IJARIIT INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 4, Issue 4)
Available online at: www.ijariit.com

Intrusion detection based on ANN by analyzing network traffic parameter

Rahul R Bhoge
bhoge.rahul@gmail.com
Prof. Ram Meghe Institute of Technology & Research, Amravati, Maharashtra

Dr. M. A. Pund
mapund@mitra.ac.in
Prof. Ram Meghe Institute of Technology & Research, Amravati, Maharashtra

ABSTRACT

Nowadays, every individual is interchange the data from information systems that are more open to the Internet and communication medium, the value of security of networks is extremely increased because of its tremendous utilization. In Internet different types of server are connected to each other now they are under the threats of network attackers. Actually, Intrusion Detection System (IDS) is the second level of defense for which it can be the most powerful system that handles the Attacks done at computer System by making alerts to do the analysts take some sort of actions to prevent this Attacks. IDS are based on the Principle of that an attacker behavior will be clearly different from that of a genuine user. In the proposed system we use a KDD-NSL dataset which will be as the first line of implementation for collect different attribute related to network packet then extract certain attributes from the actual dataset and use such attribute parameter is used to make training dataset and save it into the database. Our training dataset includes 4500+ data rows of values and forty-one attributes. Then in next step is to implement a real-time IDS again find out the different network packets features from dataset according attribute then load training dataset then apply artificial neural network algorithm which is work in three layers input layer, output layer and hidden layer which is a Back Propagation (BPN) and Feed Forward (FF) algorithms so that it provides two outputs normal packets and attacks packet. Proposed system evaluated on the base of performance are classified correctly for both anomaly-based detection and misuse based detection using a dataset of network packets and normal packets.

Keywords*— Intrusion Detection System (IDS), Artificial Neural Network (ANN), NSL-KDD dataset*

1. INTRODUCTION

An Intrusion Detection System (IDS) Checks the behavior of network or computer system activity for some suspicious pattern to recognize that any type of system policy might have overruled. The IDS itself a passive system, it always has some component which can alert at malicious activity [1].

There will be two types of IDS most popularly designed based on their place in network or system, one is Network-based IDS other is Host-based IDS. In Network IDS we need to place some component of IDS in Network so that we can identify the malicious activity in the system. NIDS is placed either in-network tap, span port, or hub, so Packet will be captured for data while in a traverse from their location, and then we have to identify a malicious or unauthorized access. In this paper, we proposed the system that also makes differentiate a malicious or normal activity of the Network by analyzing a dataset [2] [3] [4] [5].

IDS are basically two types as one is based on anomaly and other is based on misuse. In anomaly-based IDS it will classified packet in normal and remaining will be malicious packet while in misuse based IDS will be classified packet dataset as malicious and remaining will be considered as a Normal packet.

In this paper, we provide a classification of attacks according to their detection in the runtime environment and also based on either anomaly attack or misuse attack detection of specific dataset file going to find by network packet attribute. There is a different type of parameter associated with network packet like Destination port, Flag, DestIP Ratio, DestPort Ratio, DestIP & Port Ratio, SynAckRatio, NumDestIps, which we have to identify for fault finding in network where attacks detection is possible by using ANN so can training and testing will be done on dataset and that result must be in the statistical format as confusion matrix and overall performance of system to identify attacks as threshold value [10] [11].

2. RELATED WORK

IDE is a most important factor is network security which already done by many researchers which is more about to implementing idea about IDE and Defense mechanism so that makes protection from intruders.

Intrusion Detection is not an activity that will be used to find out a past intrusion into the network or system, it will be used a past experience, so that we might have able to detect such kind of malicious activity that happens currently with system by using a machine learning algorithm that is nothing but an artificial neural network, so learning thing continuously and making the alert.

NSL KDD dataset has most of the solution [6], [7], [8] that have been collected by analyzing different dataset available with the system. This dataset contains a set of an attribute of network communication signatures representing standard and malicious communication. The most important to training by selecting a proper parameter of network parameter from the packet, this is already mentioned by More and Tahalkar while working with a DDoS defense system that would be implemented with real-time IDE with some specific no of the packet and some less amount of packet Attribute [4].

3. PROPOSED NEURAL NETWORK BASED IDS

Figure 1 shows a different part of intrusion detection to analyzing parameter of traffic on the network so can actually cause of attack can identify as well as it will preprocess to form future criteria to detect the attack. Most of the time attack can happen multiple times so it takes to create a repository. Following the model included feature selection from NSL KDD dataset, the neural network model, and different software tool minimize the unused parameter, that is to minimize overhead so computation is done fast.

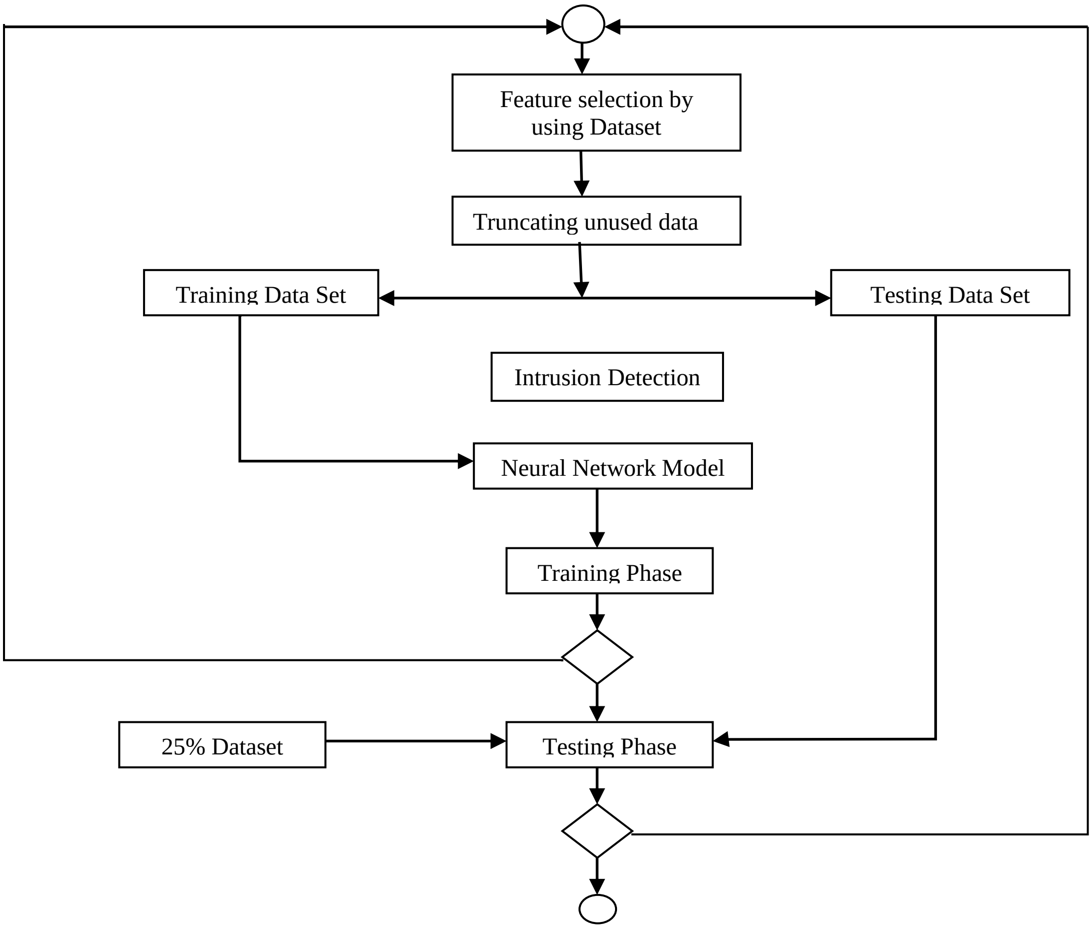


Fig. 1: Intrusion detection Model based on Neural network

KDD NSL Dataset

While detecting malicious activity its always important to the selection of a proper parameter, every attribute related to network packet is important but it is not possible to use all parameter to contribute their value to detecting malicious activity. In proposed system we can make dataset with 41 attributes and the dataset have more than 4500+ entries for all 41 parameters of network packet, in this some attribute might have been more important to analyze the IDS functioning, so we can make it minimize by using Weka is already provided with standard dataset NSL KDD, so might have possible to select some important parameter based on requirement of situation arises in network [13] [14] [15] [16].

ANN Training

In this ANN Training train neural network with 75% dataset of the selected database. Training dataset is as close as possible to an actual dataset which we expected to see. Our training dataset is made from certain generalize minimum attribute after truncation of attributes. So when we train any classifier of an attribute using the training dataset then classifier model is built and when any actual data come to that classifier then it must submit to the prediction of attack.
Real-Time Intrusion Detection

An ID is selecting a real-time packet that is in network traffic, it is captured and extracted so that we can find out its attribute for the dataset. In the proposed system, we use the predefined NSL-KDD dataset to identify and classify attacks by artificial neural network classifier. Identified attacks is might have classify in SYN Flood, TCP Flood, UDP Flood, Ping Flood any type of Denial of Service attack, Probing Attack (PROBE), Users to Root Attack (U2R), buffer overflow attacks, Remote to Local Attack (R2L), guessing password, N map attack are come from any system so for detecting this types of attack our system is more useful [19] [20]. We mostly did our work in the form of the dataset so while detecting real-time intrusion first to convert it into dataset but for in our proposed system we use 75% of a dataset of KDD NSL for training a Neural network and other 25% are used to make intrusion detection purpose[16][17].

4. IMPLEMENTATION OF PROPOSED SYSTEM

4.1 Dataset utilized

Duplicate records in both training and testing datasets divide results for frequent attacks and normal instances. KDD-NSL is a large dataset for most machine learning algorithms; therefore, we use for our studies a small percentage of it [18].

Table 1: Details of Dataset

Name of Dataset	No. of Attributes	No. of Records
Dataset_Anomaly	41	4500+
Dataset_Misuse	41	4500+

It is carried out using Weka. The first objective of our approach was to simplify the data that is to be processed. Simplifying would mean removing attributes that did not make sense. The advantage is that removing attributes would reduce the size of data getting processed which would increase the performance of the neural network. The downside to this can be if important attributes are accidentally removed then the accuracy of detecting an intrusion will suffer. The way out of this was using various iterations of removing certain attributes and then using certain attributes to figure out what suits best. We used **Weka's RemoveUseless()** that helps remove attributes that usually do not vary much.

4.2 Data Preparation

Java file is used to prepare data. Data is prepared by extracting rows randomly from each of the files named as Optimized_'name_of_attack' in the data folder. For eg: Optimized_FTPWrite. The Dataset files generated after data preparation are Dataset_Anomaly and Dataset_Misuse which would later be used by the neural network.

4.3 ANN Training

ANN Classifier has used the dataset to Train for classification of attacks once at a time of building classification as compared to another classifier. ‘neural net’ used Feed Forward (FF)[19] and Back Propagation Neural Network (BPNN)[19] for calculation of classifier so it makes fast classification otherwise naïve-byes required more computing time and it trains again and again when the actual dataset comes to that classifier.

4.4 Real-Time Intrusion Detection

For carrying out intrusion detection for Anomaly-based attacks and Misuse based attacks we had two data sets Dataset_Anomaly and Dataset_Misuse in the previous Module. In the anomaly detection data set, the class or prediction variable is either **Normal** which represents a normal case or an **Attack.** Contrary to the anomaly detection data set, the misuse detection data set has a class variable **Normal** or **Name of the attack** which represents a specific type of attack such as Smurf, NMap, Rootkit, etc. We carry out data cleaning on Dataset_Anomaly and Dataset_Misuse using Weka's to obtain Dataset_Anomaly_Selection & Dataset_Misuse_Selection which has fewer attributes that help speed our Neural Network. Standard dataset KDD NSL are recognized using an artificial neural network classifier algorithm.

This classifier work in three layers Input Layer, Output Layer, and Hidden Layer.
Hidden Layer = Input Layer + Output Layer + 1

5. ANN ALGORITHM

Input: Real-time Dataset in the form of Microsoft Excel file (D).
Output: Elements of Data classified into two 0, 1 means normal or Attack.

(i) First apply the inputs to the network, so that output could be random, as initial output could be anything, so these initial outputs were remembered.

(ii) Next, let it be finding out the error for neuron B. The error is that you expected and actually get, in other words:

Error B = Output B (1-OutputB) x (Target B. Output B)

Output (1-Output) term is necessary for the equation because of the Sigmoid Function.

(iii) Let’s find out the new weight. Let W+AB be the new (trained) weight and WAB be the initial weight that we consider in 1

W+ AB = WAB + (Error B x Output A)

Notice that it is the output of the connecting neuron (neuron A) we use (not B). We update continuously all the weights in the output layer in this way.

(iv) Likewise, we have to calculate the Errors for the hidden layer neurons. Unlike as we done for the output layer we can’t calculate these directly, so we need to Back Propagate them from the output layer (hence the name of the algorithm). This is done by taking the Errors from the output neurons and running them back to initial neuron weights to get the hidden layer errors. For example, if neuron A, B, and C then

Error A = Output A (1 – Output A) (Error A WAB + Error C WAC)

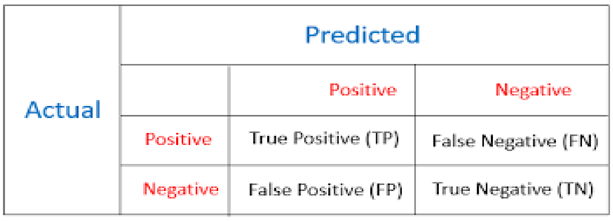
(v) Having to obtain the Error for the hidden layer neurons.

(vi) Let's proceed as in stage 3 to change the hidden layer weights. By repeating this Process we can train a network for any number of the set of layers.

The artificial neural network gives output in the form of yes or no means an attack is present or not and which type of attack is present. While using artificial neural network making the use of Backpropagation neural network we are used in training dataset means input and output both are present. And in Feed-Forward method input is given to the classifier but the output is predicted from the classifier. This is the beauty of ANN.

6. RESULT EVALUATION

The Performance of classifiers is evaluated using different merits are as follows with Confusion Matrix, In the field of machine learning and specifically the problem of statistical classification, a confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning. Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class (or vice versa). The name stems from the fact that it makes it easy to see if the system is confusing two classes (i.e. commonly mislabeling one as another) [4] [20].



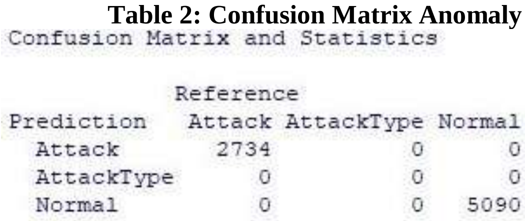
Actual		Predicted	
		Positive	Negative
	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Fig. 2: Confusion matrix

6.1 Anomaly Detection using NN

The actual result we get by executing script below Confusion Matrix for Dataset Anomaly Detection

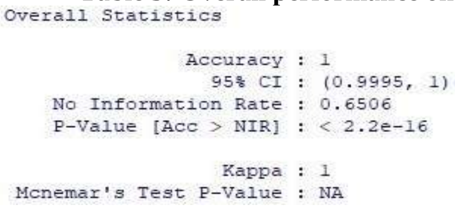
Table 2: Confusion Matrix Anomaly



```
Confusion Matrix and Statistics

             Reference
Prediction    Attack AttackType Normal
  Attack        2734          0      0
  AttackType       0          0      0
  Normal           0          0   5090
```

We got to know that how the system makes overall performance of the neural network for Dataset Anomaly-based detection and we find out that their will speedily execution by using neural network this is shown as below.

Table 3: Overall performance on Anomaly

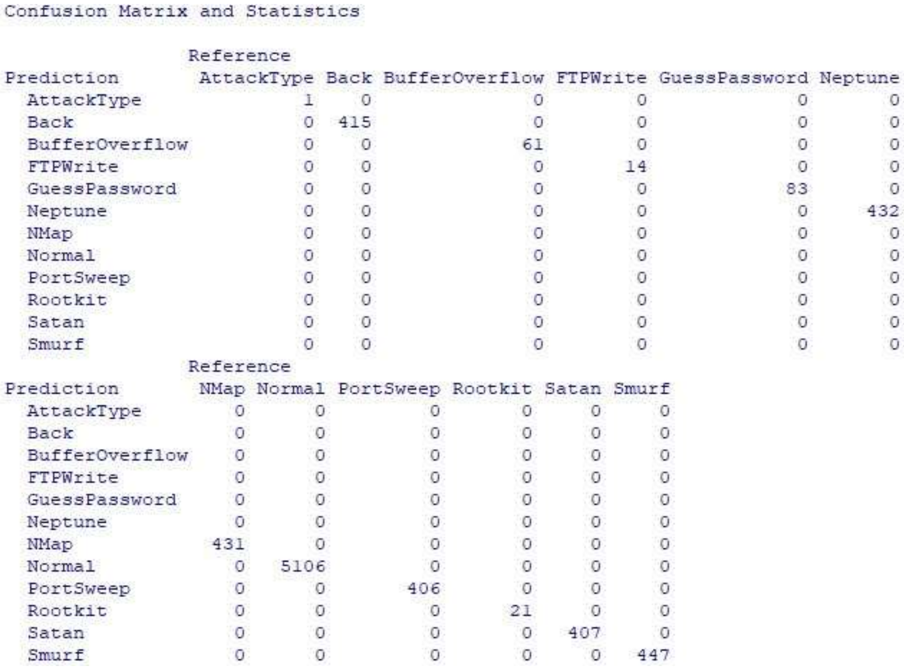


```
Overall Statistics

               Accuracy : 1
                 95% CI : (0.9995, 1)
    No Information Rate : 0.6506
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 1
 Mcnemar's Test P-Value : NA
```

6.2 Misuse Detection using NN

The actual result we get by executing script below Confusion Matrix for Dataset Misuse Detection

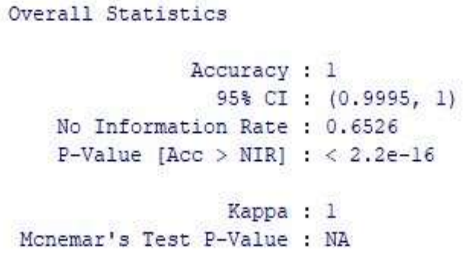
Table 4: Confusion matrix misuse



```
Confusion Matrix and Statistics

                Reference
Prediction        AttackType Back BufferOverflow FTPWrite GuessPassword Neptune
  AttackType               1    0              0        0             0       0
  Back                     0  415              0        0             0       0
  BufferOverflow           0    0             61        0             0       0
  FTPWrite                 0    0              0       14             0       0
  GuessPassword            0    0              0        0            83       0
  Neptune                  0    0              0        0             0     432
  NMap                     0    0              0        0             0       0
  Normal                   0    0              0        0             0       0
  PortSweep                0    0              0        0             0       0
  Rootkit                  0    0              0        0             0       0
  Satan                    0    0              0        0             0       0
  Smurf                    0    0              0        0             0       0
                Reference
Prediction        NMap Normal PortSweep Rootkit Satan Smurf
  AttackType         0      0         0       0     0     0
  Back               0      0         0       0     0     0
  BufferOverflow     0      0         0       0     0     0
  FTPWrite           0      0         0       0     0     0
  GuessPassword      0      0         0       0     0     0
  Neptune            0      0         0       0     0     0
  NMap             431      0         0       0     0     0
  Normal             0   5106         0       0     0     0
  PortSweep          0      0       406       0     0     0
  Rootkit            0      0         0      21     0     0
  Satan              0      0         0       0   407     0
  Smurf              0      0         0       0     0   447
```

The overall performance of the neural network for Dataset while Misuse detection.

Table 5: Overall performances on Misuse



```
Overall Statistics

               Accuracy : 1
                 95% CI : (0.9995, 1)
    No Information Rate : 0.6526
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 1
 Mcnemar's Test P-Value : NA
```

We can plot a graph of performance in NN while Executing both Anomaly and Misuse based detection of attacks.

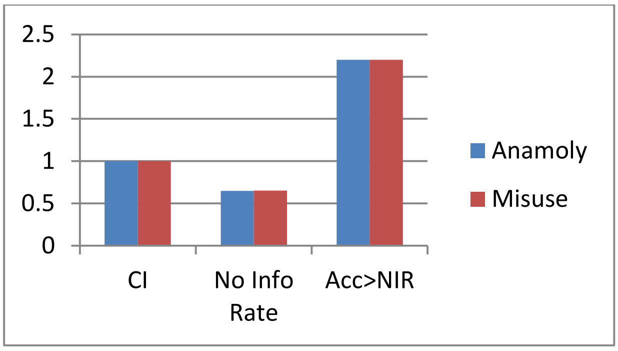


Fig. 3: Anomaly vs. Misuse detection

7. CONCLUSION

The results that we have above are drawn from running tests for 3 approaches, the summary and snapshots are attached above. The original dataset that we had with 41 attributes which were reduced to 36 using Weka's RemoveUseless() and then by using a neural network we able successfully classify the network dataset into different attack types. The Proposed intrusion detection system gives higher accuracy for detection of attacks. ANN is the best classifier for intrusion detection system. ANN classifies the detected attacks more accurately in less timing. As compared to other algorithms ANN gives 99.60 percent accuracy.

8. REFERENCES

[1] Charles P. Pfleeger, Shari Lawrence Pfleeger, “Security in Computing”, Pearson; 4th edition (2009), 527 pages.

[2] Norbert Ádám, Branislav Madoš, Anton Baláž, Tomáš Pavlik, “Artificial Neural Network based IDS”, in IEEE 15th International Symposium on Applied Machine Intelligence and Informatics, January 26-28, 2017, pp.159-164.

[3] Basant Subba, Santosh Biswas, Sushanta Karmakar, “A Neural Network Based System for Intrusion Detection and Attack Classification”, in Twenty-Second National Conference on Communication (NCC), Year: 2016, pp. 1- 6.

[4] More Amruta, Nitin Talhar, “Effective Denial of Service Attack Detection using Artificial Neural Network for Wired LAN”, International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)-2016, IEEE, PP 230-235.

[5] L. Vokorokos, A. Baláž, B. Madoš, “Anomaly and Misuse Intrusions Variability Detection” in Acta Electrotechnica et Informatica, Vol. 10, No. 4, 2010, pp. 5-9.

[6] L. Vokorokos, A. Baláž, N. Adám, “Events Planning in Intrusion Detection Systems” in Acta Electrotechnica et Informatica, Vol. 7, No. 4, 2007, pp. 82-86.

[7] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” in IEEE Symposium on Computational Intelligence for Security and Defense Applications, July 2009, pp. 1–6.

[8] J. Ryan, M. Lin, and R. Miikkulainen, “Intrusion Detection with Neural Networks,” in Advances in Neural Information Processing Systems 10, [NIPS Conference, 1997, pp. 943–949.

[9] J. Cannady, “Artificial Neural Networks for Misuse Detection,” In National Information Systems Security Conference, 1998, Pp. 443–456.

[10] S. Mukkamala, G. Janoski, and A. Sung, “Intrusion detection using neural networks and support vector machines,” in Proceedings of the 2002 International Joint Conference on Neural Networks, 2002. IJCNN ’02, vol. 2, 2002, pp. 1702–1707.

[11] S. Mukkamala, G. Janoski, and A. Sung, “Intrusion detection using neural networks and support vector machines,” in Proceedings of the 2002 International Joint Conference on Neural Networks, 2002. IJCNN ’02, vol. 2, 2002, pp. 1702–1707.

[12] P. L. Nur, A. N. Zincir-Heywood, and M. I. Heywood, “Host-Based Intrusion Detection Using Self-Organizing Maps,” in Proceedings of the IEEE International Joint Conference on Neural Networks, 2002, pp. 1714–1719.

[13] F. E. Heba, A. Darwish, A. E. Hassanien, and A. Abraham, “Principle components analysis and Support Vector Machine based Intrusion Detection System.” in ISDA, 2010, pp. 363–367.

[14] S. Devaraju, S. Ramakrishnan, “Performance Comparison for Intrusion Detection System Using Neural Network with KDD Dataset”, in Ictact Journal on Soft Computing, April 2014, vol. 04, issue 03, pp.743-752.

[15] L.L. Ray, “Training And Testing Anomaly-Based Neural Network Intrusion Detection Systems”, in International Journal of Information Security Science, vol. 2, no. 2, pp. 57-63.

[16] Atilla Ozg, ur _ Hamit Erdem, ” A Review of KDD99 Dataset Usage in Intrusion Detection and Machine Learning between 2010 and 2015”, PeerJ Preprints, https://doi.org/10.7287/peerj.preprints.1954v1, 4.0 Open Access,14 Apr 2016.

[17] Z. Zhou, Ch. Zhongwen, Z. Tiecheng, G. Xiaohui, “The Study On Network Intrusion Detection System of Snort”, in Proceedings of The 2nd International Conference on Network and Digital Society (ICNDS), Wenzhou, China, Hong Kong Section CAS/ COM Joint Chapter, Guizhou University, Peking University, 2, pp. 194-196, 2010.

[18] Gajanan P. Bherde, M. A. Pound, “Recent attack prevention techniques in web service applications”, 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), 9-10 Sept. 2016, pp 1174 – 1180.

[19] M.A.Pund, S.V.Athawale, “NGIPS: The roadmap of next-generation intrusion prevention system for wireless LAN”, 2017 1st International Conference on Intelligent Systems and Information Management (ICISIM), Year: 2017, pp 276-280.

[20] Shwetambari G. Pundkar, G. R. Bamnote, “Analysis of Firewall Technology in Computer Network Security”, International Journal of Computer Science and Mobile Computing (IJCSMC), Vol 3 Issue 4, April- 2014, pp 841-846.