Effective Intrusion Detection System using Combination of Data Mining Techniques

Kailas S. Elekar
M.E. IT (II[nd]),
Department of Computer Engineering
Dattakala Faculty of Engineering,
Swami Chincholi, Daund,
Pune, India
ekailas@gmail.com

Prof. M.M. Waghmare,
Assistant Proffesor
Department of Computer Engineering
Dattakala Faculty of Engineering,
Swami Chincholi, Daund,
Pune, India
monawaghmare25@gmail.com

***Abstract*— Security is a major issue in computer system and computer network since their introduction. Intrusion detection is the process of analyzing network traffic data for identification spiteful actions. These action will compromise the security of a system. In last decades various research and efforts are made to build effective intrusion detection system (IDS) but still there is a vast gap between the competence of IDS and attackers. Since the manual development process of detection model is very slow and expensive it is necessary to automate construction of it, using cost-sensitive machine learning algorithms. Data mining algorithms like Hoeffding tree, J48, Random Forests, Random Tree, and REPTree are used to build intrusion detection models using NSL-KDD train dataset. The performance of these models are evaluated using NSL-KDD test dataset, by conducting series of experiments and measured using correct attack detection ratio. The combination of data mining algorithm will increase accuracy of attack detection i.e. false positive and false negative.**

Keywords- Hoeffding tree, J48, Random Forests, Random Tree, and REPTree, IDS, Data Mining, NSL-KDD

I. INTRODUCTION

Intrusion incidents to computer systems are increased due to easy availability of internet and networks. The main purpose of any attack is to compromise system security and perform unauthorized vicious actions. These actions could be anything like obtaining access to files or data, destruction of the system or user files. An intrusion detection system (IDS) watches network devices and searches for anomalous or vicious behaviors in the network traffic. It has capability to distinguish between standard and vicious user behaviors.

Intrusion detection has emerged as an important field of research, because it is not possible to build a system without vulnerabilities. Main goal of any intrusion detection system is to find out the hidden attacks from a network traffic.

Data mining (also known as Knowledge Discovery in Databases - KDD) generally refers to the process of deriving conclusive models from huge volume of data. Frawley[1] defined data mining as "The nontrivial extraction of implicit, previously unknown, and potentially useful information from data".

The main goal of the data mining is to excerpt knowledge from a large amount of data and convert it into coherent form for further use [2]. Data mining-based IDSs require less expert knowledge (only need to label the traffic data to indicate intrusions instead of hand-coding rules) yet provide good performance.

Classification is the most generally used data mining task. The main aim of classification is to construct model by analyzing training data (known class) and predict unknown class. The constructed model can be presented in different forms such as decision trees, if then rules [2]. Several data mining algorithms such as J48, Random Forest, Random Tree, and more have been extensively employed for intrusion detection. These algorithms have ability to process large quantity of complex and dynamic data sets and generate rules for differentiating normal and strange actions. Past research shows that data mining algorithms can detect known and unknown attacks. There for many researchers are using data mining as main tool for building intrusion detection systems.

One of major problem with many data mining algorithm is that they can't detect all types of attack with acceptable detection ratio. To solve this problem we use the combination of data mining algorithms to improve attack detection ratio.

Rest of paper is organized as follows, Section II contains literature survey, proposed system is described in section III, section IV contains description of data mining and data set used, experiment details and results are given in section V, analysis of result is done in Section VI, Section VII contains conclusion and future work.

II. LITERATURE SURVEY

Intrusion is a set of actions which are intended to compromise the security of computer system in terms of confidentiality, integrity and availability [3].

There are two types of intruders i.e. external and internal [4]. External intruders are unauthorized users of the system, whereas internal intruders are authorized users of the system, but do not have rights to access other user's data. There are various classes of intrusions or attacks like Virus, Worm, Trojan, Denial of service (DoS), User to Root (U2R), Remote to Local (R2L), Probe, Password and Network attack in computer systems [5].

Network intrusion detection has been studied since last so many years. Generally, an intruder's actions are possibly different from that of legitimate user and hence can be categorized [6]. IDS are categorized based on their set up such as Host-based IDS (HIDS) which monitors and

analyzes the internal activity of a computer system [7]. HIDS can detect internal activity which attempts unauthorized access. For example text editor application starts modifying the password file. Network-based IDS (NIDS) which analyze network traffic and detects intrusions. Intrusion typically occur as anomalous patterns through various techniques [7]. The NIDS analyze incoming traffic, for identifying doubtful patterns. For example, port scan.

IDS can be broadly divided into three major categories based on their approach [8][9]. Misuse-based approach is based on signatures of known attacks. While Anomaly-based approach is based on normal action profile [9]. Any deviation to normal action profile treated as intrusion. Hybrid approach is the combination of Misuse-based and Anomaly-based approach.

Since last twenty five years KDDcup99 dataset is used for testing of IDS. This data set was build using the DARPA98 IDS evaluation program [10].

Elkan [11] compiled the results of the KDD'99 competition and they were mostly modified C5 decision tree algorithm (see Quinlan [12]). After the competition an extensive set of other algorithms were tested by many researcher and they found similar results [13][14]. These results are mainly based on ten percentage of training dataset only [14].

M. Tavallaee and others [15] after in depth analysis of the KDD dataset figured out 2 main problems in dataset. First one is very large number of unnecessary records in the dataset, and second one is immense number of similar records in train and test set. To solve these problems they have created NSL-KDD dataset.

Chakchai So and others [16] evaluated data mining algorithms such as Decision Tree, Ripper Rule, Neural Networks, Naïve Bayes, k–Nearest–Neighbour, and Support Vector Machine classifiers using KDD CUP dataset and HTTP BOTNET attacks.

Weiming Hu and others [17] proposed A new Online Adaboost Algorithm using Gaussian mixture models (GMMs) as weak classifiers. They further propose a distributed intrusion detection framework,

III. PROPOSED SYSTEM

A. *Proposed Architecture*

The proposed system architecture is shown in Figure 1 and it consists of following components.

Network Device:
A network device is a network interface which transmit and receive network traffic.

Attack Detection Engine:
It consist following sub components.
Packet Capture: Receives data from network device, analyzes and translate it into the instance form.
Feature Extraction: Extract features such as duration, protocol, service, etc from captured packet.
Data Mining Classifiers: Data mining classification models.
Anomaly Detector: Detects intrusions based on data mining algorithms as normal or anomaly. Once attack is detected system will send alert message.

Configuration: Store system configuration parameters such as protocol for packet capture, data mining algorithm for attack detection.

Database System: Store packet information, attack detected and message sent.

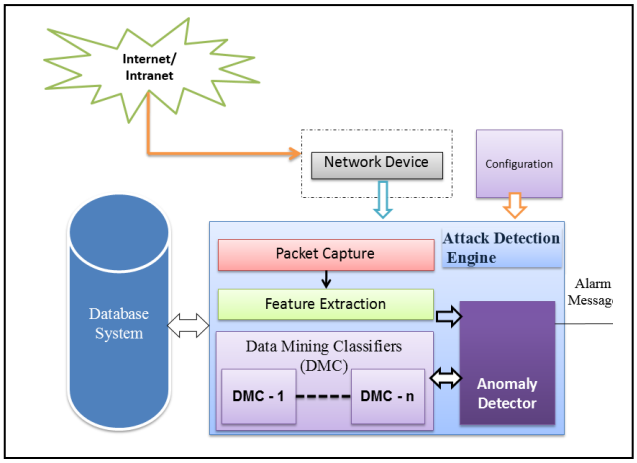

Figure 1 Proposed Architecture

B. *Mathematical Model*

Assumption: IDS always output (decision) corresponding to each input.

$$Z_i = C(R(D_i))$$

where
Z_i is the output of the IDS i.e. {Normal, DOS, PROBE, U2R, R2L}
D_i is network traffic (packet stream)
R is mapping/transition function, which maps the given data to a proper feature vector representation
C is a classification algorithm.

C. *Proposed Algorithm*

Input: Raw Network Packet, Detection Engine Parameters
Output: Attack Type
Initialization;
WHILE packetCapture DO
    read packet;
    process raw network packet and extract features;
    attackType = DetectAttack (packetFeature, DEParameters);
    IF attackType = NORMAL THEN
        Normal packet do nothing ;
    ELSE
        Attack packet;
        process response to attack (alert message);
    END IF
END DO

FUNCTION DetectAttack
Input: Packet Features, Parameters
Initialization
    Classify input packet using first data mining algorithm
    Classify input packet using second data mining algorithm

If results of both the classification are same then
return result
If result of any of the classification is attack type then
return attack type
END FUNCTION

IV. Descriptions of Data Mining Allogirtms, NSL-KDD Data Set and Weka

A. Data miningAlgorithms

For analyzing the network traffic and categorization of network attacks, five different tree base classification algorithms such as Hoeffding tree, J48, Random Forests, Random Tree, REPTree are used for evaluation in this research.

Hoeffding tree: It is a streaming decision tree induction algorithm based on Hoeffding bound. It is used for building decision tree from unbounded, high volume real time data feeds using limited system resources [18].

J48: Java implementation of C4.5 algorithm developed by Ross Quinlan [12]. It is used to generate pruned or unpruned decision tree [19].

Random Forests: Random forests is a collection of decision trees. Each tree is built separately with same distribution. It's predication is based on mode of the classification or mean regression of the individual trees [20].

Random Tree: It build a tree using K randomly selected attributes for every node. It does not perform pruning. It can estimate class probabilities using back fitting [19].

REPTree: It constructs decision tree using information gain. Missing values are handled by dividing the matching instances into fragments [19].

B. Data Set Description

In order to build and evaluate the performance of our system we use NSL-KDD data set available at http://nsl.cs.unb.ca/NSL-KDD/ web site. This data set was created by Tavallaee and others details can be found on the web site. After detailed analysis of KDD'99 data set they identified two major drawbacks in the data set. First drawback is large number of redundant records. Second drawback is large number of duplicate records in test set. To overcome these problem they created new data set called NSL–KDD [21]. The new data set will remove biasness towards more frequent records. This will also improve performance of machine learning algorithms which has better classification rates on the periodic records. The new data set contains good number of records in train as well as test sets. Hence there is no need of randomly selecting record for research [21].

This data set contains all features from original dataset. The attack in data set can be classified in to four groups as DOS, U2R, R2L, and PROBE [10].

DOS i.e. Denial of Service attack. This type of attack consumes computer resources due which the system will become overloaded. The overloaded system will not be able to provide services to genuine users.

U2R i.e. User to root attack. In this type of attack, attacker initially logged in to the system using normal user account. Then it gain access to super user or more privileged user account by exploiting weakness or bug in the system. Due to which attacker get permission to perform restricted functionality which are not meant for normal user.

R2L i.e. Remote to Local attack. In this type of attack, attacker gain access to remote machine by sending false network packets.

Probe attack. In this type of attack, attacker examine the system for knowing more details about system such as OS, open port etc.

The number of records for each category of attacks in train set as well as test of NSL-KDD data set is shown in Table I

TABLE I. No of Attacks in Dataset

Type	Training	Test
DoS	45927	7458
PROBE	11656	2421
R2L	995	2887
U2R	52	67
NORMAL	67343	9711
TOTAL	125973	22544

C. Waikato Environment for Knowledge Analysis(WEKA)

It is java library which contains complete set of tools for machine learning and data mining tasks. It was developed and maintained by Machine Learning Group at the University of Waikato, New Zealand. It provides nice GUI as well as command line interface for various data mining tasks such as pre-processing, classification, regression, clustering, association rules, and visualization. New machine learning algorithms can be smoothly integrated into it [18].

V. Experiments and Results

We have performed experiment using full NSL-KDD train dataset with125973 records and test data set with 22544 record

A. Experimental Setup

All our experiments are performed on following hardware and software.
Hardware: Intel Core i5, 2.8 GHz processor with 4 GB RAM and Network Interface
Software: Windows 7 64 bit, WEKA 3.7.11, Java (JDK 1.6), JPCAP, NetBeans IDE 7.2

B. Performance Measure

Performance of our approach was tested using two parameters i.e. attack detection rate and false attack detection rate. Attack detection rate is defined as total number attack detected using combination of data mining algorithms divided by total number of attacks in test data set. False attack detection rate is defined as total number of attack detected as normal or other attack category using combination of data mining algorithms divided by total number of category wise attack in test data set. Using these two parameters we have evaluated our approach. These two parameters tell us what percentage of intrusion is detected by our approach and how many incorrect classification it can make.

C. Experiment Results

The results achieved from our research work are summarized below. These results are obtained from confusion matrix tool provided by WEKA. The result shown in table II is taken from our previous work which is under publication [22] for comparison.

It shows overall classification accuracy in terms of correctly classified and wrongly classified record of test data set. Similarly Table III shows category wise percentage of attack detection.

TABLE II. PERFORMANCE METRICS

Classifiers	Classified Instances	
	Correctly	***Incorrectly***
J48	74.7028	25.2972
Random Forest (RF)	77.8921	22.1079
Random Tree (RT)	74.2814	25.7186
Hoeffding Tree (HT)	**79.0454**	**20.9546**
REPTree (RepT)	75.3504	24.6496

TABLE III. % OF ATTACK DETECTION USING SINGLE DATA MINING ALGORITHM

Classifiers	Attack Types					
	DOS	**R2L**	**PROBE**	**U2R**	**Normal**	**All**
J48	76.026	6.235	64.519	13.433	97.003	74.703
RF	82.153	7.101	73.276	4.478	**97.323**	77.892
RT	76.629	10.010	66.956	25.373	93.749	74.281
HT	81.335	**26.290**	**78.645**	34.328	93.379	**79.045**
RepT	**82.220**	10.703	69.021	**47.761**	91.062	75.350

None of single classifiers performs better in all types of attack category. So we performed another experiment by combining two classifiers and results obtained are summarized in Table IV.

TABLE IV. NO OF ATTACK DETECTED BY COMBINATION OF ALGORITHMS

Classifiers Combination	DOS		R2L	
	Correct	***False + ve***	***Correct***	***False + ve***
J48 & RF	6171	1287	252	2635
J48 & RT	5896	1562	362	2525
J48 & HT	6168	1290	768	2119
J48 & RepT	6194	1264	339	2548
RF & RT	6283	1175	347	2540
RF & HT	6172	1286	763	2124
RF & RepT	6185	1273	314	2573
RT & HT	6299	1159	805	2082
RT & RepT	6311	1147	402	2485
HT & RepT	6181	1277	772	2115

Classifiers Combination	PROBE		U2R	
	Correct	***False + ve***	***Correct***	***False + ve***
J48 & RF	1858	563	12	55
J48 & RT	1799	622	21	46
J48 & HT	2082	339	28	39
J48 & RepT	1836	585	34	33
RF & RT	1848	573	18	49
RF & HT	2149	272	23	44
RF & RepT	1865	556	33	34
RT & HT	2062	359	29	38
RT & RepT	1698	723	37	30
HT & RepT	2078	343	42	25

Classifiers Combination	NORMAL		TOTAL	
	Correct	***False + ve***	***Correct***	***False + ve***
J48 & RF	9455	256	17748	4796
J48 & RT	9462	249	17540	5004
J48 & HT	9457	254	18503	4041
J48 & RepT	9435	276	17838	4706
RF & RT	9472	239	17968	4576
RF & HT	9474	237	18581	3963
RF & RepT	9454	257	17851	4693
RT & HT	9231	480	18426	4118
RT & RepT	9113	598	17561	4983
HT & RepT	9123	588	18196	4348

VI. RESULT ANALYSIS

Percentage of correct attack and false attack detection using different combination of algorithms are presented in Table V and VI. From table V it is clear that the combination of Random Forest with Hoeffding Tree combination performs comparatively better than any other combination in PROBE and Normal attack category. Similarly Random Tree with Rep Tree, Random Tree with Hoeffding Tree and Hoeffding Tree with REP Tree combination performs comparatively better than any other combination in DOS, R2L, U2R attack category respectively.

TABLE V % OF CORRECTATTACK DETECTION USING TEST DATA SET

Classifiers Combination	Attack Types				
	DOS	**R2L**	**PROBE**	**U2R**	**Normal**
J48 & RF	82.743	8.729	76.745	17.910	97.364
J48 & RT	79.056	12.539	74.308	31.343	97.436
J48 & HT	82.703	26.602	85.998	41.791	97.384
J48 & RepT	83.052	11.742	75.836	50.746	97.158
RF & RT	84.245	12.019	76.332	26.866	97.539
RF & HT	82.757	26.429	**88.765**	34.328	**97.559**
RF & RepT	82.931	10.876	77.034	49.254	97.354
RT & HT	84.460	**27.884**	85.171	43.284	95.057
RT & RepT	**84.621**	13.924	70.136	55.224	93.842
HT & RepT	82.877	26.741	85.832	**62.687**	93.945

TABLE VI % OF FLASE ATTACK DETECTION USING TEST DATA SET

Classifiers Combination	Attack Types				
	DOS	**R2L**	**PROBE**	**U2R**	**Normal**
J48 & RF	17.257	91.271	23.255	82.090	2.636
J48 & RT	20.944	87.461	25.692	68.657	2.564
J48 & HT	17.297	73.398	14.002	58.209	2.616
J48 & RepT	16.948	88.258	24.164	49.254	2.842
RF & RT	15.755	87.981	23.668	73.134	2.461
RF & HT	17.243	73.571	**11.235**	65.672	**2.441**
RF & RepT	17.069	89.124	22.966	50.746	2.646
RT & HT	15.540	**72.116**	14.829	56.716	4.943
RT & RepT	**15.379**	86.076	29.864	44.776	6.158
HT & RepT	17.123	73.259	14.168	**37.313**	6.055

Table VII shows comparison of attack detection % using combination of algorithm with attack detection % of single best algorithm in each category of attack. The combination of algorithms achieves better attack detection ratio compared to single algorithm in all types of attack.

Table VIII shows percentage of improvements in attack detection using combination of algorithms over single best data mining algorithm in each attack category. The combination of Random Tree with Rep Tree, Random Tree with Hoeffding Tree, Random Forest with Hoeffding Tree, Hoeffding Tree with REP Tree and J48 with Hoeffding Tree performs comparativelybetter than any other combination in overall correct attack detection.

TABLE VII COMPARISON OF ATTACK DETECTION % USING COMBINATION OF ALGORITHM WITH ATTACK DETECTION % OF SINGLE BEST ALGORITHM IN EACH CATEGORY

Classifiers Combination	**Attack Types with *Best Single Algorithm & % of Detection***					
	DOS ***RepT 82.22***	**R2L** ***HT 26.29***	**PROBE** ***HT 78.64***	**U2R** ***RepT 47.76***	**Normal** ***RF 97.37***	**Overall** ***HT 79.04***
J48 & HT	**82.70**	26.60	**86.00**	41.79	**97.38**	**82.08**
RF & HT	**82.76**	**26.43**	**88.77**	34.33	**97.56**	**82.42**
RT & HT	**84.46**	**27.88**	**85.17**	43.28	95.06	**81.73**
RT & RepT	**84.62**	**13.92**	70.13	55.22	93.84	77.89
HT & RepT	**82.88**	**26.74**	**85.83**	**62.69**	93.95	**80.71**

TABLE VIII % OF IMPROVEMENTS IN ATTACK DETECTION USING COMBINATION OF ALGORITHM OVER SINGLE BEST ALGORITHM IN EACH CATEGORY

Classifiers Combination	**Attack Types**					
	DOS	**R2L**	**PROBE**	**U2R**	**Normal**	**Overall**
J48 & HT	0.483	0.312	7.352	-5.970	0.062	3.030
RF & HT	0.536	0.139	**10.120**	-13.433	**0.237**	**3.376**
RT & HT	2.239	**1.593**	6.526	-4.478	-2.265	2.688
RT & RepT	**2.400**	-12.366	-8.509	7.463	-3.481	-1.149
HT & RepT	0.657	0.450	7.187	**14.925**	-3.378	1.668

J48 and HoeffdingTree combination achieve improvements in DOS, R2L, PROBE and Normal categoryhowever it does not acvhive improvements in U2R category.Overall this combination performance was improved more than 3% in correct attack detection compared to single best algorithm.

Random Forest and Hoeffding Tree combinationachieves significant performance improvements i.e more than 10% in probe category and its performance was slightly improved in DOS,R2L and Normal category but does not improved in U2R category. Overall this combination performance was improved more than 3% in correct attack detection compared to single best algorithm.

Hoeffding Treeand REP Treecombination achieve improvements in DOS, R2L, PROBE and U2Rcategory.Althoughthis combinationachieves significant performance improvements i.e more than 14% in U2R category, it's performace was decreased in Normal category. Overall this combination performance was improvedmore than 1% in correct attack detection compared to single best algorithm.

Figure 2 and Figure 3 shows graphical representation of percentage of correct attack detection and percentage of false attack detection using combination of algorithms respectively.

Figure 4 shows graphical representation of percentage of improvements in attack detection using combination of algorithm over single best algorithm in each type of attack category.

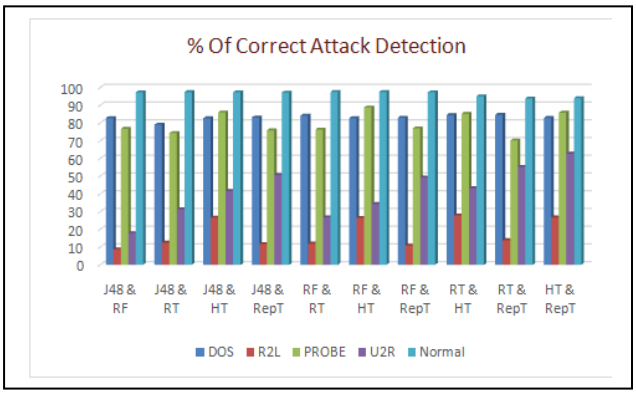

Figure 2 % of correct attack detection using combination of data mining algorithms

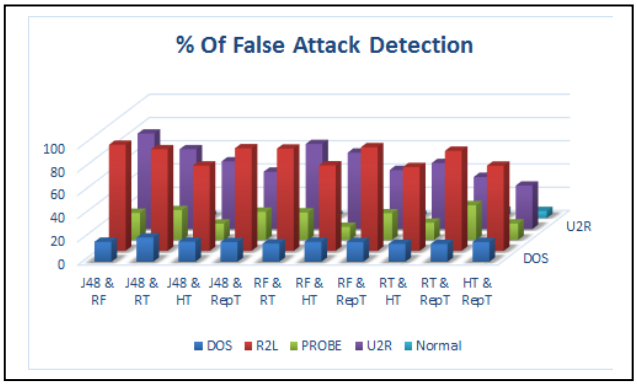

Figure 3 % of false attack detection using combination of data mining algorithms

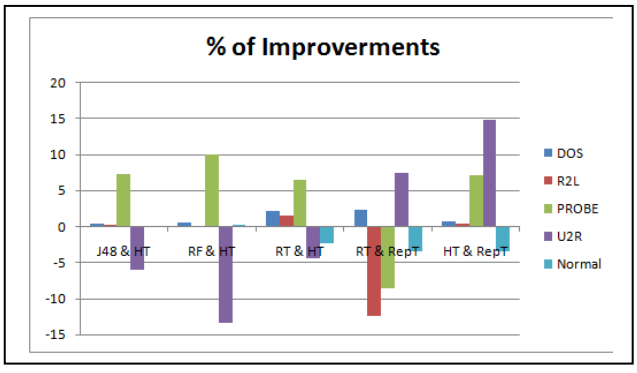

Figure 4 % of improvements in attack detection using combination of data mining algorithms.

Most of the combination achieve better performance than single algorithm because single algorithms can't perform better in all types of attack. None of combination performs better in R2L and U2R category because number of records in training set are very less compared to test data set.

VII. Conclusion and Future Work

This paper presents the concept of combination of data mining algorithms for improving attack detection rate and reducing false attack detection rate. We presented evaluation results by combining J48,Random Tree, Random Forest, Hoeffding Tree and REP Tree with each other. Obtained results are summarized in Table IV. Percentage of attack detection and false categorization are shown in Table V and VI. Comparison of our approach with best single algorithm in each attack category is presented in Table VII. Performance improvement of our approach over single algorithm is presented in table VIII. After analyzing table VII and VIII we conclude that combining Hoeffding Tree with J48 or Random Forest or REP Tree improves performance of intrusion detection for both the parameters. The performance was increased because Hoeffding tree performs better in PROBE and R2L type of attack and J48 and Random Forest performs better in Normal type and REP Tree performs better in U2R Category. Thus by combining advantages of each classifier we can achieve better attack detection rate and able to reduce false attack detection rate.

Despite the improvements in most of the category none of combination has achieved improvements in all the category.

In future we try our method i.e. combining two or more different types of data mining algorithm to achieve more attack detection rate.

Acknowledgment

We wish to express our sincere thanks to our Director, Head of Department, teachers and staff members of Computer Engineering Department at Dattakala Faculty of Engineering, Swami Chincholi, Daund. We are very grateful to the research community for their articles and papers due to which we have gained lot of knowledge.

We are thankful to National Informatics Centre, SDU, Pune for supporting this work.

Last but not the least, We would like to thank all our Friends and Family members who have always been there to support and helped us to complete this research work.

References

[1] W. J. Frawley, G. Piatetsky-Shapiro, and C. J. Matheus, “Knowledge Discovery in Databases: An Overview,” AI Magazine, 1992, pp. 213-228

[2] http://www.tutorialspoint.com/data_mining/index.htm

[3] R. Heady, G. Luger, A. Maccabe, and M. Servilla, “The architecture of a network level intrusion detection system,” pp. 1–18, August 15-1990.

[4] J. P. Anderson, “Computer security threat monitoring and surveillance,” April 1980.

[5] H. G. Kayacik, A. N. Z. Heywood, and M. I. Heywood, “Selecting features for intrusion detection a feature relevance analysis on kdd 99 intrusion detection datasets,” pp. 1–6, October 2005.

[6] P. Ning and S. Jajodia, Intrusion Detection Techniques. (Ed) The Internet Encyclopedia,2003.

[7] F. Wikimedia, Intrusion detection system, Feb 2009.

[8] Sundaram, “An introduction to intrusion detection,” Crossroads,vol. 2, no. 4, pp. 3–7, April 1996.

[9] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly Detection : ASurvey,” ACM Computing Surveys, vol. 41, no. 3, pp:1–15, September 2009.

[10] KDD Cup 1999 Data available at http://kdd.ics.uci.edu/databases/kddcup99/task.html, 1999.

[11] C. Elkan. "Results of the KDD'99 classiffer learning". http://cseweb.ucsd.edu/~elkan/kdresults.html

[12] J. R. Quinlan. "C4.5: programs for machine learning". Morgan Kaufmann Publishers Inc., 1993. ISBN 1-55860-238-0.

[13] M. Sabhnani and G. Serpen. "Application of machine learning algorithms to KDD intrusion detection dataset within misuse detection context". In International Conference on Machine Learning, Models, Technologies and Applications (MLMTA), pp. 209-215. CSREA Press, 2003.

[14] S. Sung, A.H. Mukkamala. "Identifying important features for intrusion detection using support vector machines and neural networks". In Proceedings of the Symposium on Applications and the Internet (SAINT), pp. 209-216. IEEE Computer Society, 2003.

[15] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” in Proc. 2nd IEEE International Conference on Computational Intelligence for Security and Defense Applications. USA: IEEE Press, 2009, pp. 53–58.

[16] S. Chakchai ,N. Mongkonchai, P.t Aimtongkham,K. Wijitsopon and K. Rujirakul, “An Evaluation of Data Mining Classification Models for Network Intrusion Detection,” IEEE, Digital Information and Communication Technology and it’s Applications , Fourth International Conference, May 2014, pp 90 - 94 .

[17] W. Hu, J. Gao, Y. Wang, O. Wu, and S. Maybank, “Online Adaboost-Based Parameterized Methods for Dynamic Distributed Network Intrusion Detection,” IEEE Transactions On Cybernetics, 2014,Vol. 44, pp. 66-82.

[18] P. Domingos and G. Hulten. "Mining high-speed data streams". In Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '00). ACM, New York, NY, USA, 2000, 71-80.

[19] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H,“The WEKA data mining software: an update”, ACM SIGKDD explorations newsletter, 11(1),2009 10-18.

[20] Breiman, Leo. "Random forests." Machine learning 45.1 (2001): 5-32.

[21] K. Elekar and M. Waghmare “Comparison Of Tree Base Data Mining Algorithms For Network Intrusion Detection” ICETRESM2015 Under Publication.

[22] The NSL-KDD Data Set http://nsl.cs.unb.ca/NSL-KDD/