Ensemble Model to Detect Wireless Attacks in Mobile Ad hoc Networks

N. Ravi, G. Ramachandran

Abstract— Recent development in internet connectivity based mobile technologies leads a tremendous growth on Mobile Ad hoc Networks (MANETs). Further, these networks are specialized and serving the need of high-speed internet connection to the general user and for industries. Furthermore, the raise in the technology rises along with the security issues and threats. It is noticed that the attack over these MANET are high and severity of the attack launched over the networks are increasing drastically. An existing security system, which completely relies on the conventional signature based approach fails in detecting these attacks. Hence, an efficient classification model is required. Further, to create a robust and efficient model, this paper proposes an ensemble model, which holds the group of classifier for classification. The ensemble model validates the input in each classifier and finally ranks the accuracy. The best accuracy among the classifiers is taking into consideration. The entire experimental setup is experimentally tested and validated using a secure test bed with four machines running in Kali Linux operating system. From the experimental results, it is confirmed that the proposed ensembling model has an accuracy of more than 90% even for unknown attacks.

Keywords: Ensemble, Machine Learning, Wireless Attacks, MANET, Wireless Networks

1. INTRODUCTION

MANETs are self-organized mobile nodes, which are capable of communicating with each other without any central support infrastructure. Nodes in the MANET share the wireless medium for communication and the topological structure in MANET is dynamic and changes frequently depending on the node mobility. The proximity and density of the nodes depends upon the applications in which they are using. The main goal of MANET is to provide the mobile paradigm for many wireless domains. Yet, the MANET possesses a lot of vulnerabilities and security threats and the need for effective security system is required. The existing security system in the MANET focus only on the signature based security systems. Hence, there is a need for efficient and robust security system for MANET.

The contributions of this paper are stated as follows:

- This paper investigates the state of the art techniques and gives the clear insight about the real threat to the MANET.
- Proposed an ensemble of classifiers and analyzed the performance of the classifiers.
- Validated and processed a whole set of data with real time test bed and validation.

This paper is organized as follows. Section I deals with the Introduction and Section II deals with the literature survey. In this section, the state of the art techniques available in the literature are discussed and observation is clearly made. Section III exhibits the proposed ensemble of classifiers model. Section IV discusses about the results and performance analysis. Finally, Section V concludes the paper with the future direction of extending the proposed model.

II LITERATURE SURVEY

This section deals with the state of the art Intrusion detection technique available in the literature. Cheng-Yuan Ho et al. proposed a mechanism for false positive/negative assessment with multiple IDSs/IPSs to collect FP and FN cases from real-world traffic and statistically analyze these cases. From the statistical analysis results, three interesting findings were obtained. First, more than 92.85 percent of false cases are FPs even if the number of attack types for FP and FN are similar. Secondly, about 91 percent of FP alerts, equal to about 85 percent of false cases, are not related to security issues, but to manage policy. Hence, this causes alerts to be triggered easily regardless of whether the P2P application has malicious traffic or not. The last finding shows that buffer overflow, SQL server attacks, and worm slammer attacks account for 93 percent of FNs, even though they are aged attacks. To evade IDS/IPS detection, this indicates that these attacks always have new variations [2].

Tian et al. presented a novel method for detecting anomalous program behavior, which is applicable to host based intrusion detection systems that monitor system call activities [3]. To characterize the normal behavior of a privileged program and associates the states of the Markov chain with a unique system calls in training data, the method constructs a homogeneous Markov chain model. The probabilities that the Markov chain model supports the system call sequences generated by the program are computed at the detection stage. Based on the number of anomalous sequences in a locality frame is adopted to classify the program's behavior. The method is applicable to both computational efficiency and detection accuracy. This is especially suitable for on-line detection [4].

Revised Version Manuscript Received on 14 February, 2019.

N. Ravi, Research Scholar, Department of Computer and Information Sciences, Annamalai University, Annamalainagar, Tamil Nadu, India (Email:csravi2017@gmail.com)

Dr. G. Ramachandran, Assistant Professor, Department of Computer and Information Sciences, Annamalai University, Annamalainagar, Tamil Nadu, India (Email: gmrama1975@gmail.com)

Shengrong Bu et al. described an intrusion detection system which is important in MANETs to effectively identify malicious activities. Continuous user authentication is a preventive method to protect high security mobile adhoc networks. They have formulated the problem as a Partially Observable Markov Decision Process (POMDP) multi-armed bandit problem, to obtain the scheme of combining continuous user authentication and IDSs, in a distributed manner. To solve the problem for a large network with a variety of nodes, structural results are presented. The structural results are easy to implement in practical MANET. In the proposed scheme, the simulation results are effective [5].

Zhao et al. presented a new intrusion detection model based on neural networks. The proposed new model which uses neural network to detect, transforms pattern recognition in to numerical calculation, thereby speeding up the detection rate, while combining with expert system detection. The real time neural network training set improves the detection accuracy [6].

Carol et al proposed Dirichlet-based trust management to measure the level of trust among IDSs according to their mutual experience. According to their trustworthiness, an acquaintance management algorithm is also proposed to allow each IDS to manage its acquaintances. This approach achieves strong scalability properties and is robust against common insider threats. Then it is resulting in an effective CIDN. By evaluating the approach based on simulated CIDN, demonstrating its improved robustness, efficiency scalability for collaborative intrusion detection in comparison with other existing models [7].

Salama et al. proposed a signature based intrusion detection system using data mining and machine learning. These proposed system works based on apriori algorithm of data mining. Signatures are captured and analyzed through data mining techniques and result outperforms existing algorithms showed through demonstration. They proposed a system, which is an order based system which rely on optimization for LAN security, They mentioned that firewall, IDS and CCDS are three terms used to protect the system under attacks. The order which they given through optimization is used to protect against covert channel detection, which also increase the system performance in certain conditions [8].

Zhao et al. proposed a hybrid IDS which is a combination of Network IDS and Host IDS. They presented that analysis of intrusions is very complicate due to its abnormal behavior, so the analysis of behavior is done with set of extracted features with data mining and proposed a hybrid IDS. The achieved result is far better than existing algorithms by experimenting simulations [9].

Hu et al. presented a general correlation model that comprised a comprehensive set of components in which the parameters are included to find out the alert correlation and differences by using intrusion detection data sets. A framework analyzed a different data set with effective correlation exist with different nature of components [10]. Sang et al. presented an intrusion detection system using evolutionary neural network which reduces the time complexity of attaining neural values with weights and it also finds the neural structures simultaneously. The proposed model is tested with KDD99 datasets and attains an expected results and tool is used to find an intrusion in a system with less time [11].

Nong et al. presented an intrusion detection system with forecasting method. In this work they presented two methods such as (i) long time averaging forecasting method (ii) Exponentially weighted moving average. In the second method, they used markov chain process to training and testing. The two forecasting method is tested with some intrusions provided through systems, in that chi square distance metric used EWMA method provides an expected results which is better than long time averaging method [12].

Fovino et al. presented an intrusion detection system for grids, power plants and SCADA systems. Traditional security systems fail to protect these kinds of systems. The intrusion detection system is proposed to protect these kinds of smart grids, power plants. It has critical state analysis method to achieve better results than existing method [13].

Papamartzivanos et al. presented novel misuse IDS to increase the Attack Detection rate (ADR) [14] which has scalable, self-learning, MAPK framework, which is used to detect misuse IDS. Deep learning is used to extract the optimized features in less time which increases the accuracy of 73.3 % misuse IDS in various environments [15].

Peiying et al. presented SVM IDS in which genetic algorithm plays a vital role in selecting the optimized features. The three parameters like feature selection, weights, parameter optimization is done with support vector machine for classification [16]. The genetic algorithm predicts the best features for classification through optimization techniques, which increases the accuracy of IDS and increases the time complexity with less false positive rate [17].

From the literature, it is observed that the existing schemes in MANET IDS are unstable and less responsive. Building a reactive IDS scheme for MANET is also very challenging and requires precision time accuracy due to its mobility as it is the primary nature. Further, the state of art schemes investigating on the attacks related to MANET is also very less in number and most of the techniques deals only with the simulation environment. Hence, there is a pressing need for an efficient Intrusion Detection System for MANET. To address these issues, this paper proposes a novel and harden classification technique, which comprises of multiple classifiers called ensembles.

III PROPOSED ENSEMBLE OF CLASSIFIER MODEL

Ensemble of classifiers is a technique where various ML models are integrated together to build a more precise prediction output. In this methodology, we typically use a labeled data set for supervised learning with which, our interest would generally be classification. Ensemble of Classifiers builds up on the normal method of using the labeled data to label a new data set that has yet to be labeled. The idea is very simplistic, since it is known to us that we get

variations even while training using the same parameters and hyper-parameters, we try to obtain the best of all the possible classifiers that can be built up. To put it in a line, we choose the best set of weights for our particular classification problem.

Now that we know what our final goal is, we can look at how to get there. Since we want the best possible set of weights for classification, it is obvious that we must somehow build up on the previous "dumber" set of weights that did not reach a satisfactory level of classification, i.e. a weak classifier. It is here that you will be interest to note that, using multitudes of weak classifiers indeed gives us a single strong classifier. However, one must be care whilst using ensembles, as they are particularly prone to overfitting due to flexibility over the dataset as each weak classifier will be sure of a particular feature or set of features in the data set. Thus while using ensembles it is highly advised to promote diversity among the weak classifiers, although using more random techniques such as the random-forest have been known to give better ensembles.

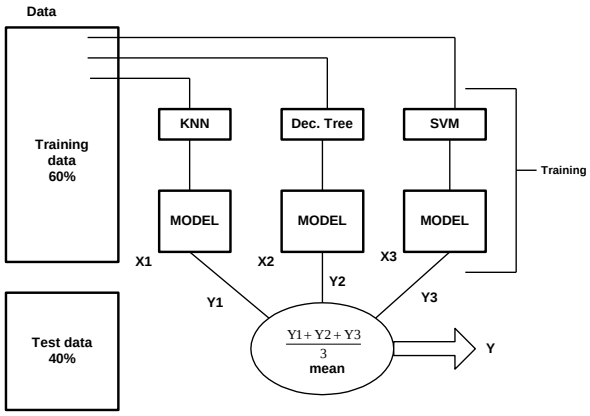

Figure 1. Ensemble of Classifier Model

Figure 1 shows the architecture of the ensemble model. In ensembles, we can further divide the methods into two groups based on the way the weak classifiers are generated. They are as shown below:

Sequential or Dependent Ensemble Methods: In this form of implementation of ensembles, the weak classifiers are generated sequentially. This means that weak classifiers that were made in the previous iteration will play their part in influencing the next weak classifier. This method repeatedly runs through the data set making weak classifiers, which are then combined to make a single strong classifier, and since we are only interested in learning from the misclassifications, we usually ignore the correctly classified instances.

Example of where sequential ensembles would be used is Boosting, AdaBoosting.

Parallel or Independent Methods: In this form of implementation of ensembles, the weak classifiers are generated parallelly. This means that we exploit the fact that each weak learner is good at one particular part of classification, and we then combine all of these weak leaners to create a strong classifier. Due to this, the weak leaners themselves need not work with the entire set of features, unlike the case with Dependent methods, rather their training set may be a subset of the original.

Example of where parallel ensembles would be used is Bagging.

Implementation: Sequential or Dependent Ensemble Methods

Boosting

The most used strategy is known as Boosting, and a slight variation of that is AdaBoost. In Boosting, we repeatedly run the weak leaners through the database, and the classifiers produces are then combined to a single strong classifier that has higher accuracy than the weak leaner had. In the next iteration, the weak learner is combined with the already made strong classifier to again improve the accuracy we get whilst classifying.

AdaBoosting

AdaBoost which stands for Adaptive Boosting, is a slight variation of what we have seen up till now in Boosting. Here instead of simply changing the weights equally, we tweak the weight so that we can give more focus to harder to notice patterns. We achieve this by initially having the same set of weights in each iteration, however, we increase the weights of the correctly classified instances and reduce the weights of those that are not. By doing this, the weak leaner is forced to go over the harder to notice patterns more thoroughly. In addition to this, each weak classifier is itself assigned a weight that states how accurate it is, higher weight states more overall accuracy. And due to this we get a series of classifiers that complement the previous one and have increasing weights assigned to them.

However, Boosting and AdaBoosting suffers from major drawbacks,

The final classifier may still misclassify if the weak classifiers had a very high rate of misclassification. Alternatively, the final classifier can also lean towards picking overfit weak leaner leading to the final classifier also being overfit

The process of rerunning through the database every time after changing the weights increases the weights exponentially

Many variations in AdaBoosting are carried out to overcome these drawbacks with different modifications to the algorithm such as M2, P-AdaBoost, Local AdaBoost. AdaBoost-r, etc.

Implementation: Parallel or Independent Methods

Bagging

Bagging stands for Bootstrap Aggregation. Here we sample the original dataset with replacement to create new training dataset of varying sizes. This kind of sampling is called a bootstrap sample. Then models are then fitted using the newly created datasets. The final classifier combines all these different models' weights by averaging in the case of regression, or by voting in the case of classification. Bagging is especially good in the case of unstable procedures such as

regression trees, and they have adverse effects on the accuracy for stable methods such as KNN. Bagging differs from Boosting in the way that, in Bagging each instance/feature is chosen with equal probability whereas in Boosting they are chosen according to their weights. Also as mentioned, Bagging requires the weak classifier to be unstable, on the other hand Boosting does not, requiring only the weights assigned to each classifier according to their weights.

Random Forest

In this method, we construct many decision trees each of which is run a very small number of features when comparing to the entire training set. It is interesting to note that, Bagging can be thought of a special case of Random forest when each decision tree works with the entire dataset rather than a small subset. It is due to this reason that random forest can easily correct the decision trees' tendencies to overfit to the training dataset. Usually, we use a top-down approach when using Random Forest, where the decision tree is modified in accordance of the importance of the feature being considered, which is not the case in normal decision tree is. Features, which score more, are ranked as more important hence the live on to the next iteration. However, this method of scoring the features has drawbacks such as while including categorical variables with different number of levels, random forests are biased in favor of those attributes with more levels.

Experimental Setup

The proposed ensemble model is experimentally tested in the test bed. The test bed is configured with four Intel i7 machine running in Kali Linux Operating System with 8 GB RAM and a dedicated wireless router is used. The various tools used to launch wireless attacks are Airmon-ng, Airodump-ng, Aireplay-ng and Aircrack-ng. These tools are mostly used through Kali linux operating system to attack the wireless router. The first stage of the attack is by Airmon-ng for sniffing the Access Points (AP) interface and to retrieve basic channel information for WLAN0 and monitor mode is enabled (MON0, MON1 …. MONn) in this phase. Once the information about WLAN channel interface is obtained, the second stage of injection will be carried on, using Airodump-ng. Finally, based on the gained information, a replay attack is performing against the wireless router. Figure 2 shows the experimental setup (test bed) of the proposed ensemble model.

Figure 2. Experimental Setup – Test Bed

IV RESULTS AND DISCUSSION

The performance of the proposed ensemble model is tested and validated with the state of the art model [1]. Figure 3 to Figure 12 shows the performance results of the proposed ensemble of classifier model versus the cuckoo search intrusion detection model.

Figure 3 and Figure 4 shows the power consumption during the normal scenario and attack scenario for the proposed model and the state of the art model. From the Figure 3 and 4, it is inferred that the proposed ensemble model consumes very less power when compared to the state of the art model for both the scenarios.

Figure 5 and Figure 6 shows the traffic during the normal scenario and attack scenario for the proposed model and the state of the art model. From the Figure 5 and 6, it is inferred that the proposed ensemble model takes very less time than the state of the art model for both the attack and normal scenarios.

Figure 7 to Figure 12 shows the traffic and behaviour of the test bed when the attack severity is increased. In order to analyze the behaviour, attack severity level has been increased and tuned. The tuning level increases with respect to multiple iterations. The iteration levels are also validated and tested up to 1000 iterations. From the experimental study, it is observed that the proposed ensemble model is effective even the attack severity level is increased.

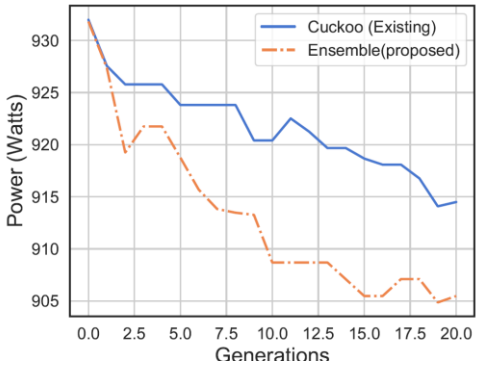

Figure 3. Power Consumption (Normal Scenario): Proposed Ensemble Model vs Cuckoo Search Model

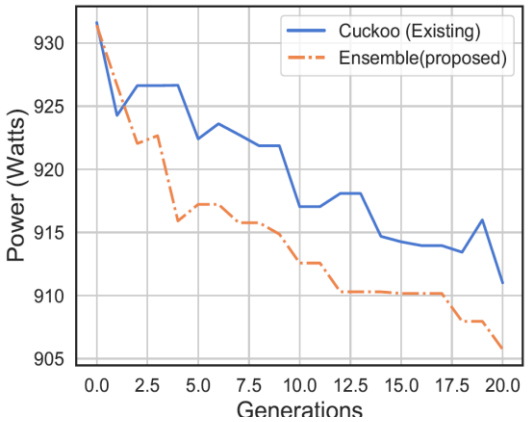

Figure 4. Power Consumption (Attack Scenario): Proposed Ensemble Model vs Cuckoo Search Model

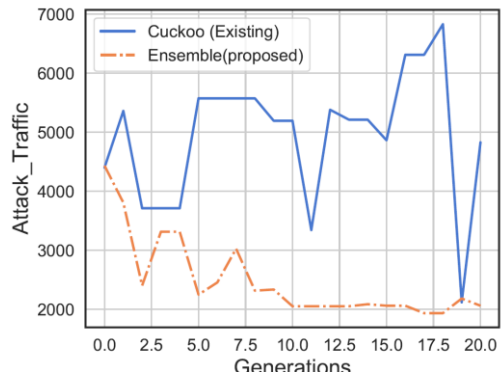

Figure 5. Normal Traffic Classification (Time in secs): Proposed Ensemble Model vs Cuckoo Search Model

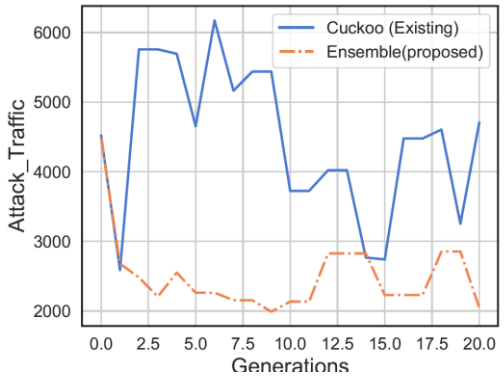

Figure 6. Attack Traffic Classification (Time in secs): Proposed Ensemble Model vs Cuckoo Search Model

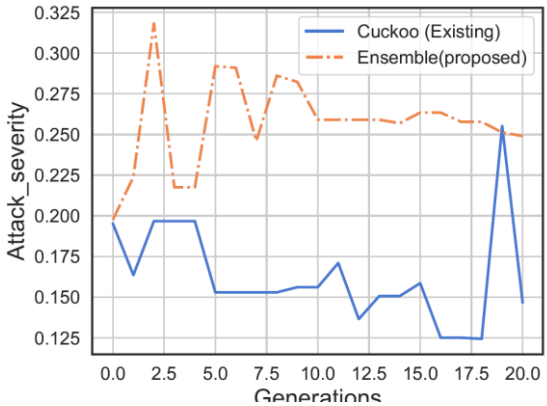

Figure 7. Attack Severity (Iteration 10): Proposed Ensemble Model vs Cuckoo Search Model

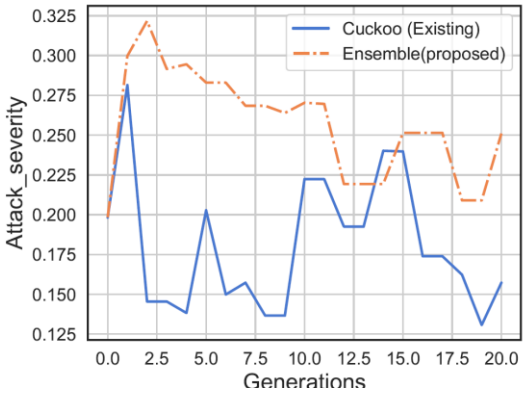

Figure 8. Attack Severity (Iteration 100): Proposed Ensemble Model vs Cuckoo Search Model

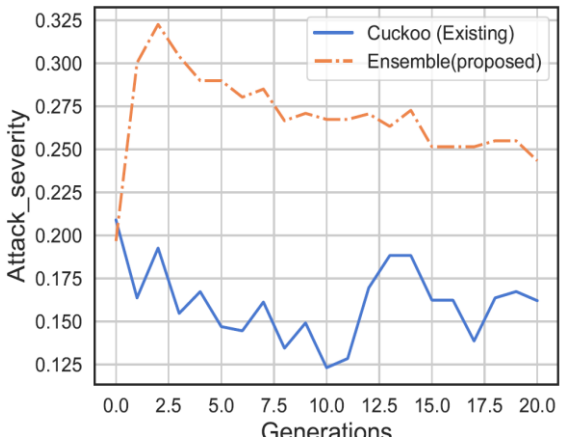

Figure 9. Attack Severity (Iteration 250): Proposed Ensemble Model vs Cuckoo Search Model

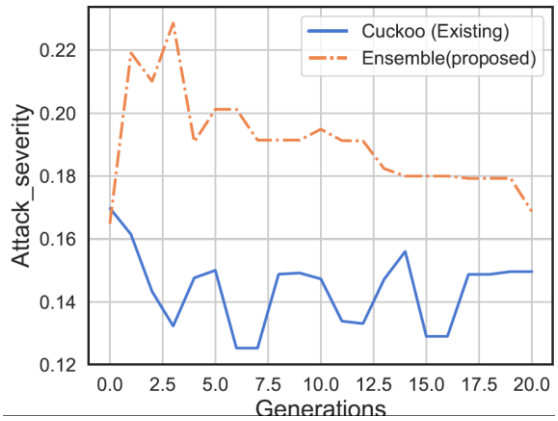

Figure 10. Attack Severity (Iteration 500): Proposed Ensemble Model vs Cuckoo Search Model

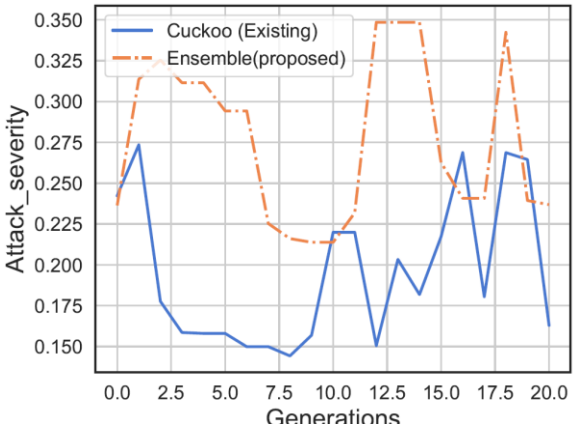

Figure 11. Attack Severity (Iteration 700): Proposed Ensemble Model vs Cuckoo Search Model

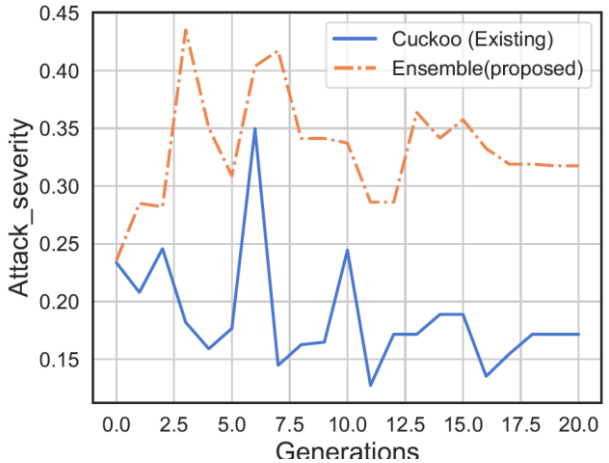

Figure 12. Attack Severity (Iteration 1000): Proposed Ensemble Model vs Cuckoo Search Model

V CONCLUSION

In this paper, an ensemble model for wireless attack classification in MANET is proposed. The existing state of the art methods are not upto the mark to detect recent sophisticated attacks. The proposed ensemble model has better detection rate when compared to the state of the art models. The experimental results confirms that the proposed model is robust and capable to detect any form of wireless attacks. Moreover, the unique point of the proposed model has ability to detect unknown attacks with more than 90% of accuracy. Furthermore, the proposed ensemble model exhibits maximum accuracy of 99.5% for known attacks. As an evident, it is clear that the proposed model is unique, efficient and has better detection ability than the state of the art schemes in terms of accuracy.

In future, the proposed ensemble models can be deployed at the mobile client version so that the user level attacks can also be prevented. Further, it is planned to go for parallelizing the server version of the proposed model for better processing and speed.

REFERENCES

1. Wang, C., Cai, W., Ye, Z., Yan, L., Wu, P., & Wang, Y., "Network Intrusion Detection Based on Lighting Search Algorithm Optimized Extreme Learning Machine", 2018 13th International Conference on Computer Science & Education (ICCSE), 2018.
2. C. Ho, Y. Lai, I. Chen, F. Wang and W. Tai, "Statistical analysis of false positives and false negatives from real traffic with intrusion detection/prevention systems", In IEEE Communications Magazine, vol. 50, no. 3, pp. 146-154, March 2012.
3. Tian, Xinguang, Xueqi Cheng, Miyi Duan, Rui Liao, Hong Chen, and Xiaojuan Chen., "Network intrusion detection based on system calls and data mining", Frontiers of Computer Science in China Vol. 4, No. 4, 522-528, 2010.
4. K. Xu, K. Tian, D. Yao and B. G. Ryder, "A Sharper Sense of Self: Probabilistic Reasoning of Program Behaviors for Anomaly Detection with Context Sensitivity", 2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Toulouse, 2016, pp. 467-478.
5. S. Bu, F. R. Yu, X. P. Liu and H. Tang, "Structural Results for Combined Continuous User Authentication and Intrusion Detection in High Security Mobile Ad-Hoc Networks", In IEEE Transactions on Wireless Communications, vol. 10, no. 9, pp. 3064-3073, September 2011.
6. J. Zhao, M. Chen and Q. Luo, "Research of intrusion detection system based on neural networks", 2011 IEEE 3rd International Conference on Communication Software and Networks, Xi'an, 2011, pp. 174-178.
7. Carol J. Fung, Jie Zhang, and RaoufBoutaba., "Dirichlet-based Trust Management for Effective Collaborative Intrusion Detection Networks", IEEE Transaction on Network Service and Management (TNSM), Vol. 8(2), pp. 79-91, 2011.
8. EzzatSalama, Shaimaa& I. Marie, Mohamed & M. El-Fangary, Laila & K. Helmy, Yehia. (2012), "Web Anomaly Misuse Intrusion Detection Framework for SQL Injection Detection", International Journal of Advanced Computer Science and Applications. 3. 10.14569/IJACSA.2012.030321.
9. Zhao J., Li W. (2012), "Improvement Intrusion Detection Based on SVM", In: Liu C., Wang L., Yang A. (eds) Information Computing and Applications. ICICA 2012. Communications in Computer and Information Science, vol 308. Springer, Berlin, Heidelberg.
10. Hu, Y., Yang, A., Li, H., Sun, Y., & Sun, L. (2018)., "A survey of intrusion detection on industrial control systems", International Journal of Distributed Sensor Networks.
11. Kai Peng, Victor C. M. Leung, Lixin Zheng, Shangguang Wang, Chao Huang, and Tao Lin, "Intrusion Detection System Based on Decision Tree over Big Data in Fog Environment", Wireless Communications and Mobile Computing, vol. 2018, Article ID 4680867, 10 pages, 2018.
12. Nong Ye, Xiangyang Li, Qiang Chen, S. M. Emran and Mingming Xu, "Probabilistic techniques for intrusion detection based on computer audit data", In IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 31, no. 4, pp. 266-274, July 2001.
13. Fovino I.N., Masera M., Guglielmi M., Carcano A., Trombetta A. (2010), "Distributed Intrusion Detection System for SCADA Protocols", In: Moore T., Shenoi S. (eds) Critical Infrastructure Protection IV. ICCIP 2010. IFIP Advances in Information and Communication Technology, vol 342. Springer, Berlin, Heidelberg.
14. Sethuraman, Sibi Chakkaravarthy; Dhamodaran, Sangeetha; Vijayakumar, Vaidehi, "Intrusion detection system for detecting wireless attacks", In IEEE 802.11 networks, IET Networks, 2018, DOI: 10.1049/iet-net.2018.5050
15. D. Papamartzivanos, F. Gómez Mármol and G. Kambourakis, "Introducing Deep Learning Self-Adaptive Misuse Network Intrusion Detection Systems", In IEEE Access, vol. 7, pp. 13546-13560, 2019.
16. Ethala, Kamalanaban, et al. "WIDS Real-Time Intrusion Detection System Using Entrophical Approach." Advances in Intelligent Systems and Computing Artificial Intelligence and Evolutionary Algorithms in Engineering Systems, 2014, pp. 73–79,DOI:10.1007/978-81-322-2126-5_9
17. P. Tao, Z. Sun and Z. Sun, "An Improved Intrusion Detection Algorithm Based on GA and SVM", In IEEE Access, vol. 6, pp. 13624-13631, 2018.