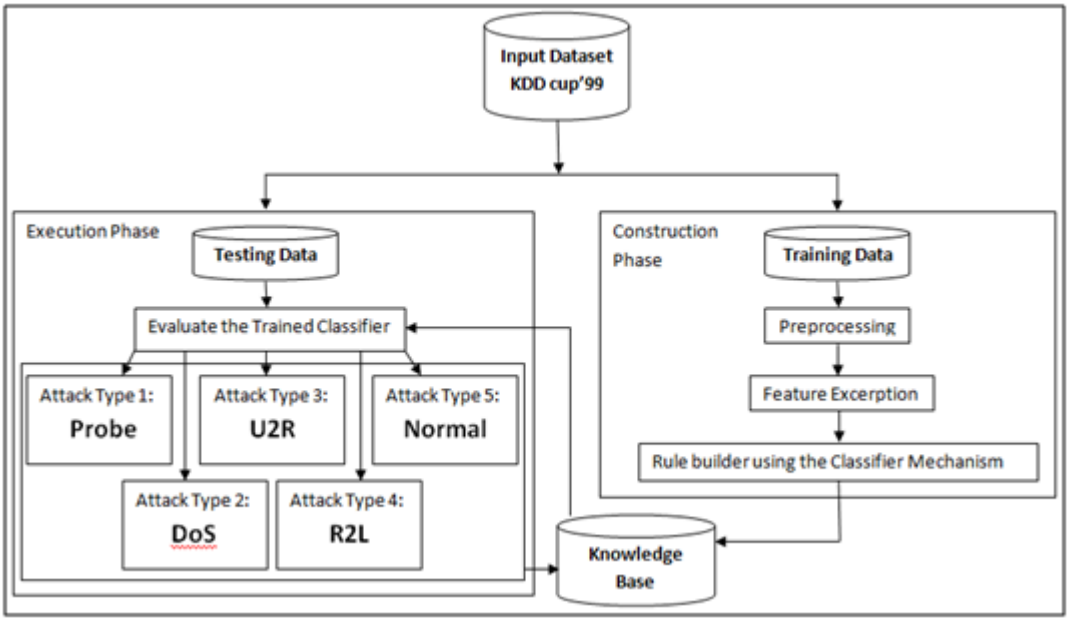

Figure 2. Workflow of the Proposed IDS.

3.1 Construction Phase of the Proposed MLP based IDS

The Construction Phase (CP) of the Proposed MLP works with the KDD CUP'99 dataset which has 42 features [5] and are stated in Table 2. The training data is pre-processed in the CP for cleaning incomplete data and convert it to a complete data. The Feature Excerption is the method of selecting suitable features from the fundamental data set such as KDD Cup data set for building models.

The rule builder mechanism of the proposed MLP scheme constructs the classifier with MLP feature to identify attacks and normal behavior on the testing data. The outcomes of the CP are updated in the KB, which would be referred by the Execution phase (EP). For the exact classification of request type the procedure of selecting necessary features for the attack types are essential. In our proposed MLP approach, the system is trained separately to detect the attack categories. It is to be observed that the attack groups are dissimilar in their nature and hence, it becomes essential to handle them separately. Hence, the projected mechanism selected features for each type of attacks and on the basis the trainer is built for execute the testing data at the Execution Phase.

Table 2. KDD Cup' 99 dataset format (value, name and its Type).

S.No	Name	Type	S.No	Name	Type
1	Duration	Continuous	22	Is_guest_login	Discrete
2	Protocol_type	Discrete	23	Count	Continuous
3	Service	Discrete	24	Srv_count	Continuous
4	Flag	Discrete	25	Serror_rate	Continuous
5	Src_bytes	Continuous	26	Srv_serror_rate	Continuous

6	Dst_bytes	Continuous	27	Rerror_rate	Continuous
7	Land	Discrete	28	Srv_rerror_rate	Continuous
8	Wrong_fragment	Continuous	29	Same_srv_rate	Continuous
9	Urgent	Continuous	30	Diff_srv_rate	Continuous
10	Hot	Continuous	31	Srv_diff_host_rate	Continuous
11	Num_failed_logins	Continuous	32	Dst_host_count	Continuous
12	Logged_in	Discrete	33	Dst_host_srv_count	Continuous
13	Num_compromised	Continuous	34	Dst_host_same_srv_rate	Continuous
14	Root_shell	Continuous	35	Dst_host_diff_srv_rate	Continuous
15	Su_attempted	Continuous	36	Dst_host_same_src_port_rate	Continuous
16	Num_root	Continuous	37	Dst_host_srv_diff_host_rate	Continuous
17	Num_file_creations	Continuous	38	Dst_host_serror_rate	Continuous
18	Num_shells	Continuous	39	Dst_host_srv_serror_rate	Continuous
19	Num_access_files	Continuous	40	Dst_host_rerror_rate	Continuous
20	Num_outbound_cmds	Continuous	41	Dst_host_srv_rerror_rate	Continuous
21	Is_host_login	Discrete	42	Normal or Attack	Discrete

The rule builder works on the principle of the dataset features, in which classification would be done at the initial level to class the types of requests. The complete set of features with name, number and its corresponding class are stated in the Table 3.

Table 3. Features of dataset with Class Type.

S.No (Features)	Class Type
1 – 9	Individual TCP Connections
10 – 22	Domain Knowledge
23 – 31	Traffic Details
32 - 41	Host Information

3.2 Execution Phase of the Proposed MLP based IDS

The Execution Phase (EP) of the Proposed MLP works with the testing dataset of KDD CUP'99 and classifies the requests in to various classes of attacks on the basis of the trainer, which is constructed and updated in the Knowledge Base. The outcomes of the trainer are also updated in the Knowledge Base for future purpose. The Attack types and its mechanism of detection are stated as follows:

The Attack_type 1 [5] of class Probe executes the session duration and packets to be sent. The Attack_type 2 of class DoS layers assures the packets sent by nodes for path finding through inundating procedure and it is useful to limit the swamping or flooding type attacks. The Attack_type 3 of class U2R executes the track of files created,

commands used by the operating system and the type of protocol is being used. The Attack_type 4 of class R2L focuses on the rules and monitors the sessions, services requested from various sources and number of login attempts. In this proposed scheme of classification, the work has been presented as a Enhanced MLP based approach by extending the existing Multilayer Perceptron mechanism [5] in which features for the attacks identification are selected randomly. Application is trained to detect attacks on the basis of its categories. Finally a numerical value named contribution has been assigned here for all features. Based on the cumulative value of contribution the proposed model fixes a threshold value to find the accurate features for all attacks. Identified features are maintained in the Array 'Feature-List (FL)'. The rule builder mechanism works with the FL data to detect an attack and it executes on the base of the collective contribution value of each feature by applying the rules. If a feature collective contribution value is greater than threshold then, the proposed MLP approach elects the particular feature for a type of attack detection.

Procedure: Feature Excerption of Construction Phase.

Input: the dataset with all parameters

Output: Exact parameters for the attack prediction and attack type

Step_1: Read all the input parameters (features) of a record.

Step_2: Identify the features that are essential for the Feature Excerption process using MLP on the input parameters and store the outcome it in the 'FL' array.

Step_3: Apply the Conditional probability on the FL to obtain the contribution value.

Step_4: Store the contribution value in VAL.

Step_5: Compare the VAL with the threshold value for the attack with its constraints.

Step_6: The outcome of Step 5 is Success then the Features has been elected for the attack detection else not.

Step_7: Display the selected parameters and store it in the FL.

Traditional IDS attacks have been classified using the features of the dataset and the efficiency of the system is determined by the total parameters that have been considered for evaluation. The Table 4 shows the comparison of existing and proposed scheme attack types and its corresponding features count.

Table 4. Attacks and its corresponding Features Count.

Attack type	Existing [5] Parameters Count	Existing [2] Parameters Count	Proposed Parameters Count
Probe	5	5	5
DoS	9	5	7
U2R	14	11	11
R2L	8	5	7

By the selected features the rule builder of the classification process has been constructed and these elected features are essential for detecting the four major types of attacks with accuracy. To offer an efficient Classification application of IDS, the features selection would plays the major role and the efficiency of the application may be offered by selecting the minimum features for evaluation, but the accuracy of the application is decided by the selection of the suitable features for attacks detection are stated in Table 8 and the efficiency and accuracy of the proposed scheme is stated with observation in Section 4. The proposed MLP classifier receives the selected features and identifies the attacks on the basis of the content from KB and the execution pattern of the MLP classifier is stated in the Figure 3.

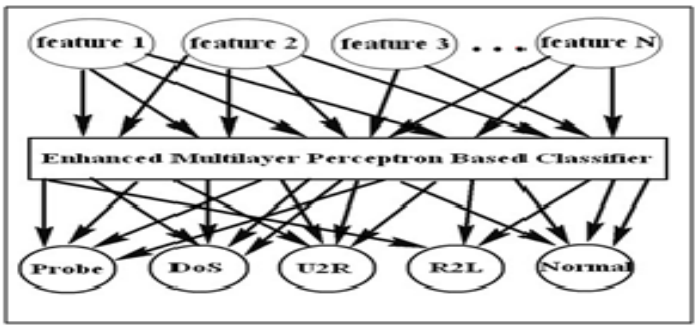

Figure 3. Execution pattern of the Enhanced MLP classifier.

Figure 3. Execution pattern of the Enhanced MLP classifier

Procedure: Proposed Enhanced MLP Based Classification approach

Input: KDD CUP'99 Dataset, Knowledge Base (KB)

Output: Classified dataset with attack labels

Step_1: Read the dataset (input).

Step_2: Extract the Knowledge from KB.

Step_3: Extract the features from the dataset for classification.

Step_4: Apply the knowledge (trained MLP based classifier) from the KB on the processed testing dataset as per the procedure stated in Figure 3.

Step_5: Observe the outcome from the trained classifier.

Step_6: Fix the label for the dataset that is either attack or normal.

Step_7: Check the label with attack to identify the exact pattern of attack.

Step_8: Update the observation from Step_6 and Step_7 on KB.

Step_9: Stop the execution if all the training samples are over else follow the Steps from 1 to 8.

The Proposed mechanism works for identifying the four types of attacks with nature such as Probe, DoS etc., and a normal request on the basis of the rules from KB and the outcomes of the application that is either attack with label or normal request are updated in the KB as knowledge for the future use.

4. Experimental Observations

The proposed IDS application with Enhanced MLP based approach is implemented using JAVA language as front end and WEKA tool as data miner software. The performance of the algorithm is assessed on a Pentium Core 2 Duo system of 2.6 GHz with 4 GB RAM on Windows 8 platform. Moreover, the proposed method have been designed to detect four types of intrusive attacks, where P_i is the number of features that have been considered for the classification from the dataset, N is the number of training dataset, LB is the number of labels and T is the total number of iterations. The Time Complexity measure of the proposed IDS application for the four attack types and it overall complexity are stated in the Table 5.

Table 5. Attacks type and its Time Complexity.

Attack type	Time Complexity
Probe	$O((P_i)\, LB^2\, NT)$
DoS	$O((P_2 - P_1)\, LB^2\, NT)$
U2R	$O((P_3 - P_2)\, LB^2\, NT)$
R2L	$O((P_4 - P_3)\, LB^2\, NT)$
Overall Complexity	$O(P_4\, LB^2\, NT)$

As for the proposed system testing, 10 percentage of the total data is taken as training data for the Construction Phase and 10 percentage of total data is taken as test data (with corrected labels) for the Execution Phase. The detailed structure of the considered Training dataset with Attack Labels is stated in Table 6. The proposed application has been tested with 4.25,028 records. The outcomes of the experiment are the attack classes, which are Normal, Probe, DoS, U2R and R2L.

Table 6. Training Dataset of KDD CUP'99 for the Proposed MLP based Classification.

Attack Class	**Type**	**Number of records**	**Total Percentage**
Probe	Portsweep	4125	1.240
	Satan	6025	1.811
	Nmap	1875	0.563
	Ipsweep	3093	0.930
DoS	Back	1050	0.315
	Land	200	0.060
	Teardrop	740	0.222
	Pod	60	0.180
	Smurt	165600	49.800
	Neptune	65000	19.547
U2R	Perl	4	0.001
	Rootkit	15	0.004
	Loadmodule	30	0.009
	Buffer_overflow	35	0.010
R2L	Phf	7	0.002
	Imap	10	0.003
	Spv	2	0.0
	Warezclient.	2600	0.781
	Guess_Passwd.	10500	3.157
	Warezmaster.	3008	0.904
	Multihop	5	0.001
Normal		68000	20.449
Total		**332524**	**100**

4.1 Evaluation Measures for Accuracy

The Accuracy of the Proposed IDS has been computed through the metrics such as recall, precision and F-Measure, which affect the accuracy of the system. The parameters for the metrics computations are stated in the Table 7.

Table 7. Confusion Matrix for the Attacks Evaluation.

Attack Type	**Connection**	
	Attack	**Normal**
Normal	False Positive -	True Negative- TN

	FP	
Attack	True Positive – TP	False Negative - FN

The equations for the metrics computations are presented below:

The ratio between the relevant and irrelevant data to the context would be computed using the metric 'Recall'. It is computed by the equation:

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)} \qquad (3)$$

The ratio between the relevant and irrelevant data to be conditioned would be computed using the metric 'Precision. It is computed by the equation:

$$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Positive\ (FP)} \qquad (4)$$

The process of comparing the metrics Recall with Precision would be computed using the metric 'F-Measure'. It is computed by the equation:

$$F - Measure = \frac{(1+\beta^2) * Recall * Precision}{\beta * (Recall + Precision)} \qquad (5)$$

The ratio between the detected True Positive (TP) attacks to the total number of Attacks could be detection rate (DR) and it is to be computed as:

$$DR = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)} \qquad (6)$$

The False Alarm rate (FAR) could be computed as:

$$FAR = \frac{False\ Positive\ (FP)}{True\ Negative\ (TN) + False\ positive\ (FP)} \qquad (7)$$

The features of the Dataset considered for the proposed systems attack classification have been stated in the Table 8.The outcomes from the proposed classifier have been stated in the Table 9. It presents the accuracy comparison of the proposed mechanism with existing in-terms of precision, recall and F-measure values. Though the proposed MLP scheme processed the same dataset as of the existing method [2] and the proposed scheme offers an efficient result with high level of accuracy with the selected features, which are stated in Table

Table 8. Features selected for the Attack Classification of the proposed System.

Attack Class	Features	Total	%
Probe	1, 2, 3, 4, 5	5	11.90

DoS	22, 23, 34, 38, 39, 40, 41	7	16.67
U2R	1, 5 ,10, 11, 12, 13, 17, 18, 19, 21, 22	11	26.19
R2L	1, 5, 11, 18, 19, 21, 22	7	16.66

Table 9. Evolution of Accuracy Metrics.

Attack Class	**Existing [2]**			**Proposed**		
	Recall	**Precision**	**F-Measure**	**Recall**	**Precision**	**F-Measure**
Probe	97.8	88.1	92.7	97.8	90.03	94.1
DoS	97.05	99.98	98.5	97.05	99.9	98.71
U2R	62.3	55.07	58.1	63.8	59.2	59.3
R2L	27.08	94.7	42.0	29.3	94.71	44.8

Table 10. Evolution of Execution Time (in secs).

Attack Class	**Existing [2]**		**Proposed**	
	Training Time	**Testing Time**	**Training Time**	**Testing Time**
Probe	6.9	2.04	7.59	4.02
DoS	26.59	15.17	22.5	12.9
U2R	5.30	5.96	6.01	7.02
R2L	0.85	2.67	2.5	2.05

The Table 10 states the execution time for the construction and execution phase of the proposed plan and is compared with the training and testing time of the existing system. The proposed system offers an accurate output for the input data set and the execution time purely depends on the underlying system configuration and the total number of instances for the computation.

The Table 11 states the instances count comparison in the training phase of the existing model with the proposed plan. The accuracy of the proposed plan is compared with the existing classification methods like decision tree, C4.5 etc., and the observations shows that the proposed plan works better and offers an efficient result in terms of its accuracy. The comparison results are presented in the Table 12.

Table 11. Evolution of Training Dataset Count.

Attack Class	Existing [5]	Proposed
Probe	13,860	15,118
DoS	2,15,143	2,33,190
U2R	70	84
R2L	14,845	16,132
Total	2, 43,918	2, 64,524

Table 12. Comparison of the Classification mechanisms accuracy.

Classification Approach	Attacks			
	Probe	DoS	U2R	R2L
C4.5 (Decision Tree) [2]	80.8	97.0	1.80	4.60
Enhanced C4.5 [2]	81.5	97.12	6.24	12.57
Conditional Random Field [2]	98.6	97.4	86.3	29.6
Existing MLP [2]	88.70	97.2	13.2	5.60
Enhanced Multilayer Perceptron (Proposed)	98.71	97.4	87.4	31.3

5. Conclusion

In this research work a new intrusion detection system have been presented with MLP based approach that improves the false positive intrusive attacks detection accurately with time efficiency. Experimental results states that the proposed work with the Construction and Building Phase works effectively and the outcomes of the proposed model is compared with the existing system outcomes. From the comparative study it is to be evident that the proposed model works effectively and it is well suitable for detecting the above mentioned four types of attacks with considerable execution time. The Layered approach of the proposed MLP scheme significantly offers accurate attack detections and hence it is suitable for detecting the four major types of attacks. The proposed work could be planned to upgrade with temporal models and it would the future direction of this research work.

References

1. Nawal A. Elfeshawy, Osama S. Faragallah, "Divided Two-part Adaptive Intrusion Detection System," Journal of Wireless Networks, Digital Object Identifier: 10.1007/s11276-012-0467-7, 2012, Springer.

2. Sannasi Ganapathy, Pandi Vijayakumar, Planichamy Yogesh and Arputharaj Kannan, "An Intelligent CRF Based Feature Selection for Effective Intrusion Detection", *The International Arab Journal of Information Technology*, vol 13, No. 1, 2016.
3. Durst, R., Champion, T., & Witten, B. (1999). Testing and evaluating computer intrusion detection systems. Communications of the ACM, 42(7), pp. 53–61.
4. Drum, R. (2006). IDS and IPS placement for network protection, CISSP, pp. 152–160.
5. Preecha Somwan and Woraphon Lilakiatsakun, "Anomaly Traffic Detection Based on PCA and SFAM" *The International Arab Journal of Information Technology*, vol.12, no 3, pp 253-260, 2015.
6. Linda, O., Vollmer, T., & Manic, M. (2009). Neural network based intrusion detection system for critical infrastructures, IJCNN'09, international joint INNS-IEEE conference on neural networks, Atlanta, Georgia, pp. 15–23.
7. Zhou, J., Carlson, A. J., & Bishop, N. (2005). Verify results of network intrusion alerts using lightweight protocol analysis, computer security applications conference IEEE computer society, pp. 52–60.
8. Georgios, P, & Sokratis, K. (2009). Reducing false positives in intrusion detection systems, Department of Computer Science and Biomedical Informatics, University of Central Greece, available on Science Direct Search.
9. Kurose, J., & Ross, K. (2001). Computer networking: A top-down approach featuring the internet. Boston: Addison-Wesley.
10. Lippmann, R., Haines, J. W., & Fried, D. J. (2000). The 1999 DARPA Off-line intrusion detection evaluation. The International Journal of Computer and Telecommunications Networking, 34(4), pp. 579–595.
11. Marin, J., Ragsdale, D., & Surdu, J. (2001). A hybrid approach to the profile creation and intrusion detection, DARPA information survivability conference and exposition (DISCEX II'01), Vol I.
12. Shieh, S.-P. & Gligor, V. D. (1997). On a patter-oriented model for intrusion detection, IEEE transactions on knowledge and data engineering, Vol. 9, No. 4.

13. Shieh, S. P., & Gligor, V. D. (1991). A pattern-oriented intrusion detection system and its applications. Proceedings of IEEE symposium research in security and privacy. Oakland, CA. pp. 327–342.
14. Kumar, S. (1995). Classification and detection of computer intrusions, Ph.D. dissertation, Purdue University.
15. Golovko, V., & Kochurko, P. (2005). Intrusion recognition using neural networks, IEEE workshop on intelligent data acquisition and advanced computing systems: Technology and applications, Sofia, Bulgaria, pp. 108–111, 5–7 September.
16. Zhong, J., Li, Z., Feng, Y., & Ye, C. (2006). Intrusion detection based on adaptive RBF neural network. IEEE proceedings of the sixth international conference on intelligent systems design and applications, pp. 1081–1084.
17. Montazer, G. A., Sabzevari, R., & Khatir, H. G. (2007). Improvement of learning algorithms for RBF neural networks in a helicopter sound identification system. Neurocomputing, 71(1–3), pp. 167–173.
18. Alsharafat W., "Applying Artificial Neural Network and eXtended Classifier System for Network Intrusion Detection," *the International Arab Journal of Information Technology*, vol. 10, no. 3, pp. 230-238, 2013.
19. Chimphlee W., Abdullah A., Sap M., Chimphlee S., and Srinoy S., "A Rough-Fuzzy Hybrid Algorithm for Computer Intrusion Detection," *the International Arab Journal of Information Technology*, vol. 4, no. 3, pp. 247-254, 2007.