DOI: http://dx.doi.org/10.30684/etj.36.2B.3

Sarah M. Shareef
Computer Science Department, University of Technology, Baghdad, Iraq
sarahshareef84@gmail.com

Soukaena H. Hashim
Computer Science Department, University of Technology, Baghdad, Iraq.
soukaena.hassan@yahoo.com

Received on: 25/09/2017
Accepted on: 20/12/2017
Published online: 25/12/2018

Intrusion Detection System Based on Data Mining Techniques to Reduce False Alarm Rate

Abstract- *Nowadays, Security of network traffic is becoming a major issue of computer network system according to the huge development of internet. Intrusion detection system has been used for discovering intrusion and to maintain the security information from attacks. In this paper, produced two levels of mining algorithms to construct Network Intrusion Detection System (NIDS) and to reduce false alarm rate, in the first level Naïve Bayes algorithm is used to classify abnormal activity into the main four attack types from normal behavior. In the second level ID3 decision tree algorithm is used to classify four attack types into (22) children of attacks from normal behavior. To evaluate the performance of the two proposed algorithms by using kdd99 dataset intrusion detection system and the evaluation metric accuracy, precision, DR, F-measure. The experimental results prove that the proposal system done high detection rates (DR) of 99 % and reduce false positives (FP) of 0 % for different types of network intrusions.*

Keywords- *data mining, intrusion detection system, false alarm, Decision Tree classifier, Naïve Bayes classifier..*

How to cite this article: S.M. Shareef and S.H. Hashim, "Intrusion Detection System Based on Data Mining Techniques to Reduce False Alarm Rate," *Engineering and Technology Journal*, Vol. 36, Part B, No. 2, pp. 110-119, 2018.

1. Introduction

Security of network traffic is appealing the main issue of computer network system Because of the large development of internet; therefore, intrusions on the internet are growing day-by-day. Intrusion detection system has been utilized for detecting intrusion and to maintain the security goals of information from attacks [1]. An IDS is a combination of software and hardware which are utilized to detect intrusion. It collects and analyzes the network traffic from malicious patterns and alert to the proper department [2]. Data mining is the operation of find out interesting knowledge from huge quality of data stored in databases, data warehouses, or other information stores [3]. It is involved methods at the intersection of artificial intelligence, machine learning, and database systems. Data mining applications can utilize the different of parameters to check the data [4]. Feature selection is an operation that chooses a lower subset of features from the main set features, it decreases the dimensionality of feature area, eliminates redundant, irrelevant. Selection feature has effectiveness on intrusion detection systems performance [5].

Classification is one of the most data mining methods that can be utilized for decision making. Naïve Bayes classifier is a simple probabilistic based method, which can foretell the class membership probabilities and it has some advantages: easy to utilize and only one scan of the training data desired for probability generation. A NB classifier can be easily handle missing values of attribute by deleting the corresponding probabilities for those attributes when computing the likelihood of membership for each class. It can be also desired the class conditional independence [6]. Decision tree technique is a fast classification method. Its build operation is top-down, divide-and-rule. Beginning from root node, for each non-leaf node, first step choose an attribute to test the sample set; Second step split training sample set into several subsample sets according to testing outcomes ,each sub-sample set frames a new leaf node; Third step repeat the overhead division operation, until having arrived particular end states [7] [8].

The paper introduces an approach for analysing network traffic using NB and DT algorithms. In addition, Entropy based feature selection is utilized for feature selection. We will address some topics in the following sections: on section 2 we present the related work, on section 3 data pre-processing of datasets is explained, in section 4 feature selection algorithm is discussed, in section 5 the two-level proposed NIDS are discussed in

details, section 6 evaluation performance is presented, in section 7 the experiments results and finally the conclusion.

2. Related Work

A Survey is done consisting latest papers which execute training and testing based on decision tree and self-organizing map.

Aggarwal P. and Sharma S.K., provides several classification algorithms like Random Forest, Naïve Bayes, C4.5, and Decision Table. They compared these classification algorithms in WEKA with KDD99 dataset. These classifiers were resolved according to metrics like accuracy, precision, and F-score. Random Tree displays the best outcomes aggregate in contrast the algorithms which have high detection and low false alarm rate were C4.5 and Random Forest [9]. Kumar S. and Jain S., **,** provided an effective decision tree rules for intrusion detection system by using ID3 decision tree classification for constructing intrusions rules so these rules is decided whether the network traffic behavior is normal or abnormal. Information gain and entropy is applied as feature selection, the experiment of the proposal show that the ID3 algorithm evaluation gives less false alarm (false positive and false negative) and high accurate rate [10]**.** Mukherjeea S. et al. suggested Feature Vitality Based Reduction Method to recognize important decreased input features for building an efficient and effective ID. To examine the performance of selection feature methods by utilizing three processes based Feature Selection (Gain Ratio and Information Gain and Correlation); NB classifier will be applied on NSL KDD dataset for intrusion detection system. The outcomes show that chosen little Features give better performance to build effective NIDS [11]. Mukund Y.R et al., proposed the present mechanism for intrusion detection system to inform afflicted way of employing the HDFS (Hadoop Distributed File System) of machine learning algorithms, so to minimize the rate of false alarm, they were used decision tree technique and augment it in the operation with the multi-device capacity of the HDFS, therefore this approach was reduced the time taken by the DFS and improved the accuracy of the IDS [12].

3. Dataset Preprocessing

KDDCup99 intrusion detection dataset includes 41 features of symbolic and numeric values. To train the algorithm, the standardization and normalization operations of data is necessary. The following step shows the preprocessing operation:

Standardization: Transform the symbolic value of feature to numeric value from [1….N], such as three types of protocols (tcp, udp, icmp), 68 types of service, and 11 types of flag in KDD cup 99 dataset. See Table 1.

Normalization: is applied on the continuous features through use Min.Max algorithm, the normalization process improves effectiveness and implementation of the system by creating the values of feature in range [0 to 1]. Therefore normalization can be avoided the bias problem that can be caused by larger features values.

4. Features Selection Methods

To improve the effectiveness of the system must be used feature selection technique for recognizing the irrelevant and redundant feature and removing them as much as possible. Feature selection techniques such as information gain, relief, gain ratio and the proposed system will be used entropy as feature selection.

5. The proposed Approach

The proposed of NIDS consists of Two-levels; in the first level the proposed system training with naïve bayes to classify the type of attacks from the normal behaviour in traditional network. In the second level the proposed system training with Decision Tree (DT) algorithm for building ID3 classifier. The two levels of proposed system are explained as follows:

Naïve Bayes classifier

NB classifier is a probabilistic classifier based on Bayes theorem with independence assumption. It is a supervised learning and will be trained very efficiently; it can be defined as equation [13]:

$$c(X) = \arg\max_{c \in C} \; P(c) * P(X|c) \qquad (1)$$

Where:

P(Ci | X) is the probability of feature to fall in class Ci

P (Ci) = the prior probability for class Ci.

C(X) is maximum posteriori used to assign the class c having maximum

p (X|c).

The class conditional independence is explained as this equation:

$$P(X|Ci) = P(a_1, a_2, \ldots\ldots, a_n|c) = \prod_{j=1}^{n} P(a_j|ci) \qquad (2)$$

The by simplicity of calculating P(C) and$P(a_i|c)$, NB classifier was easily builded.

Table 1: Convert symbolic value into numeric value of KDD cup99 dataset

Protocol type	Feature value	Service	Feature value	Service	Feature value	Service	Feature value	Flag	Feature value
Tcp	1	Private	1	time	23	shell	45	SF	1
Udp	2	Smtp	2	mtp	24	Efs	46	SH	2
icmp	3	http	3	gopher	25	login	47	S0	3
		ftp_data	4	rje	26	printer	48	S1	4
		IRC	5	link	27	netbios_ssn	49	S2	5
		telnet	6	Ctf	28	csnet_ns	50	S3	6
		Domain	7	Hostnames	29	nntp	51	RSTR	7
		Finger	8	iso_tsap	30	supdup	52	REJ	8
		Other	9	pop_2	31	http_443	53	RSTO	9
		ftp	10	netbios_dgm	32	uucp_path	54	RSTOSO	10
		Imap4	11	netbios_ns	33	domain_u	55	OTH	11
		pop_3	12	sql_net	34	ntp_u	56		
		Sunrpc	13	bgp	35	ecr_i	57		
		pm_dump	14	vmnet	36	eco_i	58		
		Echo	15	Z39_50	37	tim_i	59		
		Discard	16	ldap	38	urp_i	60		
		Systat	17	nnsp	39	red_i	61		
		Daytime	18	kshell	40	Remote_job	62		
		Netstat	19	klogin	41	X11	63		
		Ssh	20	uucp	42	http_8001	64		
		Name	21	courier	43	urh_i	65		
		whois	22	exec	44				

A Decision tree classifier

A Decision Tree (DT) is defined as a predictive modeling technique from the subfield of machine learning within the large field of artificial intelligence. DT is a recursive partition of the instance space which is constructed a simple tree like structure for expressing classification rules. One of the most popular DT algorithms is ID3 which utilized entropy of shannon's like a criterion for choosing the extreme important feature as shown in equation (3) [14]:

$$Entropy(s) = \sum_{i=1}^{c} - \ pi \, Log_2 \ pi \qquad (3)$$

Where:

S is the Pi is the probability of class C_i in S; c is the type of classes.

The suspicion in every node was minimized by selecting the attribute that most decreases its entropy. For realize this outcome; Information gain (Info gain) that degrees predictable reduction wihin entropy occasion by learning amount of a feature F_j, as shown in equation (3):

$$\boldsymbol{info\ gain(S, Fj) = Entropy(s) - \sum_{Vi \in V_{Fj}} \frac{|S_{vi}|}{|S|} . Entropy(S_{vi})} \qquad (4)$$

Where:

(V_{Fj}) was represented of whole potential amounts of feature (Fj)and (S_{vi}) is subset of (S) for which feature (Fj) has value (Vi). Algorithm (1) shows the Two-level proposed system of NIDS and see

figure (1) the general structure of the proposed system.

Algorithm (1) Two-level proposed system of NIDS
Input: Training and Testing Dataset
Output: classify the abnormal traffic into four types of attack and classify these attacks into their subclasses

Begin
Step1:**Preprocesing dataset**
1) Standardize the symbolic value of feature in both training and testing datasets
2) Normalize the continuous features in training and testing datasets

Step2: K-fold cross validation
select the value of K- fold (k=3) to divide the value of dataset into k equal parts (folds) approximately.

Step3: Feature selection
- Apply entropy in training dataset and select the best 30, 15 features using Eq. (3).

Step4: The first level of the proposed system: Naive bayes classifier
In training phase do the following:
1) calculate the probability of all class in training dataset when the p(c) is:
Pro. of Cj =(no. of each class) / (no. of total classes)
2) calculate the probability of every value within classes for all features when the p(Vj) is:
Pro. of Vj =(no. of Vj) / (no. of class)
In test phase do the following:
1)For every record in testing dataset calculate the probability of each value in training data with class when the p(Vj) is :
Pro. of Vj =(no. of Vj) / (no. of class)
2)Multiply the probability of every value in the record as Eq.(2)
$P(X|Ci) = P(a_1, a_2, a_3 \ldots, a_n|c) = \prod_{j=1}^{n} P(a_j|ci)$
3) Use the multiplication result of point 2 to multiply by the probability of class
4) Select the maximum value result from point 3 to classify the record as in Eq.(1)
Step5: Apply NB classifier with all feature
Apply NB classifier with best 30 features based on entropy in KDD cup 99
Apply NB classifier with best 15 features based on entropy in KDD cup 99

Step6: The second level of the proposed system:ID3 classifierIn training phase do the following:
For every class c in dataset training
- Calculate the p(c) from training dataset
- Compute the entropy of all training dataset utilizing Eq.3

End for
For every feature F in dataset training
- Compute the entropy

$$\boldsymbol{Entropy(s)} = \sum_{i=1}^{c} - \boldsymbol{pi\, Log_2\ pi}$$

- Compute the Info gain using Eq.4

$$\boldsymbol{info\ gain(S,Fj)} = \boldsymbol{Entropy(s)} - \sum_{\mathbf{Vi} \in \boldsymbol{V_{Fj}}} \frac{|\boldsymbol{S_{vi}}|}{|\boldsymbol{S}|} . \boldsymbol{Entropy(S_{vi})}$$

- Find the highest info gain

Repeat until all entry values are empty.
End for
In test phase do the following:
For every record in testing data:
1-Max=0, Class=““
2-For every Rule in training rule do steps 3,4
3-calculate Match that is a number of Rule conditions which is matched by record
3-If Match> Max
Then Max=Match, Class=class label of Rule
4-class of record is allocate by class label of Rule
End for
End for
End.

6. Performance Measures

The measure efficiency of IDS relies on four outcomes are [15]:
1. **True positive (TP):** Number of attack correctly identified attack event.
2. **True negative (TN):** Number of normal classified correctly normal event.
3. **False positive (FP):** normal Number incorrectly identified attack event.
4. **False negative (FN):** Number of attack incorrectly recognized normal event, when a detector unsuccessful to detect the attack because the virus is new and no signature is yet available. To correct the performance of the suggested system four possible results can be gained and confusion matrix displayed in Table 2.
5.

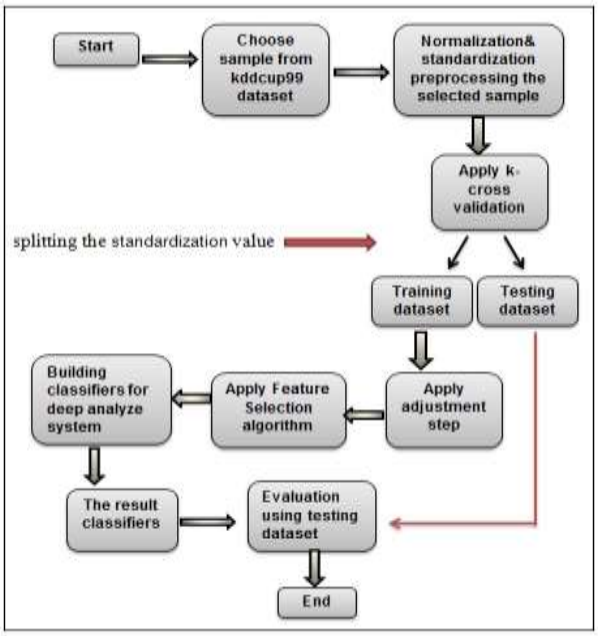

Figure 1: the general structure of the proposed system

Table 2: confusion matrix

Actual class	Predicted Class	
	Negative class(normal)	Positive class(attack)
normal	True negative (TN)	False positive (FP)
attack	False negative (FN)	True positive (TP)

(FP), (FN) should be minimizing to increase the efficiency and accuracy of IDS system. by utilizing the evaluation metric, The performance of the proposed system tested as follows [16]:
(ACC) Accuracy: measures performance by display the rate of samples which are truly detected as normal or attack to the overall number of samples as using the equation:

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

(DR) Detection Rate: measures performance, which mentions of the number samples that are correctly detected as attack to the overall number of attack samples as using equation:

$$DR = \frac{TP}{TP+FN} \quad (6)$$

(FAR) False alarm rate: measures performance, which mentions to the rate of samples that are incorrectly classified as attack to the total number of normal behaviour as using equation:

$$FAR = \frac{FP}{TN+FP} \quad (7)$$

7. Experimental Results

The proposed system used (4000) records of instances for training phase and test with (2000) records which are selected from KDD 99 dataset includes normal behaviour samples besides the other subclass types of attack. To evaluate the proposal system used three evaluation criteria which are (ACC, DR, FAR). To check the efficiency of the proposed will used k-fold cross validation which defined as follows: the data is equally partitioned number of times into k=3 equal size of segments or folds. In each k iteration, using holdout method to split the dataset of (6000) record into two parts:
one part of the data is held-out for testing (35% testing data) of (2000) records while the remaining k-1 parts are utilized for learning (65% training data) of (4000) record. In the first level for each k=1, k=2, k=3 the NB algorithm is tested with (2000) dataset records contain normal behaviour in addition to four attack types. In the second level for each k=1, k=2, k=3 the ID3 algorithm is tested with (2000) dataset records contain normal behaviour and the subclass of four type attack. The time for building model in the first level is (0.35) second and for testing model is (0.13) second while the time for building model in the second level is (0.44) second and for testing model is (0.23) second. The result of the two levels display classifier performance in each k iteration after best feature selection as shown in Tables 3 and 4 respectively.

Table 3: The results of first level using naïve bayes

Attack types	Dataset testing with k-fold	Number of feature	DR	Accuracy TP	Accuracy FP
Normal	K=1	41	0.99	0.98	5
		30	100	0.98	5
		15	0.80	0.90	0.1
	K=2	41	0.85	0.97	0
		30	0.99	0.98	0.5
		15	0.80	0.90	0.4
	K=3	41	0.84	0.98	0
		30	100	0.98	0
		15	0.80	0.91	0.1
Dos	K=1	41	0.99	0.97	5
		30	100	0.98	0.2
		15	0.99	0.99	0.6
	K=2	41	0.99	0.99	0.1
		30	100	0.97	0.1
		15	100	0.99	0.007
	K=3	41	0.99	0.99	0.1
		30	100	0.99	0.01
		15	0.99	0.98	0.003
Probe	K=1	41	0.97	0.98	0.1
		30	0.99	0.98	0
		15	0.85	0.97	0
	K=2	41	0.91	0.98	0
		30	100	0.96	0.1
		15	0.85	0.97	0.07
	K=3	41	0.91	0.96	0
		30	100	0.98	0
		15	0.85	0.96	0.1
U2r	K=1	41	0.77	0.95	0.001
		30	0.99	0.98	0
		15	0.70	0.97	0.2
	K=2	41	0.93	0.90	0
		30	0.88	0.94	0.2
		15	0.70	0.97	0
	K=3	41	0.91	0.91	0
		30	100	0.95	0
		15	0.70	0.97	0.2
R2l	K=1	41	0.95	0.98	0.003
		30	100	0.98	0
		15	0.75	0.97	0.2
	K=2	41	0.90	0.96	0
		30	0.99	0.97	0.6
		15	0.75	0.97	0.1
	K=3	41	0.92	0.98	0
		30	100	0.98	0
		15	0.75	0.97	0.1

Table 4: The results of second level using ID3

Attack types	Dataset testing with k-fold	Number of feature	Accuracy	FAR
Normal	K=1	41	0.98	0.02
		30	0.90	0.02
		15	0.80	0.001
	K=2	41	0.99	0.02
		30	0.90	0.02
		15	0.80	0.001
	K=3	41	0.99	0.01
		30	0.90	0.001
		15	0.80	0.001
Back	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Buffer_ overflow	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Ftp_write	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Guess_ passwd	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Imap	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Ipsweep	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0

Land	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Loadmodule	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Multihop	K=1	41	100	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Neptune	K=1	41	0.99	0.005
		30	0.76	0.02
		15	0.76	0.02
	K=2	41	0.99	0.005
		30	0.76	0.02
		15	0.76	0.02
	K=3	41	0.99	0.005
		30	0.99	0.02
		15	0.76	0.02
Nmap	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Perl	K=1	41	0.99	0
		30	100	0
		15	100	0
	K=2	41	0.99	0
		30	0.99	0
		15	100	0
	K=3	41	100	0
		30	0.99	0
		15	100	0
Phf	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Pod	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Portsweep	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0

	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Rootkit	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Satan	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Smurf	K=1	41	0.99	0.0001
		30	0.70	0.8
		15	0.70	0.8
	K=2	41	0.99	0.0001
		30	0.70	0.8
		15	0.70	0.8
	K=3	41	0.99	0.0001
		30	0.99	0.8
		15	0.70	0.8
Spy	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	100	0
		15	0.99	0
Teardrop	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Warezclient	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0
Warezmaster	K=1	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=2	41	0.99	0
		30	0.99	0
		15	0.99	0
	K=3	41	0.99	0
		30	0.99	0
		15	0.99	0

The result show that the performance of two levels of NIDS with 41 best features gave the best result in all terms of evaluation especially in the second level, and the performance of two levels of NIDS gave the second level best result in Acc and FAR. Finally, the two classifiers achieved the efficiency of the proposed system.

Conclusion

In this work two-level system proposed to classify the type of intrusion and detecting their subclasses of intrusion. The proposed work in the first level can classify the class of four attacks (dos, probe, U2R, R2L) with high detection average (99.6%) and little false positive average (0.01%) from three types of cross validate,. In the second level can detect the subclass of four attacks (Portsweep, Rootkit, Neptune, Teardrop, Phf, Spy, Ipsweep, Perl, Warezclient, Pod, Land, Satan, Back, Guess_passwd, ftp_write, Buffer_overflow, Nmap, Multihop, Smurf, Imap, Loadmodule, Warezmaster) with high detection average (99.9%) and little false positive average (0 %) from three types of cross validate.

References

[1] K. Kumar, R. Jha and S. Afroz, "Data Mining Techniques for Intrusion Detection: A Review," International Journal of advanced research in computer and communication engineering, ISSN (online): 2278-1021, ISSN (print): 2319-5940, Vol.3, Issue, 2014.

[2] S. Rajakshmi and J.S. Shanthini, "Data Mining Techniques For Efficient Intrusion Detection System: A Survey," International Journal on engineering technology and sciences- IJETS, ISSN (p):2349-3968, ISSN (0): 2349-3976, vol II, Issue XI, 2015.

[3] I. A. Abdulminem and S. H. Hashim, "A Proposal to Detect Computer Worms (Malicious Codes) Using Data Mining Classification algorithms," Engineering and Technology Journal, Vol 31, No 2, 2013.

[4] S. Singh and M. Bansal, "Improvement of Intrusion Detection System in Data Mining Using Neural Network," International journal of Advanced Research in Computer Science and software Engineering, ISSN: 2277 128X, Vol. 3, Issue 9, 2013.

[5] J. Novakovic, P. Strbac and D. Bulatovic, "Toward Optimal Feature Selection Using Ranking Methods and Classifications Algorithms," Yugoslav Journal of Operations Research, Doi: 10.2298/YJoR1119N, No 1, pp 119-135, 2011.

[6] F. Dewan, L. Zhang, C.M. Rahman, M.A. Hossain and R. Strachan, "Hybrid Decision Tree and Naïve Bayes Classification for Multi-Class Classification Tasks," Expert System with Applications 41(2014) 1937-1946, Journal homepage: WWW.Elsevier.com/Locate/Eswa, 2014.

[7] S. Kumar and S. Jain, "Intrusion Detection and Classification using Improved ID3 Algorithm of Data Mining," International Journal of Advanced Research in Computer Engineering and Technology, Vol. 1, Issue 5, 2012.

[8]. M. J Kelain and E.K. Jabbar, "Classification of Images Using Decision Tree," Engineering and Technology journal, Vol 31. Part (B), No 6, 2013.

[9] P. Aggarwal and S.K. Sharma, "An empirical comparison of classifiers to analyze intrusion detection," Advanced computing & communication technologies (ACCT), 2015 fifth International conference on 2015, eISSN: 2327-0659, DOI:10.1109/ACCT.2015.59, 2015.

[10] S. Kumar and S. Jain, "Intrusion Detection and Classification using Improved ID3 Algorithm of Data Mining," International Journal of Advanced Research in Computer Engineering and Technology, vol. 1, Issue 5, 2012.

[11] S. Mukherjeea, N. Sharmaa, "Intrusion Detection using Naive Bayes Classifier with Feature Reduction," Elsevier Ltd, 2012.

[12] Y.R. Mukund, S.S. Nayak and K. Chandrasekaran, "Improving False Alarm Rate in Intrusion Detection Systems using Hadoop," Inti. Conference on advance in Computing, communications and Informatics (ICACCI), India, Jaipur, sept.21-24, 2016.

[13]. V.Barot, S.S. Chauhan and B.Patel, "Feature Selection for Modeling Intrusion Detection", I.J. Computer Network and Information Security, pp.56-62, DOI: 10.5815/ijcnis.2014.07.08, 2014.

[14] A.M.A. Brifcani and A.S. Issa, "Intrusion Detection and Attack Classifier based on Three Techniques: A Comparative Study," Journal of Engineering and Technology, Vol.29, No.2, 2011.

[15] X.W. Shelly, B. Wolfgang, "The use of computational intelligence in intrusion detection systems, A review," Elsevier, Applied Soft Computing 10, Vol. 10, pp. 1–35, 2010.

[16] M.A**.** Hogo, "Temporal Analysis of Intrusion Detection", IEEE, International Carnahan Conference on Security Technology (ICCST), in Rome, Italy, **ISSN:** 2153-0742, 13-16, 2014**.**