IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-5, May 2018 E-ISSN: 2347-2693

Performance Study on Malicious Program Prediction Using Classification Techniques

K. Thyagarajan [1*], N. Vaishnavi [2]

[1*] Dept of Computer Science, AVC College, Mayiladuthurai, India
[2] Dept of Computer Science, AVC College, Mayiladuthurai, India

Corresponding Author: vaishumca1494@gmail.com
Available online at: www.ijcseonline.org

18/May/2018, Published: 31/May/2018

***Abstract*—** Data mining is that the method of move queries and extracting patterns, typically antecedently unknown from giant quantities of data using pattern matching several applications in security as well as for national security likewise as for cyber security. Research focus on Detecting Malicious Packet uses weka. Once network routers are a unit subverted to act during a malicious fashion. To observe the existence of compromised routers during a network, then take away them from the routing fabric. Our approach is to separate the matter into three sub-problems: 1) crucial the traffic data to record upon that to base the detection, 2) synchronizing routers to gather traffic data and distributing this data among them thus detection will occur, and 3) taking countermeasures once detection happens. Experimental results show that ready to observe and isolate a spread of malicious router actions with acceptable overhead and quality. Our work has ready to tolerate attacks on key network infrastructure elements.

***Keywords*—** Data mining, Malicious program, JRip, PART, OneR, Malicious classifier, classification, WEKA tool.

I. INTRODUCTION

In this system we are utilizing data mining strategies, we were arranging a system that will naturally plan and manufacture a scanner that precisely identifies malicious executables before they get an opportunity to keep running on the system.

Data mining techniques recognize designs in a lot of data, for example, byte code, and utilize these examples to distinguish future questions in comparative data in the database. Our framework utilizes classifiers to recognize new malicious executables. A classifier is an arrangement of principles, or identification show which is created by the data mining algorithm that was prepared on a given arrangement of preparing data.

One of the essential issues looked in this day and age by the virus community is to discover strategies for identifying new malicious programs that have not yet been examined before . Numerous malicious programs are made each day on the planet by hackers and most can't be precisely recognized until the point when legitimate marks have been produced for them. Amid this day and age, systems ensured by signature-based algorithms are helpless against assaults on such malicious codes.

In this paper we have used a malicious dataset for order system. The means incorporate gathering of informational data set for the basic the exactness, order and afterward correlation the outcomes. The data set has been utilized 42 attributes and 34041 instances.

In this paper we have played out the examination utilizing algorithms by utilizing explorer of WEKA Tool. The indication of the paper is introduced as takes after. Section II records over each other work. Section III communicates some major idea of classification algorithms. Section IV depicts trial after effects of the arrangement classification algorithm for malicious dataset. At last, Section V the finish of this exploration work.

II. RELATED WORK

I.H. Witten et al. [1] We consider details of most relevant malware detection techniques in this section. In recent years many malware researchers have focused on data mining to detect unknown malwares. Datamining is the process of analyzing electronically stored data by automatically searching for patterns. Machine learning algorithms have been used widely for different data mining problems to detect patterns and to find correlations between data instances and attributes. Many researchers have used n-grams or API calls as their primary type of feature that are used to represent malware instances in a suitable format for data mining purposes.

M. G. Shultz et al. [2] proposed a method using data mining techniques for detecting new malicious executables. Three different types of features are extracted from the executables,

i.e. the list of DLLs used by the binary, the list of DLL function calls, and number of different system calls used within each DLL. Also they analyze byte sequences extracted from the hexdump of an executable. The data set consisted of 4,266 files out of which 3,265 were malicious and 1,001 were legitimate or benign programs. A rule induction algorithm called

W. Cohen, [3] Ripper was applied to find patterns in the DLL data. A learning algorithm Naïve Bayes (NB), which is based on Bayesian statistics, was used to find patterns in the string data and n-grams of byte sequences were used as input data for the Multinomial Naïve Bayes algorithm. A data set is partitioned in two data sets, i.e., a test data set and a training data set. This is to allow for performance testing on data that are independent from the data used to generate the classifiers. The Naïve Bayes algorithm, using strings as input data, yielded the highest classification performance with an accuracy of 97.11%. The authors compared their results with traditional signature-based methods and claimed that the data mining-based detection rate of new malware was twice as high in comparison to the signature-based algorithm.

A similar approach was used by J. Z. Kolter et al. [4], where they use n-gram analysis and data mining approaches to detect malicious executables in the wild. The authors used a hexdump utility to convert each executable to hexadecimal code in an ASCII format and produced n-gram features by combining each four-byte sequence into a single term. Their primary dataset consisted of 1971 clean and 1651 malicious programs They used different classifiers including Instance based Learner, TFIDF, Naive-Bayes, Support vector machines, Decision tree, boosted Naive-Bayes, SVMs and boosted decision tree. They used information gain to select valued features which are provided as input to all classifiers. The area under an ROC curve (AUC) is a morecomplete measure compared with the detection accuracy as they reported T. Fawcett, [5]. AUCs show that the boosted decision trees outperform rest of the classifiers for both classification problems.

M. Siddiqui et al. [6] used Data Mining for detection of Worms. They used variable length instruction sequence. Their Primary data set consists of 2,775 Windows PE files, in which in which 1,444 were worms and 1,330 were benign. They performed detection of compilers, common packers and crypto before disassembly of files. Sequence reduction was performed and97% of the sequences were removed. They used Decision Tree, Bagging and Random Forest models using Random forest performed slightly better than the others.

Johannes Kinder [7] explained a model checking method for detecting malicious code. In this paper, author presents a soft method to detect malicious code sets in executables files by using model checking. While model checking was developed to check the correctness of system against specifications, author commented that it grants equally well to the identification of malicious code patterns. In the end, they introduced the specification language Computation Trees Predicate Logics which is extending the well-known logics CTL and gave description about an efficient model checking approach their practical experiments demonstrate that they are able to detect a large number of worm variants with a single specification.

Bhavani Thuraisingham [8] explained various data mining techniques for security application. These requisition include but are not limited to malicious executables detection by mining it binary executables, anomaly detecting and data stream mining process. They summarize their acquirement and present works at the University of Texas at Dalla on intrusions detection and cyber-security research.

Kirti Mathur [9] explained the techniques for detecting and analyzing Malware executables. Computer system's security is threatened by weapons named as malware to accomplish malicious intention of its writers. Various solutions are available to detect these threats like AV Scanners, Intrusion Detection System, and Firewalls etc. These solutions of malware detection traditionally use signatures of malware to detect their presence in our system. But these methods are also evaded due to some obfuscation techniques employed by malware authors. This survey paper highlights the existing detection and analysis methodologies used for these obfuscated malicious code.

Guillermo Suarez-Tangle [10] ,showed malware in current smart devices that equipped with powerful sensing, computing and networking capabilities have proliferated lately, range from famous smart android phones and tablets to Internet devices, smart TVs, and others that will soon appear. One main feature of devices is that they have ability to incorporate third-party applications from markets. This has very strong security features and secrecy problems to user and infrastructure operator, specifically via software of malicious nature that got access to the service given by the devices and gather the sensory data and personal data. Malware in latest smart devices – Smart phones and tablets– has got fame in the previous few years, in some cases supported by best techniques designed to provide better security architecture presently in use by these devices. As important advances have been made on malware detection in computers in the last decades it is still a challenging problem.

Parisa Bahraminikoo [11] implemented artificial Intelligence in anti-virus engines. Malicious software is the software which gives partial to full control of your computer to do whatever the malware creator wants. Malware can be defined as a viruses, worms, Trojans, adwares, spywares and root kits. Spyware is a class of malware which is installed on computer that is able to collect information regarding clients without having knowledge. In 1956, the purpose of establishment of Artificial Intelligence(AI) Dart muth College during a conference. Artificial Intelligence has been implemented in anti-virus engines. AI has many approaches that implemented in spyware detection systems such as Artificial Network, Heuristic Technologies and Data Mining

Techniques. In this work, they focused on DM-based malicious code detectors by using Breadth-First Search approach for knowing work well for detection virus and software. BFS is the method for searching in a tree when search is very limited to essentially two operations (a) visit and inspect a node of a tree; (b) gain access to visit the nodes that are neighbour to currently visited node.

III. METHODOLOGY

This research uses data mining techniques for analysis and evaluation of classification algorithms of Malicious program dataset. Through open source WEKA data mining techniques, we can generate predictive model for classification of Malicious program, evaluate accuracies, and performance of several techniques.

To compare these data mining classification techniques and comparison analysis, we need the datasets. This research chooses Malicious program dataset. Directly we can apply this data in the data mining tools (Weka) and predict the results.

JRip:

In 1995 JRip was implemented by Cohen, W. W, in this algorithm were implemented a propositional rule learner, Repeated Incremental Pruning to Produce Error Reduction (RIPPER). By the way, Cohen implementing RIPPER [12] in order to increase the accuracy of rules by replacing or revising individual rules. Reduce Error Pruning was used where it isolates some data for training and decided when stop from adding more condition to a rule. By using the heuristic based on minimum description length as stopping criterion. Post - processing steps followed in the induction rule revising the regulations in the estimates obtained by global pruning strategy and it improves the accuracy.

PART:

PART algorithm [13] is a relatively simple algorithm who does not execute global optimization to generate accurate rules, but it is practiced separately and-conquer strategy, for example it builds a rule, removes the instances it covers, and continues to create a recursive rule for instances rest until there is no longer the instances is left. Furthermore, Eibe and Witten [13] said that the algorithm producing sets of rules called 'decision lists' which are ordered set of rules. A new data is compared to each rule in the list in turn, and the item is assigned the category of the first matching rule (a default is applied if no rule successfully matches). PART builds a partial C4.5 decision tree in every iterative and makes the best" leaf into a rule. The algorithm is a combination of C4.5 and RIPPER rule learning.

OneR:

OneR, short for "One Rule", is a simple classification algorithm that generates a one-level decision tree. OneR is able to infer typically simple, yet accurate, classification rules from a set of instances. Comprehensive studies of OneR's performance have shown it produces rules only slightly less accurate than state-of-the-art learning schemes while producing rules that are simple for humans to interpret. OneR is also able to handle missing values and numeric attributes showing adaptability despite simplicity. The OneR algorithm creates one rule for each attribute in the training data, and then selects the rule with the smallest error rate as its 'one rule'. To create a rule for an attribute, the most frequent class for each attribute value must be determined. The most frequent class is simply the class that appears most often for that attribute value. A rule is simply a set of attribute values bound to their majority class; one such binding for each attribute value of the attribute the rule is based on [14].

Malicious Classifier:

Our approach uses weka to detect the malicious packet. They develop a compromised router detection protocol that dynamically infers the precise number of congestive packet losses that will occur. Once the congestion ambiguity is removed, subsequent packet losses can be safely attributed to malicious actions. They trust our convention is the first to consequently predict congestion in a precise way and that it is essential for making any such network fault discovery down to earth. The conduct of the line is deterministic, the traffic validation components identify traffic faulty routers at whatever point the real conduct of the line strays from the predicted conduct. Be that as it may, a faulty router can likewise be protocol faulty: it can carry on protocol faulty routers using distributed detection. It have turned to measuring the interaction of traffic load and buffer occupancy explicitly. Given an output buffered first-in first-out (FIFO) router, congestion can be predicted precisely as a function of the inputs (the traffic rate delivered from all input ports destined to the target output port), the capacity of the output buffer, and the speed of the output link. A packet will be lost only if packet input rates from all sources exceed the output link speed for long enough. If such measurements are taken with high precision it should even be possible to predict individual packet losses. It is this approach that they consider further in the rest of this thesis. They restrict our discussion to output buffered switches for simplicity although the same approach can be extended to input buffered switches.

IV. EXPERIMENTAL RESULT

In this section, we conducted an experiment using Weka application. Weka is a comprehensive suite of Java class libraries that perform many advanced machine learning and data mining algorithms [17]. The data set of experiment has been collected UCI repository. This data set contains 42 attributes and 34041 instances. We analyze and compare the performance of Rule classifier algorithms, namely: JRip, PART, OneR, Malicious classifier.

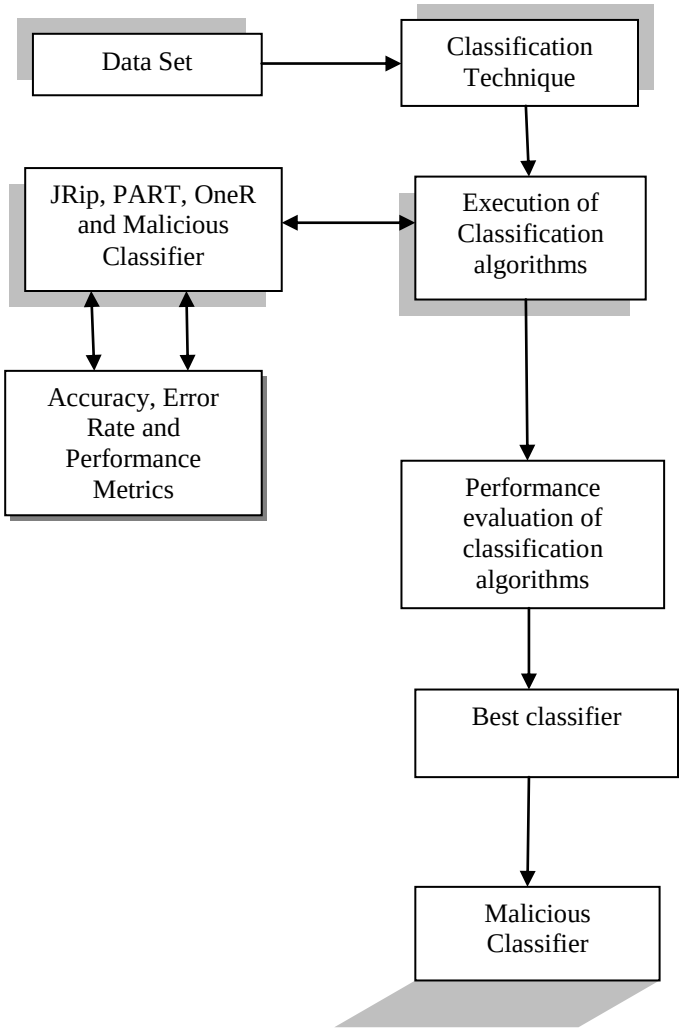

Figure 1. Working Architecture of Proposed work

The similarity comes about are parceled into a few sub things for less demanding examination and assessment. Diverse execution network like accuracy, True Positive rate, False Positive rate, Precision, Recall and F Measure are introduced in numeric incentive training preparing and testing stage. The rundown of those outcomes by running the systems in WEKA.

Table 1.Comparision of accuracy measures for the classification algorithm using malicious datasets.

Datasets	Algorithm	Correctly Classified	Incorrectly Classified
Malicious	JRip	30216	3825
	PART	30056	3985
	OneR	30186	3855
	Malicious Classifier	31541	2500

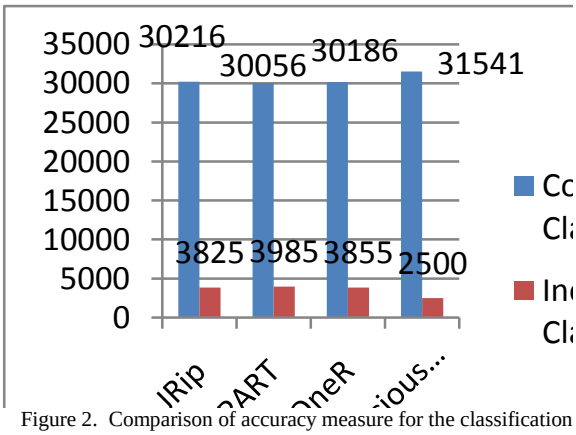

Figure 2. Comparison of accuracy measure for the classification algorithm using malicious datasets.

From the outcomes Table 1 it is gathered that for the malicious dataset the proposed algorithm performs well when contrasted with JRip (RIPPER), PART and OneR. The proposed algorithm (Malicious Classifier algorithm) gives all the more Correctly Instances contrasted with others.

Table 2. Comparison of Error rate measures for the classification algorithm using malicious datasets.

Algorithm	MAE	RMSE	RAE	RRSE	Kappa Statistic
JRip	0.0122	0.0752	0.0952	0.0889	0.0453
PART	0.0155	0.0554	0.0983	0.0870	0.0439
OneR	0.0185	0.0400	0.0855	0.1084	0.0303
Malicious Classifier	0.0100	0.0325	0.0652	0.0936	0.0416

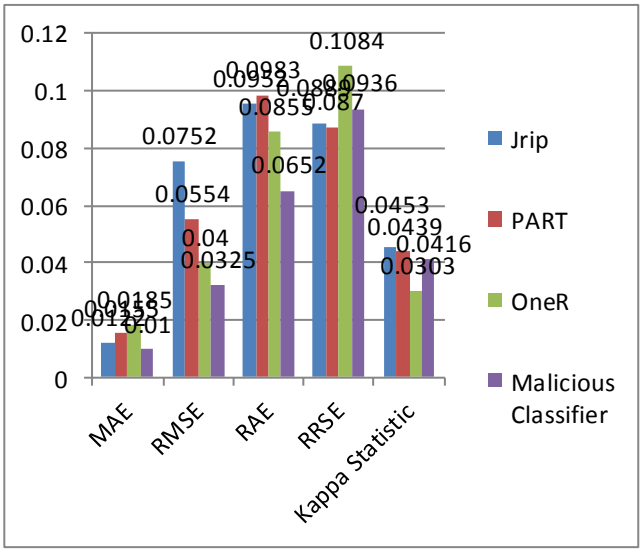

Figure 3. Comparison of Error rate measure for the classification algorithm using malicious datasets.

From the outcomes Table 2 it is derived that the Malicious datasets, the Error Rate for proposed algorithm (Malicious Classifier) is less contrasted with others. From the test comes about that the proposed algorithm the parameter Root Relative Squared Error (RRSE) expands, Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Relative Absolute Error (RAE) esteem reductions and Kappa statistics esteem fluctuates. For the JRip, PART, and OneR calculation kappa, RAE, RMSE, RRSE and MAE fluctuates.

Table 3. Comparison of Performance Measure for the classification algorithms.

Algorithm	TP Rate	FP Rate	Precision	Re-Call	F-Measure
JRip	0.996	0.425	0.896	0.996	0.985
PART	0.883	0.574	0.728	0.883	0.996
OneR	0.998	0.553	0.651	0.998	0.651
Malcious Classifier	0.842	0.438	0.699	0.774	0.648

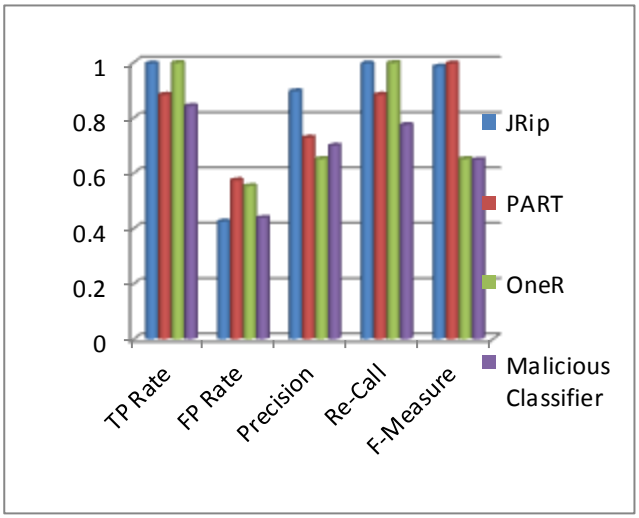

Figure 4. Comparison of Performance Measure for the classification algorithms.

The results of follow analysis on the datasets are clearly given by the table 2 and 3. Table 3 listed the True Positive Rate (TP Rate), False Positive Rate (FP Rate), Precision, Re-Call and F-Measure to analyses the classifier.

Table 4. Accuracy and Error Rate Analysis for Classification algorithms

Algorithm	Accuracy	Error Rate
JRip	88.76%	11.23%
PART	88.29%	11.70%
OneR	88.67%	11.32%
Malicious Classifier	92.65%	7.34%

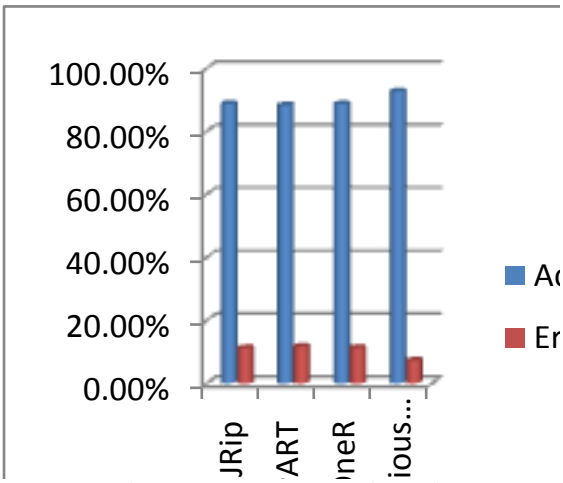

Figure 5. Diagram of Accuracy and Error Rate of Classification Algorithms

V. CONCLUSION AND FUTURE SCOPE

Data mining-subordinate malicious s code finders have been extremely effective in identifying malicious code, for example, viruses and worms. There are numerous methods that have been created till now that can powerfully adjust to new discovery techniques and kept on observing the enemy. There is a requirement for a strategy in which identification of malicious patterns in executable code arrangements should be possible all the more effectively. In this work, centers around finding the correct calculation for classification of data that works better on assorted data sets, we have met our target which is utilized to assess and examine to chosen JRip, PART, OneR and Malicious Classifier classification algorithms in view of Weka tool to foresee of best model of Malicious program. The best algorithm in light of the Malicious data is Malicious classifier with a more accuracy and less Error Rate. These outcomes recommend that among the algorithm tried in light of the fact that it can possibly altogether enhance the regular arrangement strategies to be utilized on the Knime platforms.

. References

[1] I.H. Witten, E. Frank, "Data Mining: Practical Machine Learning Tools and Techniques", 2nd ed.Morgan Kaufmann, 2005.

[2] M. G. Schultz, E. Eskin, E. Z., and S. J. Stolfo, "Data mining methods for detection of new malicious executables," in Proceedings of the IEEE Symp. on Security and Privacy, pp. 38-49, 2001.

[3] W. Cohen, ."Fast effective rule induction,." Proc. 12th International Conference on Machine Learning, San Francisco, CA: Morgan Kaufmann Publishers, pp. 115-23, 1995.

[4] J. Z. Kolter and M. A. Maloof, "Learning to Detect Malicious Executables in the wild," in Proceedings of the ACM Symp. on Knowledge Discovery and Data Mining (KDD), pp. 470-478,August 2004.

[5] T. Fawcett, "ROC Graphs: Notes and Practical Considerations for Researchers", TR HPL-2003-4, HP Labs, USA, 2004.

[6] M. Siddiqui, M. C. Wang, J. Lee, "Detecting Internet worms Using Data Mining Techniques", Journalof Systemics, Cybernetics and Informatics, volume 6 - number 6, pp: 48-53, 2009.

[7] Johannes kinder, "Detecting Malicious Code by Model Checking",pure.rhul.ac.uk/portal/files/17566588/mcodedimva05.pdf.

[8] Bhavani Thuraisingham, "Data Mining for Security Applications", IEEE/IFIP International Conference on Embedded and Ubiquitous Computing,2008 .

[9] Kirti Mathur, " A Survey on Techniques in Detection and Analyzing Malware Executables", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 4, April 2013.

[10] Guillermo Suarez-Tangle, "Evolution, Detection and Analysis of Malware for Smart Devices" IEEE communications surveys & tutorials, accepted for publication, pp.1-27, 2013.

[11] Parisa Bahraminikoo "Utilization Data Mining to Detect Spyware", IOSR Journal of Computer Engineering (IOSRJCE),Volume 4, Issue 3, pp.01-04,2012.

[12] F. Leon, M. H. Zaharia and D. Galea, "Performance Analysis of Categorization Algorithms," International Symposium on Automatic Control and Computer Science, (2004).

[13] E. Frank and I. H. Witten, "Generating Accurate Rule Sets Without Global Optimization," International Conference on Machine Learning, pages 144-151, (1998).

[14] Gaya Buddhinath and Damien Derry, "A Simple Enhancement to One Rule Classification", Department of Computer Science & Software Engineering. University of Melbourne, Australia, (2006).

[15] Umesh Kumar Singh, Jalaj Patidar and Kailash Chandra Phuleriya, "On Mechanism to Prevent Cooperative Black Hole Attack in Mobile Ad Hoc Networks", International Journal of Scientific Research in Computer Science and Engineering, Vol.3, Issue.1, pp.11-15, 2015.

[16] Meenakshi Jamgade and Vimal Shukla , "Comparative on AODV and DSR under Black Hole Attacks Detection Scheme Using Secure RSA Algorithms in MANET", International Journal of Computer Sciences and Engineering, Vol.4, Issue.2, pp.145-150, 2016.

[17] L. Khan, M. Awad, and B. Thuraisingham, "A New Intrusion Detection System using Support Vector Machines and Hierarchical Clustering", The VLDB Journal: ACM/Springer-Verlag, 16(1), page 507-521, 2007.

[18] M. M. Masud, L. Khan, and B. Thuraisingham, "Feature based Techniques for Auto-detection of Novel Email Worms", In Proc. 11th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2007), Nanjing, China, May 2007, page 205-216.

[19] M. M. Masud, L. Khan, B. Thuraisingham, , X. Wang, P. Liu, and S. Zhu, "A Data Mining Technique to Detect Remote Exploits", In Proc. IFIP WG 11.9 International Conference on Digital Forensics, Japan, Jan 27-30, 2008.

[20] Bhavani Thuraisingham, "Data Mining for Security Application"s, IEEE/IFIP International Conference on Embedded and Ubiquitous Computing,2008 .

[21] N. Landwehr, M. Hall, and E. Frank, "Logistic model trees". For Machine Learning, Vol. 59(1-2), pp.161-205, (2005).

[22] M. G. Schultz, E. Eskin, E. Zadok and S. J. Stolfo, "Data Mining Methods for Detection of New Malicious Executables", Proceedings of the IEEE Symposium on Security and Privacy, IEEE Computer Society, 2001.

Authors Profile

Dr.K. Thyagarajan received his M.Sc (Computer Science) degree from Vinayagamission University, M.Sc(maths) degree from Madurai Kamaraj University, M.Phil (Computer Science) ,M.Phil(Maths) and Ph.D (Computer Science) degrees from Bharathidasan University. He is currently working as Hod and Associate Professor in the Department of Computer Science at AVC College (Autonomous), Mayiladuthurai. He has published serveral research papers in international jounals. His research area is Data Mining.

Miss N.Vaishnavi has completed her MCA degree from Bharathidasan University at A.V.C College(Auto) . Currently she is doing M.Phil in Computer Science at A.V.C College(Auto), Mayiladuthurai. She is doing research in the area of Datamining.

