Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/130218

Semi-Supervised Multi-Instance Neurologic Adaptive Learning Intrusion Detection System

A. P. Jagadeesan[1,*] *and K. Gnanambal* [2]

[1] Department of Computer Science and Engineering, R.V.S. College of Engineering, Dindigul, Tamil Nadu, India
[2] Department of Electrical and Electronics Engineering K.L.N. College of Engineering, Madurai, Tamilnadu, India

Received: 2 Sep. 2018, Revised: 2 Oct. 2018, Accepted: 8 Dec. 2018
Published online: 1 Mar. 2019

Abstract: In the earlier work, Single Instance Single-Label Learning mechanisms are proved faster and converge for better results but the accuracy is the concern in the Intrusion Detection systems (IDS) in large scale data centric networks. Evolution of supervised training mechanisms have favored more accurate decision making, however failed to incorporate the dynamism of new instances which are in early stages and non-overlapping with current training data sets. This paper proposes a cascaded and hybridized mechanism involving Semi-supervised Multi-Instance Neurologic Adaptive Learning (SMI-NAL) mechanism. The objective is to estimate the threshold level of convergence and improve the system accuracy. In addition, the proposed learning mechanism reduces the computational complexity to achieve proactive measure which is the real challenging phenomenon. Compared to the conventional IDS process, the proposed learning mechanism improves the accuracy of IDS and updates the machine learning set (Training set) more appropriately and wrong decision making can beeliminated. The key features of IDS such as robustness, scalability and ease of use are achieved using the proposed learning mechanism.

Keywords: Intrusion Detection System(IDS),Semi-supervised Multi-Instance Neurologic Adaptive learning (SMI-NAL), Machine learning set.

1 Introduction

The evolution of IDS begins with the introduction of networks to share and store data. The growth in the mechanisms of intrusion detection systems laterally depends on the advancement in the various network technologies. A new mechanism called Semi-supervised Multi-Instance Neurologic Adaptive Learning SMI-NAL that suits the recent networks has been developed by cascading the fast adaptive neural network classifier and multi instance multi-learning mechanism. It improves the performance of IDS in terms of computational time and accuracy of classification.

IDS can be defined as process of monitoring activities in system which can be computer or network system. In the conventional IDS process, the given IP address and port numbers are equally separated and stored in the respective bags of requests. For example, if the input of 100 requests is fed-in, by the above process it is splitted equally basis and stored in the $Good - ipbag$ and $Bad - ipbag$. There is a possibility of a bad IP address present in the Good-ipbag. On the other hand, by introducing suspect ratio, the system is able to avoid wrong decision made by the IDS. Outlier Detection approach was proposed to detect intrusions with big datasets which takes less execution time and less storage for dataset [1]. In particular several Neural Networks-based approaches were employed for intrusion detection. Intrusions disturbingintegrity, availability and confidentiality of cloud resources and services was analyzed and various types and methodologies of Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) in cloud were discussed. In addition, IDSIPS positioning in Cloud architecture to achieve preferred security in the subsequent generation networks is recommended [2]. After an extended review and complicated association, we propose the taxonomy to sketch modern IDS. In addition, the overall picture of IDS can be easily grasped using pictorial representations [3].A review related to the false positives problem and alerts load and their reduction techniques using data mining techniques is discussed. The research approach is

* Corresponding author e-mail: apj.tharun@gmail.com

classified into (a) the detection methodology that functions during detection phase (b) alert processing methods that are used to generate alerts after detection phase. In addition, many open issues have also been addressed for further exploration [4]. The multi-instance multi-learning framework described by multiple instances and associated with multiple class labels is more convenient and likely to represent complicated objects which have several semantic meanings. In addition, the MimlnBoost and MimlSvmalgorithms are proposed for learning MIML examples.Also, the instance differentiation and sub-concept discovery algorithms are proposed where the former works by transforming single instances into the MIML representation,while the latter works by converting single-label examples into the MIML representation for learning [5]. Statistical methods, machine learning and data mining techniques have been proposed achieving higher mechanization capabilities than signature-based methods; while taking into account many essential features. In addition, the automation helps to identify the potential causes at the rear end of the negligence of novel techniques by network security experts [6]. In addition to various factors like speed, cost, resource utilization, effectiveness etc., false alarms rate and accuracy of detection are found to be the most significant issues and challenges while designing effective IDS [7]. A neural network-based soft computing technique such as Self organizing map for identifying the intrusion in network intrusion detection is proposed. This methodology can be applied to detect terrorists and their supporters using legal ways of Internet access and also unseen attack [8]. Back-propagation neural network with all features of KDD data is employed and the classification rate for experiment result for normal traffic is 100%, known attacks are 80%, and unknown attacks are 60% [9]. The application of some neural networks (NNs) to identify and categorize intrusions is discussed and determination of which NN classifies well the attacks and produces the higher detection rate of each attack is performed. Resilient back propagation for detecting each type of attack along is suggested with the accurse detection rate of 95.93 [10]. Back propagation neural network with many types of learning algorithm is employed and the performance of the network is 95.0 [11].

Studies convey that SISL-single instance single-label learning mechanisms are faster and converge for better results but the percentage of false positive and false negative alarms are of concerns. Learning mechanisms are linear in nature in most of the proposed algorithms. Evolution of supervised training mechanisms have helped more accurate decision making however it failed to incorporate the dynamism of new instances which are in early stages and non-overlapping with current training data sets. Machine learning algorithms are proven to be highly convergent and give true normalized data sets. However, increase in internet or intranet traffic and cloud based or virtual infrastructure based applications demands even higher convergent and more accurate decision-making mechanisms to prevent multiple threats.

2 Proposed Work

2.1 Intrusion detection system based on SMI-NAL

1.Every user is uniquely identified by his/her IP address
2.User cannot be identified as an intruder with just one request
3.Every request from a user is monitored and intrusion coefficient of the user is updated
4.If the intrusion coefficient is above the certain threshold then the user is identified as an intruder

In this system, initially-detected IP address and port numbers is re-routed by the application of suspect ratio for a close and finer result of suspecting a bad IP address and bad port numbers. We make use of port numbers to identify the suspicious IP address. After finding a set of suspicious IP address (which is not under good and bad IP address bag) is given as input to the SMI-NAL classifier where parallel computation of all features takes place thus single packet is taken and labeled good or bad in every feature taken thus multiple labeling is taken place. If two or more packets contain the same suspicious in a particular feature then it is further processed to the already separated bad bags and update the database. Thus, IDS learning mechanism has a faster updates of known bad IP Port numbers and quickly makes an analysis of blocking suspicious IP address and Port numbers.

2.2 IDS Process

When a set of client request hits the system, the request is forwarded to IDS. IDS process considers the request and classifies it based on its process into two different categories,

1.Good IP- Good Port.
2.Bad IP-Bad port.

Training set gets created with initial set of good and bad IP and port information. Considering a scenario, where the training set is initial created using 100 IP address, in which the training bag $Good-ip-tbag$ contains a list of all Good IP address after IDS processing of around 50 IPs and $Bad-ip-tbag$ contains a list of all Bad IP addresses of around 50 IPs (where the list of IP address provided for creating the training set is known Good or Bad IP address). Consider, at a particular instant of time, the number of requests that hits the server is about 150, among which 100 requests occurs from already processed IP address that are retained in training set. When a request hits the server, it is forwarded to the

IDS process, which at first verifies the training set to confirm the good or bad training set which already has the IP address from which the request hits the server. In this case, since 100 among the 150 requests that hit the server are from the known IP address, the IDS processes based on the existence of the IP address within the training set (maintained in persistence storage area or in DB) i.e., either good (then it allows or authorizes that request to process further) or bad (then it denies such IP request from further processing against the server). Provided, after such validation, the IDS finds that there exists, 50 uncertain IP address that needs to be classified further, for which the IDS sends those IP address to $SMI-NAL$ for further processing.

3 Semi Supervised SMI-NAL Processing

The SMI-NAL process takes the uncertain IP address along with its port information, verifing against its multiple layer processing to classify the list of uncertain IP address into 2 different categories, i.e., either as

1.$UncertainIPGoodPort$
2.$UncertainIPBadPort$

There are multiple SMI-NAL process that gets triggered to derive multiple results, combination of which is considered for further classification to derive an efficient solution, thereby improves the accuracy and computational time of IDS. Here, uncertain IP address along with its Port information is processed through multiple SMI-NAL process in parallel classifying them into uncertain IP-good-port (adds the list of such IP address into uncertain-good-bag file or DB table) and uncertain IP-bad port (adds the list of such IP address into uncertain-bad-bag file or DB table) along with number of times the request hits the server.

3.1 Dynamic updation of training sets

Based on $SMI-NAL$ processing, there would be two categories of classification performed for uncertain IP considering its port information like the Table1 and Table 2 referred below. i.e., the $SMI-NAL$ processing derives results as uncertain $IP-good$ port or as uncertain $IP-bad$ port. Along with which the number of times the request hits the server is captured. The number of hits against an uncertain IP exceeds the threshold limit such as $k>=5$, then based on whether its a good or bad port, the training set gets updated by adding that uncertain IP with good port data into $Good-ip-tbag$ set and removing the data from $uncertain-good-bag$ and updating training set dynamically for $Bad-ip-tbag$ for uncertain IPbad port data and removing the same $uncertain-bad-bag$ settable. Dynamic and appropriate updates to training set results in improving the accuracy and reduced false alarm.

The output from the $SMI-NAL$ processing which consists of Greylist IPPort is given as an input to the fast adaptive Neural classifier which has the list of uncertain IPs and port information of each IP. The multi-instance multi-label where each trained set is associated with not only multiple instances but also multiple class labels. Multi-instance learning studies the number of instances from $SMI-NAL$ processing unit and describes one instance by number of labels. Multi-label learns from the output, where an instance can have multiple labels. By combining both, the classifier classifies the given instances as multiple labels (certain IP; good port; uncertain IP; bad port; certain IP bad port; and uncertain IP and bad port). This classifier classifies the given instances into multiple labels, and these labels are fed-in across parallel $SMI-NAL$ processors. When the given instance (Port) is verified as good port then, $SMI-NAL$ classifier creates a set named as certain IP bad port and adds or updates the data, where the data consists of IP/Port and counter value. The main objective of setting a counter value is to push the given set of samples under Whitelist IP/Port or black list IP/Port criteria. The counter maintains a threshold value i.e. k5where k is the number of instances which includes IP and Port numbers. When the instances touch the threshold limit (k5) i.e., more than 5 times port/IP hits, then the IDS will store the requests to Whitelist IP/Port training set and finally updates the machine learning set or (training set). By this way, Semi-supervised $SMI-NAL$ classifies the requests and machine learning set more appropriately. Thus, improvising the accuracy and the machine learning set is updated appropriately and the system is able to avoid taking wrong decision as taken by conventional IDS process (i.e., reducing false alarm).

3.2 Computation time

The time consumed by the IDS process to execute multiple requests is known as computation time. With single core processor, IDS handles the request based on FIFO basis. Computational time of IDS using single instance learning mechanism is analyzed across various sets of client requests. Table 3 displays computational time consumed where the measure of time for an IDS process to execute for set of client IPs(say 100 to 1000) is indicated and time in (seconds) is displayed.Figure 3 plots the overall time taken for an IDS process to execute the set of client IPs.

Table 4 indicates the measure of time for the $SMI-NAL$ logic to process to the number of hits (i.e., client IPs (say 100 to 1000) and time taken is displayed in seconds. Figure 4 also indicates processing time for the $SMI-NAL$ process to execute set of client IPs.

Table 1: Uncertain good bag

IP Address/Good Port Number 10.203.X.X/80	Number of Hits 1
10.203.X.X/23	4
10.203.X.X/45	3

Table 2: Uncertain good bag

IP Address/ Bad Port Number	Number of Hits
10.203.X.X/27	0
10.203.X.X/57	2
10.203.X.X/32	1

Table 3: Computation time for executing IDS process

Number of client IP's	Time (s)
100	0.133215
200	0.382325
300	0.474681
400	1.072448
500	1.142554
600	2.201219
700	1.944813
800	3.880215
900	4.609544
1000	4.041619

Table 4: Computation time taken by $SMI-NAL$ process

IP	Time (s)
100	0.180009
200	0.263926
300	0.679973
400	0.745594
500	1.622387
600	1.510177
700	2.810748
800	2.736981
900	3.170536
1000	3.877578

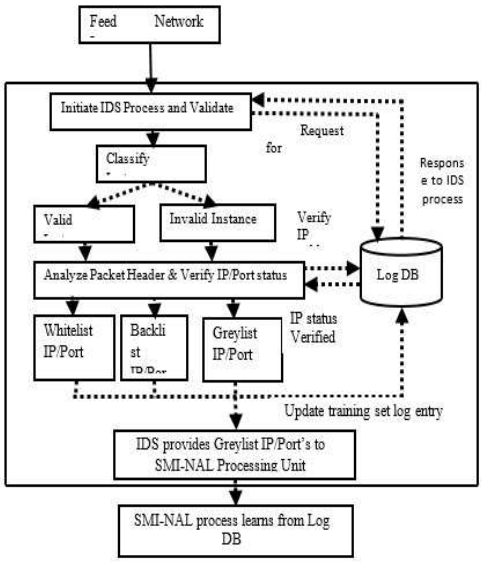

Fig. 1: The process flow diagram for IDS mechanism

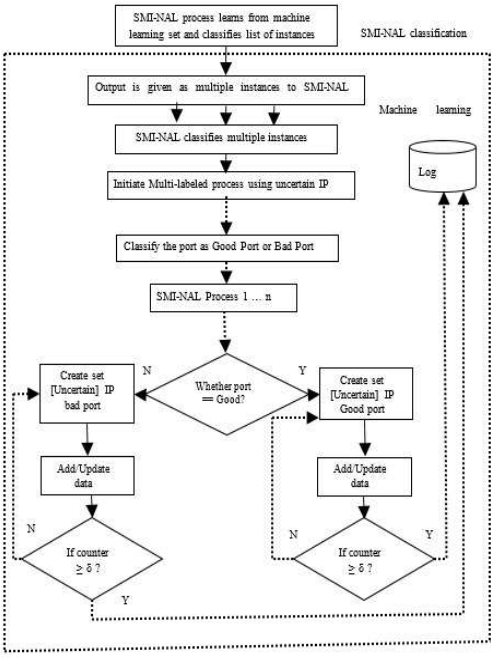

Fig. 2: The *SMI – NAL* classification

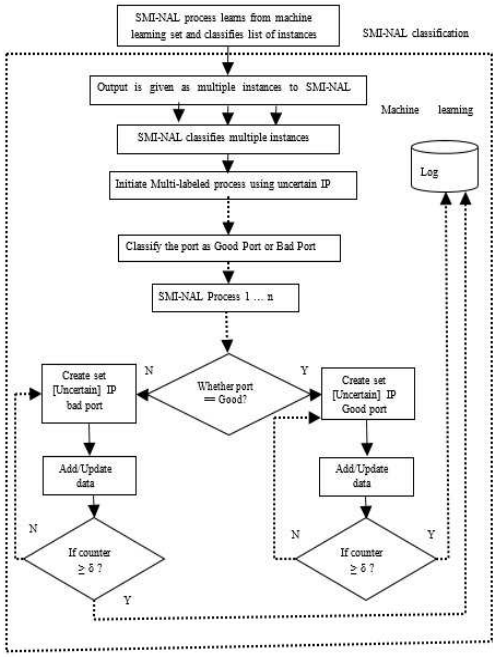

Fig. 3: Computation time

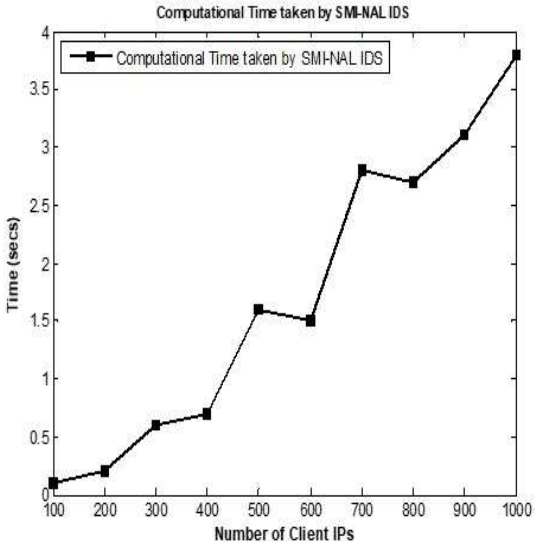

Fig. 4: Computational time taken by *SMI – NAL* IDS

3.3 Robustness

The higher the robustness (how many hits/requests the IDS process can handle), the greater is the stability of the IDS process. Experimental analysis is conducted using the *SMI – NAL* process to test the robustness of the system. Table 3 indicates the maximum number of requests against time which the IDS process is consumed to execute (i.e., client IPs (say 1000 to 20000). Figure 5 indicates that the IDS process is robust and stable to execute more number of hits with less time. In the

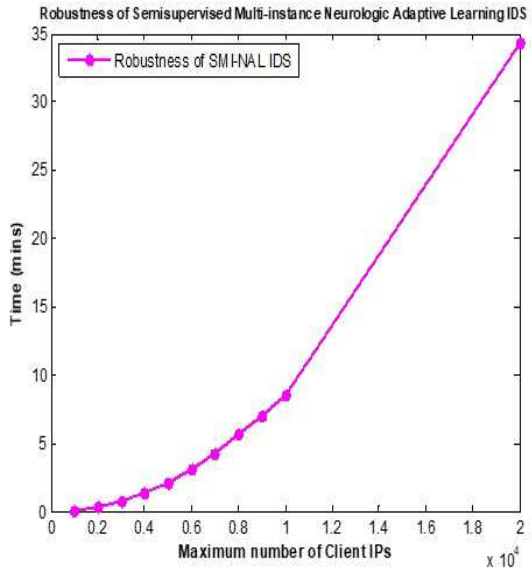

Fig. 5: Robustness of $SMI-NAL$ IDS

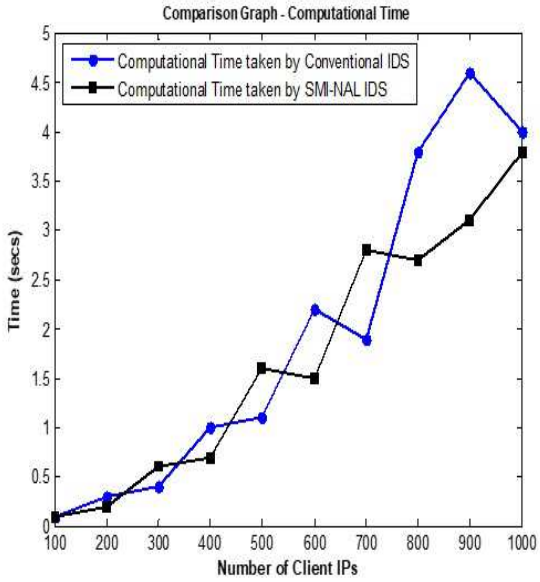

Fig. 6: Comparison for the computational time

highlighted column in Table 5, IDS mechanism can process a maximum of upto 20000 hits.

3.4 Fast-learning mechanism

Table 6 indicates the measure of time for an IDS process to execute for number of hits (i.e. client IPs (say 100 to 1000) and time taken for the $SMI-NAL$ process to execute the number of client IPs Vs time (seconds). Figure 6 also indicates the comparison between the time taken for IDS process as well as the time to perform $SMI-NAL$ process.

3.5 Effectiveness

Effectiveness of our proposed IDS scheme is evaluated using the following performance metrics:

Detection Rate (D-rate): the detection rate is defined as the percentage of attack attempts that are determined to be under attack.

Further, the ROC curves in Figure 7 shows attack detection rate for the proposed $SMI-NAL$ system is greater than 96% and close to 99%, The result shows that compared to some well-known methods the proposed system can effectively detect attacks.

Where N$(total-attempts)$ indicates a total number of attempts and $N-attacks$ indicates number of attacks detected. The $D-rate$ corresponds to the probability of detection $(P-detection$ when an attack is present and it

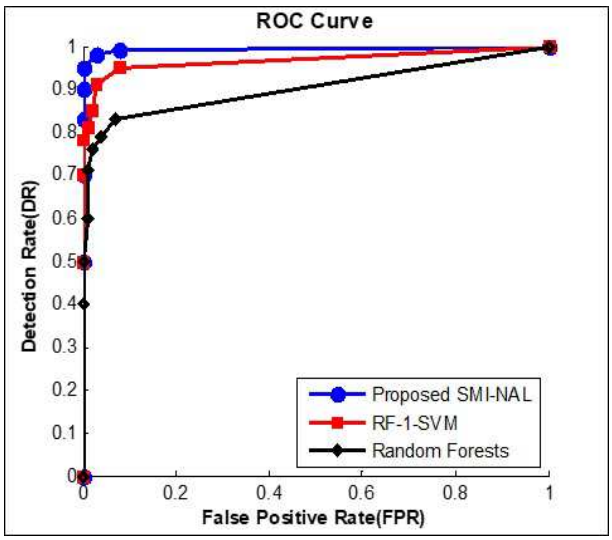

Fig. 7: ROC curve of the SMI NAL

corresponds to probability of declaring a false positive $(P-fp)$ under a normal $(non-attack)$ situation.

$$D_{rate} = \frac{N_{total-attempts}}{N_{attacks}} \quad (1)$$

Receiving Operating Characteristic (ROC) curve: To evaluate the performance of our proposed mechanism, we need to study the trade-off between the false positive rate in a null hypothesis (H-o) condition against the probability of detecting an attack under a situation when

Table 5: Robust-processing Time

Number of client IP's	Computation Time (Seconds)	Computation Time (Minutes)
1000	5.577182	0.0929
2000	21.401408	0.356
3000	47.237099	0.7872
4000	82.871065	1.381
5000	128.399645	2.139
6000	187.357531	3.122
7000	252.822781	4.2137
8000	339.724296	5.662
9000	419.255525	6.9875
10000	515.177651	8.5862
20000	2063.137341	34.385

Table 6: Robust - processing time

Number of client IPs	Time (s)	Number of client IPs	Time (s)
100	0.180009	100	0.133215
200	0.263926	200	0.382325
300	0.679973	300	0.474681
400	0.745594	400	1.072448
500	1.622387	500	1.142554
600	1.510177	600	2.201219
700	2.810748	700	1.944813
800	2.736981	800	3.880215
900	3.170536	900	4.609544
1000	3.877578	1000	4.041619

an attack is present. The ROC curve is usually used to measure the tradeoff between P_fp and P_detection.

4 Conclusion

The threefold classification of users in the proposed mechanism paves a way of avoiding static fixation of users either as a legitimate user or an intruder. It introduces dynamism in classifying the users based on the communicating parameters (IP address and port number). The usage and pattern of the requests are studied for a time period and the placing of the clients in a particular category is done. This dynamism differentiates the proposed work from other relevant and existing works. Learning on the go of the nature of the incoming requests enables us to achieve more efficiency in terms of time and accuracy. The work shows that this method of intrusion detection best adapts for the large scale data centric networks. This dynamic nature of the proposed work, in future, can be adapted to clustered kind of networks where there is no standard point of deployment for intrusion detection system is available.

Acknowledgement

We are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] J. Jabez and B. Muthukumar, Intrusion Detection System (IDS): Anomaly Detection using Outlier Detection Approach, Elsevier - Procedia Computer Science, Vol. 48, pp. 338 - 346, doi: 10.1016/j.procs.2015.04.191 (2015).

[2] ChiragModi, Dhiren Patel, BhaveshBorisaniya, Hiren Patel, Avi Patel and MuttukrishnanRajarajan, A survey of intrusion detection techniques in Cloud, Journal of Network and Computer Applications, Vol. 36, issue 1, pp. 42-57 (2013).

[3] Hung-Jen Liao, Chun-Hung Richard Lin, Ying-Chih Lin and Kuang-Yuan Tung, Intrusion detection system: A comprehensive review, Journal of Network and Computer Applications, Vol. 36, issue 1, pp. 16-24 (2013).

[4] AsiehMokarian,AhmadFaraahi and ArashGhorbanniaDelavar, False Positives Reduction Techniques in Intrusion Detection Systems A Review, International Journal of Computer Science and Network Security, Vol. 13, No.10, pp. 128 -134 (2013).

[5] Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang and Yu-Feng L., Multi-instance multi-label learning. Artificial Intelligence, Vol. 176, issue 1, pp. 2291-2320 doi:10.1016/j.artint.2011.10.002 (2012).

[6] Carlos A. Catania and Carlos GarcaGarino, Automatic network intrusion detection: Current techniques and open issues, Computers & Electrical Engineering, Vol. 38, issue 5, pp. 1062-1072, doi:10.1016/j.compeleceng.201205.013 (2012).

[7] F.N. Sabri, N.M. Norwawi, K. Seman, Identifying false alarm rates for intrusion detection system with Data Mining, IJCSNS International Journal of Computer Science and Network Security, Vol. 11, No. 4, pp. 95-99 (2011).

[8] RituRanjani Singh, Neetesh Gupta, Shiv Kumar, To Reduce the False Alarm in Intrusion Detection System using self Organizing Map, International Journal of Soft Computing and Engineering (IJSCE), Vol.1, issue 2, pp. 27-32 (2011).

[9] Adesina Simon Sodiya, OlusegunFolorunso, SaidatAdebukolaOnashoga and Omoniyi Paul Ogunderu, An Improved Semi-Global Alignment Algorithm for Masquerade Detection, International Journal of Network Security, Vol.13, No.1, pp. 31-40 (2011).

[10] I.Ahmad, S. Ullah Swati and S. Mohsin, Intrusions Detection Mechanism by Resilient Back Propagation (RPROP), European Journal of Scientific Research, Vol. 17, No. 4, pp. 523-531 (2007).

[11] S. Mukkamala, H. Andrew Sung and A. Abraham, Intrusion detection using an ensemble of intelligent paradigms, Journal of Network and Computer Applications, Vol. 28, pp.167-182 (2005).

A. P. Jagadeesan was born in Dindigul, Tamilnadu, India in 1980. He received B.E. Degree in Electrical and Electronics Engineering from Madurai Kamaraj University Madurai, India in 2001 and the M.E. degree in Computer Science and Engineering with distinction from Anna University, Chennai, India in 2007.Currently he is an Assistant Professor in the Department of Computer Science and Engineering in R.V.S College of Engineering, Anna University, Chennai, India and pursuing Ph.D. Degree at Anna University Chennai, India. His research interests are in the areas of soft computing and Network security. He is a Life member in Indian Society for Technical Education He was a recipient of BEST GUIDANCE OF PROPOSAL for various Project Proposals. He has authored or co-authored more than 10 National and International Journals and conference papers.

K. Gnanambal was born in Madurai, Tamilnadu, India in 1974. She received B.E. degree in Electrical and Electronics Engineering with distinction, M.E. degree in Power Systems with distinction from Thiagarajar College of Engineering, Madurai, India and Ph.D. degree in Electrical Engineering from Anna University,Chennai, India in 1996, 1997 and 2011 respectively. She has teaching experience of over twenty years in engineering field. She is currently a Professor in the Department of Electrical and Electronics Engineering, K.L.N College of Engineering, Anna University Chennai, India. Her area of research includes soft computing, Power System Analysis and Voltage Stability. She has authored or coauthored more than forty refereed National and International journals and conference papers. She is a Life member in Indian Society for Technical Education and member in Institution of Engineers, India.