Journal of Theoretical and Applied Information Technology
30th November 2019. Vol.97. No 22

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

RANSOMWARE DETECTION USING CLASSIFICATION METHOD AGAINST REGISTRY DATA

*ASHMIN AZMAN, [1]WARUSIA YASSIN, [1]OTHMAN MOHD, [1]MOHD FAIZAL ABDOLLAH, [1]RAIHANA SYAHIRAH ABDULLAH

*,[1]Centre for Advanced Computing Technology, Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia
E-mail: *ashminazman@gmail.com, [1]s.m.warusia@utem.edu.my

ABSTRACT

An intrusion detection system (IDS) is used to detect numerous kinds of malware attacks, and many classification methods have been introduced by the researcher to detect malware behavior. However, even though various classification method has been proposed, the detection of malware behavior remains a challenging task as the detection method focusing more on traffic data classification. Consequently, there is a lack of classification approach employed to classify Windows Registry data for malware detection. Such a situation could cause more damages if the ransomware activity intended to affect registry besides traffic. Henceforward, the objective of this paper is to study the malware behavior which targeted registry and analyzing a series of machine learning algorithm as well as identify the most accurate algorithm in the detection of malware. Thus, this paper proposes a framework for ransomware detection by using registry data as features through a number of a machine learning algorithm. Based on conducted literature, Support Vector Machine, Decision Tree, Random Forest, Jrip, and Naïve widely applied as a classification method for malware detection. The experiments have been carried out via the algorithm mentioned above against registry data that been affected by ransomware. The algorithm is capable of classifying registry data to detect ransomware activity precisely. The main contribution of this research illustrates that registry data could be examined via the proposed framework 'Malware Registry Detection Framework (MRDF)' specifically for malware detection. The findings of this experiment is the capability of the proposed method to identify ransomware activity and classify which machine learning algorithm come with the highest detection rate.

Keywords: *Ransomware, Malware Detection, Machine Learning, Registry, Classification*

1. INTRODUCTION

Malware becomes one of the significant threats in cybersecurity nowadays, with the rapid development of Internet technology. Any software performing malicious actions, including information stealing, espionage, and others can be referred to as malware [1] User can be a victim by targeted or even by accidentally downloaded software that contains malware as the malware can infect a computer and the network variously [2], [3]

Ransomware, are one of the known malware for its cases in attacking by locking victim files and demand for money in exchange for decrypting the file[4]. Usually, it locks the computer or prevents from accessing data using private key encryption until the victim pays a ransom [5].

Recently, there is a massive attack carried out using malware, which is known as Ransomware [4] that denies user access to data files and demands a ransom for decrypting the infected system and restoring the files. In a short period, Ransomware has grown exponentially and rapidly become the most aggressive and threatening malware of recent times. Unlike traditional malware threats, ransomware can be more destructive as it may affect an entire landscape of security services, such as confidentiality, integrity, and availability, which may not only result in financial losses but may also result in significant information breaches [6]. This incident shows that ransomware is one of the threats that become crucial nowadays.

Detection of malware activity is necessary as the computer system had to face security issues. Detection of intrusion attempts to identify the attacks of the computer by examining different information records observed in network processes. Moreover, the intrusion detection system monitors the network and host if there is an attack attempt or a suspicious activity, and consequently, it can give early information and early prevention. Therefore,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

security has become essential to protect all the data stored in the computer system.

According to Cyber Security Malaysia latest data, a total of 902 cybercrime incidences reported between January and February when 227 from it are from malicious code and intrusion. For 2017, there is a total of 2011 occurrences receives reported for malicious code and intrusion. For 2019, between January and May, there is a total of 3743 reported incidents where 881 cases on malicious and intrusion.

The objective of this paper is to study the behavior of malware that targeted registry and come with effective classification model that can obtain high accurate detection, the goal is to determine the suitable method and how the features should be extracted, using the most accurate algorithm that can distinguish the behavior of the malware and legitimate. The proposed detection framework could be significant for the research community as a guideline in developing malware detection model. Hence, this paper discovers the concerns of machine learning-based malware detection, as well as looks for the more applicable classification method that able to classify registry information in examining malware behavior

This reminder section is arranged as follows: Section 1 is an introduction, section 2, discuss related work in this area. In section 3, the proposed methodology is described and used to classify malware based on the data set. This paper is followed in section 4, analyze the result of the different classification algorithms, and finally, in section 5, the paper is concluded, and several future works are suggested.

2. RELATED WORK

In this section, the background of an existing classification algorithm on malware detection in windows registry is discussed accordingly.

2.1 Classification

Malware classification methods employed widely by a number of researchers due to the fact that this approach promises to solve a real-life problem[7] [9]. Malware classification technique is competent in supporting a large data set and adaptable to change the model during runtime as it identifies the high number of malware behavior variants[10][1]. This method is known as a supervised learning algorithm that creates a formal model that can explain class and sort out data into correspondent features based [11] [2]. There is a two-phase of classification evaluation available for the supervised learning algorithm. The first phase is to learn a classifier using available labeled training data, and the other one is the testing phase, where the classifier will test the instants as normal or suspicious. In the current literature review, many researchers apply a machine learning approach to classify malware [12]. Machine learning algorithms used in this area is the Support Vector Machine, Naïve Bayes, Jrip, Random Forest, and Decision [7][9].These classification methods are widely used and popular among researchers for malware categorization.

The analysis of the literature review on the classification algorithm and the initial testing which is malware classification in windows registry led to the choice of five algorithms to use in this work; Naïve Bayes (NB), Random Forest (RF), Decision Tree (DT), Support Vector Machine (SVM) and JRip. These algorithms represent the primary classification technique available based on a differing approach to the classification.

A Decision Tree (DT) is a classifier algorithm consisting structure of the tree of related data created during the training phase. This algorithm used for making predictions on the provided data test to achieve an accurate result with the fewest number of the decision [13][7]. Usually, in DT, the tree is created using the information gain approach, and the feature which has the highest information gain value is used to form a decision [14][8]. Generally, the DT algorithm tends to be faster [11], [15], capable of producing decision ruled, easy to interpret, and understand. The other benefit of the DT algorithm is the ability of processing data set that may have errors or missing values and high predictive performance for a relatively small computational effort [14] [8].

On the other hand, Naive Bayes (NB) is a classifier algorithm that depends on the Bayes Theorem [16]. NB can be used for multiclass and also binary classification problems. The prediction against observable data is based on the highest accuracy compared to the most classical approach. Otherwise, it is assigned the prediction as to the class most present in the leaf. It is essential to notice that this accuracy surpass incrementally and that the same leaf may alter between using NB or most common [17] Regardless of its simplicity, Naive Bayes can surpass more complex classification methods [18]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Beside DT and NB, Random Forest (RF) is an effective classifier algorithm that designed to handle both classification and regression problem using an ensemble DT structure as a basis to form prediction model [12][18]. Consequently, the proficiency of such characteristics could reduce the variance and simultaneously maintain low bias [20] between a series of generated and voted DT. Random forest model is based on K decision trees. Each trees votes on which class a given independent variable X belongs to, and the only vote is given considers most appropriate. [21]

Support Vector Machines (SVM) is one of the classifiers that commonly used as classification. SVMs is an algorithm that uses a hypothesis space of linear functions in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias which derived from statistical learning theory. The main significant of SVM is it less susceptible to overfitting future input from the input item as SVM is an independent feature space and contributes to higher accuracy.[22] SVM offers an excellent performance of effectiveness and vitality due to the minimization of structural risk and high generalization ability [23]. SVM was found as a promising algorithm in the design of IDS as it performed excellent performance due to structural risk minimization ability [24]

JRIP (also known as RIPPER) is one of the popular classifier algorithms that have the ability of repeated incremental pruning has proposed by [25]. In JRIP instances of the dataset are evaluated in increasing order, for a given dataset of threat, a set of rules is generated. JRIP (RIPPER) algorithm treats each dataset of a given database and creates a set of rules, including all the attributes of the classes. Then the next classes will get evaluated and do the same process as the previous classes, and this process continues until all the classes have been covered[26]

2.2 Malware Detection Via Classification

There is a lot of researchers that have study on malware detection using a classification approach. Detection of malware is essential as an amount of malware such as ransomware causes huge damages all around the world. Therefore, the classification approach could be beneficial to overcome such issues as it can identify a high number of malware variants. A study on malware detection via classification approach will be discussed further. Another analysis was done by [27] using a new approach malware detection by doing a comparative analysis of the data structure in memory. The study focuses on detecting malware through a comparison of the information user. The extracted features and sample are classified according to attributes and as for a result, 98 % of detection rate and 16 % false positive.

Supervised learning random forest to address the classification of unknown binaries that executed in the sandbox [28]. The classification was provided to have a deeper understanding. The approach using 27000, 00 malware samples, and 837 samples of benign software. The experiment was able to detect more than 90% and work with extensive data. However, the drawback was it only can work with signature behavior.

An author in [23] focused on an adaptive and robust intrusion detection technique using Hypergraph based Genetic Algorithm (HG GA) for parameter setting and feature selection in Support Vector Machine (SVM). The author applied SVM as it was found to be a promising algorithm in the designing of effective IDS. Moreover, SVM exhibits excellent performance with its efficiency and robustness due to the fact of structural risk minimization and high generalization ability. The proposed algorithm has achieved an accuracy rate of 97.14 % with 0. 83 % false alarm rate.

In the Empowering convolutional networks for malware classification and analysis paper, [29] paper focus on transferring the performance improvement achieved in the area of neural networks to model the execution sequence in disassembled malicious binaries. The result shows that a neural network has better detection, and the result is demand on the malware family behavior

Besides an author [29], the author in [20], focusing on using Random Forest as the classifier to fault classification in a complex industrial process. The aim was to improve the diversity of classification trees and the performance of classification trees in a random forest, which can simultaneously reduce the online fault complexity. The author uses a sample of 960 data sets. The outcome from this paper is Random Forest give a better accuracy when single. The Final result was 93.1% accuracy and 7.8% for false negative.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1 Malware Detection using Classification

Author's	Method	Objective	Advantage	Limitations	Achievement
[27]	Decision Tree + Random Forest + Naïve Bayes	Detecting malware through comparison of the information in the user space memory using three proposed classification	It doesn't need to install a tool to analyze the dynamic behaviour as it directly scans without an operating system involvement. The detection rate is above 90 %	In 350 malware data, the highest false-positive rate is 5.1 %, which means 17.85 data missed.	Decision tree True positive rate; 96.6& False positive rate: 5.1% Accuracy:.96.6 % Random Forest TPR: 98.4%, FTR; 2.5% Accuracy: 98.1% Naïve Bayes TPR: 98.8% FPR: 1.6 % Accuracy: 98.9 %
[28]	Random Forest	Address classification of unknown binaries that executed in the sandbox and apply classification to provided deep understanding in malware binaries. System Call and APIs using 27000,00 malware sample	Able to detect more than 90 % a and work better with large data set	Only applicable for known malware signature	Tpr: 98% fpr: 0.1% (precision: 0.9%)
[23]	HyperGraph Algorithm + Support Vector Machine	Propose an adaptive, and a robust intrusion detection technique using Hypergraph based Genetic Algorithm (HG GA) for parameter setting and feature selection in Support Vector Machine (SVM).	The detection rate was higher than 90 %	Unable to detect 4150000 data from 0.83 % false alarm out of 5000000 data.	The detection rate for the SVM with a future selection is 97.14%, and the false alarm is 0.83%
[29]	Neural Network	To improve the future extraction and classification	The neural network has better detection	Only used results from static malware	92 % score value and 93% for recall value

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

		methodology for malware dataset using neural network		analysis when it should be from both static and behavior traces.	
[20]	Random Forest + Hierarchical Clustering	To improve the diversity between classification trees and the performance of individual classification trees in random forests	WeightedRF with hierarchical clustering selection give a better accuracy	Work worse with Random selection. It provides 92.39 % accuracy with 8.4 % false negative.	False Negative Rate: 3.7% Accuracy : 96.86%

2.3 Malware Detection Via Registry Data

This section will discuss more on a number of malware detection approach in the registry data.

BOFM, a scalable model [30] developed as a detection mechanism that focuses on analyzing networks and also registry. The proposed model is to capture the interaction between malware and security-critical system resource. As for the result, the proposed model work better when it combines with the SVM with no false positive issues

Moreover, in [31], the researcher applied dynamic analysis and sandbox to build the feature vector using run-time behaviors. Furthermore, machine learning algorithms employed as a classification approach to classify malware samples. The author validates 17,900 malware, and for a result, it gets 94% as an accuracy rate.

[10] Using dynamic analysis available in the sandbox, the researcher able to detect malware location at API call and windows registry. The experiment was conducted with an extensive

amount of malware sample to identify the behavior of the data. The obtained result shows remarkable achievement with a high value of accuracy through a series of random classification methods.

Furthermore, [32] using a number of detection models such as dynamic analysis, static analysis, online evaluation, and the combination of the tree tools. The paper focuses on a number of sources, i.e., API call, DLL, and also the registry for malware detection. By using these tools, the researcher able to find out the performance on known and unknown malware that been identified by each tool prior. However, the combination of each tool at once has the highest accuracy and true positive alarm.

[33] Aim to address the classification of unknown binaries by modeling the interaction with the system sources. As such, the proposed model detection, which is MIL, Rieck, and AMAL [20] been applied. The proposed model able to detect malware approximately 95.4 % as accuracy and 6.7% as a false alarm rate.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2 Malware Detection via Registry Data

Author Name/ Year	Method	Location	Aim	Achievement
[30]	BOFM for scalable malware detection	Network, Windows Registry	Propose a simple modeling technique to captures the interactions between malware and security-critical system resource with 5300 malware set	The BOFM work better when it combines with SVM with 0% false positive
[31]	Sandbox	API call, Windows Registry	To test an extensive number of malware samples and identify the behavioural data	The result comes with a high classification rate with the value of 98% using random classification
[10]	Dynamic and static analysis	Windows Registry, API call, Network	Build the feature vector using run-time behaviors by applying online machine learning algorithms for the classification of malware samples in a distributed and scalable architecture.	The detection rate is 92 % with a high positive alarm
[32]	Dynamic analysis + Static analysis + Online Tools + Combination tools (Dynamic Static	API call, DLL, Registry	Comparing the detection tools performance on unknown and known malware	The combination of tools give the highest accuracy and true positive alarm at 87.5 %
[33]	MIL + Rieck + AMAL	Windows Registry, Network, Syscall	To address the classification of unknown binaries by modeling the interaction with the system source.	Detect more malware with 95.4% accuracies and 6.7% false alarm

Based on the related works, many previous researchers have proposed a technique in detecting malware using machine learning techniques. Even though various previous researchers detecting malware using machine learning, the method still lacks in common and less focused. First, there is less detection method focusing only on Windows Registry. They often focus on network traffic analyzing, and other locations with Windows Registry are only part of it as authors in [30], [31], [10], [32,], and [33]. The authors mostly used combination techniques and sandbox methods to detect malware in registry.

Nevertheless, the difference between this research to previous research is the study of malware behavior concentrates on Ransomware detection focusing in Windows Registry using classification method and comparing the machine learning performance in detecting malware. By analyzing registry information, a researcher can obtain malware path direction, activity, and ability to enhance the malware detection performance. This is also supported by author [34] that registry is a popular location for malware to use to maintain persistence and new malware can be discovered by analyzing the registry. Therefore, this paper focuses on detecting malware behavior in registry via proposed detection approached (MRDF)

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. PROPOSED MALWARE REGISTRY DETECTION FRAMEWORK

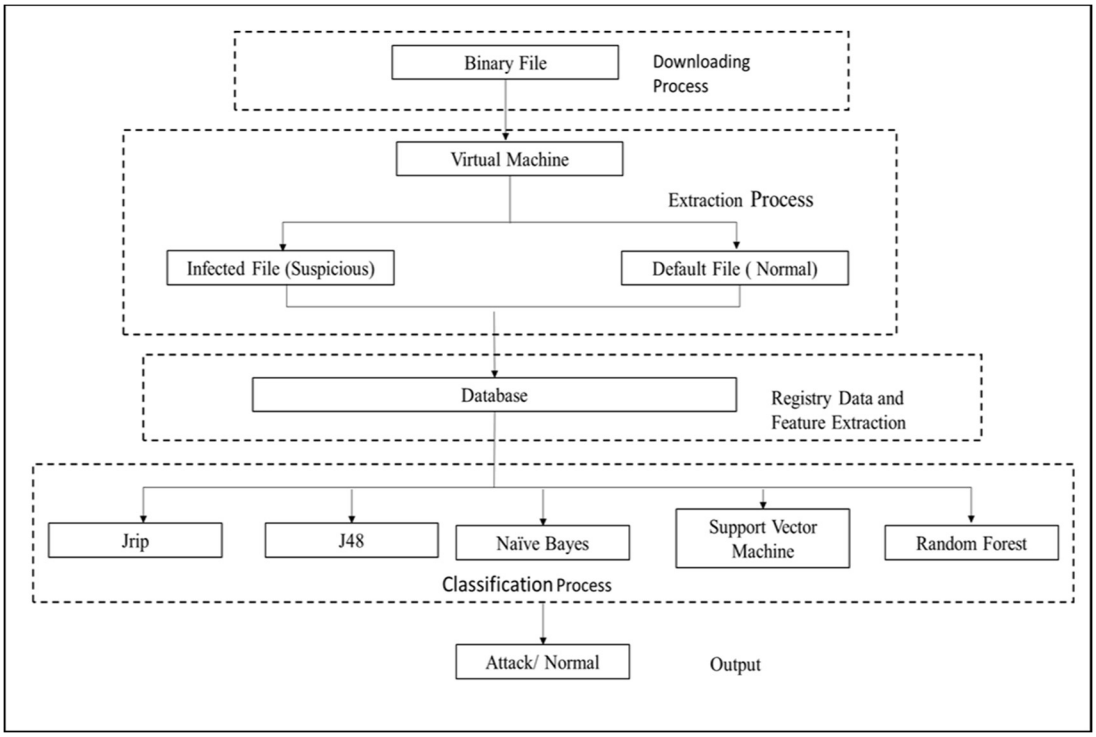

Figure 1 MRDF Framework

The proposed malware registry detection framework, namely MRDF, classifies the malware behavior based on the data and futures extracted from the dataset. The dataset is divided into two sets, such as train dataset and test dataset with the composition of 60:40. The test data is the total data samples applied to the classifier for the behavior classification. The proposed framework consists of four processes that start with downloading, followed by the extraction process, registry data, and feature and lastly classification process, as shown in Figure 1.

The first process is the downloading in which the data will be downloaded from the binary file that consists of 213 data, which will further be executed in the second process. The second process will be the extraction, where the binary file will be placed into a virtual machine and injected with malware lastly, the classification process will take place, where a series of five classification algorithm such as decision tree, Bayes Naïve, Random Forest, Support Vector Machine, and JRip employed as a malware detection method

3.1 Downloading Process

There is a total of 213 malware downloaded from Malwares.com. However, out of 213 malwares, 19 malware have similar behavior, which makes it a total of 194 malware left. Next, via using Virus Share.com, the malware identity can be determined, such as malware type, whether it is Worm, Trojan, etc. By using the Virus Total.com, the malware is classified according to their malware family. There are three types of malware families that are classified as Conficker A, Conficker B, and Conficker C. Conficker A consists of 68 malwares, Conficker B contains 67 malwares, and Conficker C contains 61 malwares, respectively. However, during the filtering process, only 32 malware found to be suited for Windows 7 environment. Due to the size of data issues, the entire 32 malwares unable to run, thus the number of malware is reduced to 9 malwares.

3.2 Extraction Process

The second process in this framework is the extraction process when the binary file is run in a

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

virtual machine, and all the information log will be captured. The information later is divided into two types, which are the default file (normal activity) and the infected file (suspicious activity).

3.3 Registry Data and Feature Extraction

Registry data and feature started with uploading the data into the database. By using the SQL query, the data is selected based on targeted data for data analysis. The RegUtil system is used to upload the srp file of the normal and infected file. It connected to a database to store the data. The normal and infected files are uploaded through the RegUtil system into a database. In the database, the data was extracted to get the only used data for the experiment by using SQLYog database. Then, the data is chunked into several paths before it can be used for analysis. To checked whether the path is correct or not, it was checked with regedit application.

Table 3 Type of Malware Type

	MD5	Size	Family
Worm	8de319c47719aac427fe4a55a36bf115	68.34 KB	A
Trojan Dropper	ff3e9e5ba698ed9f5fdcf88596cc9cf2	95KB	A
Worm	ed249f340c7c72f094af1daec58e8544	264 KB	A
Worm	1210d772c11ecfc2ec20297c0ce31ffe	124 KB	B
Worm	e76855fe8c78e98f987c598d0daf6e1d	183.23 KB	B
Worm	43657555bc398e5fc899f6f730da59ad	82.53 KB	B
Worm	700518f516a5ccfd9c476e2f569ed2a0	137.81 KB	C
Worm	0d42d8adf492ace883344082a322d1fd	528 KB	C
Worm	64ac48d91028b5f7fbe65f0121c811a6	161.86 KB	C

3.4 Classification Process

In this phase, the extraction data from the database is analyzed by using five selected algorithms, i.e., Decision Tree (DT), Random Forest (RF), Jrip, Naïve Bayes (NB), and Support Vector Machine (SVM) to perform malware detection. The performance of classification detection is evaluated based on the following measurement:

a) False Positive (FP) is the volume of normal falsely detected as an attack. Accurate ADS should be predictable to achieve missed normal to as near to zero as reasonable.
b) False Negative (FN) is the volume of attack falsely detected as normal. An accurate ADS is predictable to achieve missed attack as near to zero as reasonable.
c) True Positive (TP) is the volume of the actual attack that has been detected accurately.
d) True Negative (TN) is the volume of the actual normal that has been detected accurately.

In general, to study the performance of the proposed approach method can be used by using the following formula:

$$\text{Accuracy} = (TP+TN) / (TP+TN+FP+FN) \quad (1)$$

4. RESULT AND ANALYSIS

The result of the experiment in this work is based on attack malware behavior detection via five classifiers chosen. The comparison of the five classifiers is shown in Table 4. The result of using different classifier gives high accuracy, and the accuracy is classified using the accuracy detection rate.

Table 4 Classification Result

Classifier	Accuracy (%)	Detection Rate (%)	False Alarm (%)
DT	50	49.5	9.8
JRip	59.8	55.5	7.8
NB	59.8	55.5	7.8
RF	69.6	62.5	5.8
SVM	99	100	0

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

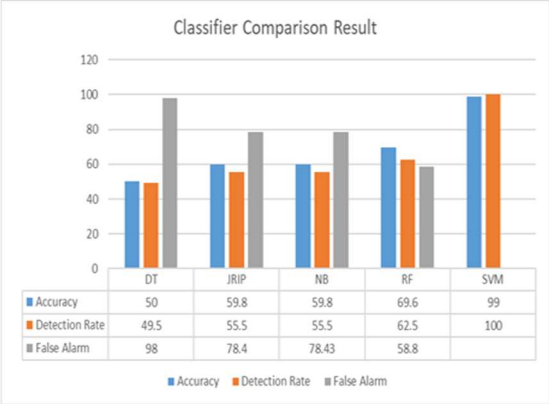

Figure 2 Classifier Comparison Result

Table 4 represents the result of classifier regarding the accuracy, detection rate, and false alarm in detecting malware obtain from different classifiers. SVM performance is more prominent than another classifier in detection normal and attack as SVM recorded with 0% of false alarm detection rate an excessive accuracy percentage. Meanwhile, RF obtains slightly higher accuracy with 69.6 % but with high false alarm as well. However, DT produces the lowest accuracy in the act of detecting malware. It only manages to classify 50 % accuracy with the higher false alarm among another classifier. Meantime, JRip and NB produce a lightly better than DT. Both classifiers shared a similar result in terms of accuracy, detection rate, and false alarm. Both classifiers achieve 59.9 % accuracy with 55.5 % of detection rate with a high false alarm.

The result of the classifier depends on the ability of the classifier to identify both normal and attack behavior. However, the data and the features play a crucial part in determining the result accuracy. An efficient data portioning improve classification performance [33].

As for the chosen approach, classifier high false alarm is caused by the misclassified data during the classification stage, hence increase the false alarm and lower accuracy that leads to a low detection rate. Both DT and Jrip classifier approach failed to identify attack behavior as both classifiers misclassified attack behavior as normal. This due to the fact that both malware has the same behavior and similar path affected. Jrip shared a similar path as normal behavior when it misclassified as malware, which is 'hkey_users,' 'control panel' and 'desktop.' Meanwhile, Jrip misclassified 'hkey_user,' 'control panel,' 'desktop' and 'muicached' as malware. As for NB, the classifier unable to identify the normal behavior even though the malware does not share a similar path as a result of the classifier itself unable to detect the behavior. RF, on the other hand, misclassified both normal and attack with the malware class, considering the malware class has similar behavior. RF misclassified the malware class when it misses class C as B when 'hkey_user,' 'control panel' 'desktop,' and 'muichaced.'

Generally, the effectiveness of detection depends on the number of false alarm created. The fewer number of false alarm is better and more efficient in detection malware. Overall, the SVM able to obtain a better magnificent detection rate compared to other classifiers in table 1 with above 99 % accuracy rate.

SVM has been misclassified the class of malware as the malware class consist of the same behavior, however it able to group respectively the normal and attack behavior. This classifier proved to better compare to another classifier that obtains high false alarm rates.

5. CONCLUSION AND FUTURE WORKS

In this paper, a comparative study of malware classification based on the registry was presented. The main goal of this paper is to study ransomware behavior using classification method, especially in Windows Registry and come with the most suitable machine learning algorithm with high detection rate. The hypothesize in this paper is the is the proposed method can detect malware with a high accuracy. As for the findings, the proposed method (MRDF) best result demonstrates a 99 % accuracy with 100 % detection rate and 0 % false alarm. The result from the MRDF shows that SVM provides high accuracy in detecting malware behavior. The limitation of this research is the experiment able to use only using three type of malware with a different variant instead of using a various samples of malware to identify their characteristics, as well their affected path in registry. The other limitation is mainly concern on the data issues such as huge volume and conversion of data format. For future work, it recommends focusing on malware family in registry information and uses statistical analysis to do scoring while including a various sample of malware to identify their characteristics, information and affected the path and to propose a new approach for registry information to detect malware behavior.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6. ACKNOWLEDGMENT

This work has been supported under Universiti Teknikal Malaysia Melaka research grant PJP/2019/FTMK(3B)/S01674. The authors would like to thank Universiti Teknikal Malaysia Melaka, and INSFORNET research group for their incredible support in this project.

REFERENCES:

[1] L. Han, S. Liu, S. Han, W. Jia, and J. Lei, "Owner based malware discrimination," *Futur. Gener. Comput. Syst.*, vol. 80, pp. 496–504, 2018.

[2] A. Feizollah, N. B. Anuar, R. Salleh, and A. W. A. Wahab, "A review on feature selection in mobile malware detection," *Digit. Investig.*, vol. 13, pp. 22–37, 2015.

[3] K. Rieck, T. Holz, C. Willems, P. Düssel, and P. Laskov, "Learning and classification of malware behavior," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5137 LNCS, pp. 108–125, 2008.

[4] S. K. Shaukat and V. J. Ribeiro, "RansomWall : A Layered Defense System against Cryptographic Ransomware Attacks using Machine Learning," pp. 356–363, 2018.

[5] R. Richardson and M. North, "Ransomware : Evolution , Mitigation and Prevention," vol. 13, no. 1, p. 2017, 2017.

[6] I. Yaqoob *et al.*, "The rise of ransomware and emerging security challenges in the Internet of Things," *Comput. Networks*, vol. 129, pp. 444–458, 2017.

[7] D. M. Farid, L. Zhang, C. M. Rahman, M. A. Hossain, and R. Strachan, "Hybrid decision tree and naïve Bayes classifiers for multi-class classification tasks," *Expert Syst. Appl.*, vol. 41, no. 4 PART 2, pp. 1937–1946, 2014.

[8] A. E. Mohamed, "Comparative Study of Supervised Machine Learning Techniques for Intrusion Detection," vol. 14, no. 3, pp. 5–10, 2017.

[9] S. Karamizadeh, S. M. Abdullah, M. Halimi, J. Shayan, and M. J. Rajabi, "Advantage and drawback of support vector machine functionality," *I4CT 2014 - 1st Int. Conf. Comput. Commun. Control Technol. Proc.*, no. I4ct, pp. 63–65, 2014.

[10] A. Pektaş and T. Acarman, "Classification of malware families based on runtime behaviors," *J. Inf. Secur. Appl.*, vol. 37, pp. 91–100, 2017.

[11] W. Yassin, "an Integrated Anomaly Intrusion Detection Scheme Using," no. June, 2015.

[12] F. M.A, W. Yassin, N. H. M.S, S. Selamat, and R. S. Abdullah, "An Analysis of System Calls Using j48 and JRIP for Malware Detection," *Journal Theor. Appl. Inf. Technol.*, vol. 96, no. 13, pp. 4294–4305, 2018.

[13] C. S. Supervisor and M. Juutilainen, "Machine Learning Methods for Malware Detection and Classification," 2017.

[14] D. AL-Nabi and S. Ahmed, "Survey on Classification Algorithms for Data Mining:(Comparison and Evaluation)," *Comput. Eng. Intell. Syst.*, vol. 1719, no. 8, pp. 18–25, 2013.

[15] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. September, pp. 1–58, 2009.

[16] X. Hoang and Q. Nguyen, "Botnet Detection Based On Machine Learning Techniques Using DNS Query Data," *Futur. Internet*, vol. 10, no. 5, p. 43, 2018.

[17] V. G. T. da Costa, B. B. Zarpelão, R. S. Miani, and S. Barbon, "Online Detection of Botnets on Network Flows using Stream Mining," *An. do Simpósio Bras. Redes Comput. e Sist. Distrib.*, vol. 36, 2018.

[18] Vikramkumar, V. B, and Trilochan, "Bayes and Naive Bayes Classifier," *arXiv*, 2014.

[19] X. Tan *et al.*, "Wireless Sensor Networks Intrusion Detection Based on SMOTE and the Random Forest Algorithm," *Sensors*, vol. 19, no. 1, p. 203, 2019.

[20] Y. Liu and Z. Ge, "Weighted random forests for fault classification in industrial processes with hierarchical clustering model selection," *J. Process Control*, vol. 64, pp. 62–70, 2018.

[21] M. A. M. Hasan, M. Nasser, S. Ahmad, and K. I. Molla, "Feature Selection for Intrusion Detection Using Random Forest," *J. Inf. Secur.*, vol. 07, no. 03, pp. 129–140, 2016.

[22] A. Sahasrabuddhe, S. Naikade, and A. Ramaswamy, "Survey on Intrusion Detection System using Data Mining Techniques," pp. 1780–1784, 2017.

[23] A. efficient intrusion detection system based on hypergraph-G. algorithm for parameter optimization and feature selection in support M. R. Gauthama Raman, N. Somu, K.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Kirthivasan, R. Liscano, and V. S. Shankar Sriram, "vector machine," *Knowledge-Based Syst.*, vol. 134, pp. 1–12, 2017.

[24] H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on SVM with feature augmentation," *Knowledge-Based Syst.*, vol. 136, pp. 130–139, 2017.

[25] B. K. B, A. Zarras, T. Lengyel, G. Webster, and C. Eckert, "Detection of Intrusions and Malware, and Vulnerability Assessment," *Detect. Intrusions Malware, Vulnerability Assess.*, pp. 419–439, 2016.

[26] A. Jain and A. Kumar, "Detection of Malicious Executables Using Rule-Based Classification Algorithms," vol. 14, pp. 35–38, 2018.

[27] M. Aghaeikheirabady, "A New Approach to Malware Detection by Comparative Analysis of Data Structures in a Memory Image," no. Ictck, pp. 26–27, 2014.

[28] S. S. Hansen, T. Mark, T. Larsen, M. Stevanovic, and J. M. Pedersen, "An Approach for Detection and Family Classification of Malware Based on Behavioral Analysis," 2016.

[29] B. Kolosnjaji, G. Eraisha, G. Webster, A. Zarras, and C. Eckert, "Empowering convolutional networks for malware classification and analysis," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2017-May, pp. 3838–3845, 2017.

[30] M. Chandramohan, H. K. Tan, L. C. Briand, and L. K. Shar, "A Scalable Approach for Malware Detection through Bounded Feature Space Behavior Modeling," pp. 312–322, 2013.

[31] R. S. Pirscoveanu, S. S. Hansen, T. M. T. Larsen, M. Stevanovic, and J. M. Pedersen, "Analysis of Malware Behavior : Type Classification using Machine Learning."

[32] Ö. Aslan, "Investigation of Possibilities to Detect Malware Using Existing Tools," 2017.

[33] J. Stiborek, T. Pevný, and M. Rehák, "Multiple instance learning for malware classification," *Expert Syst. Appl.*, vol. 93, pp. 346–357, 2018.

[34] H. Carvey, "Registry Analysis," in *Windows Registry Forensics: Advanced Digital Forensic Analysis of the Windows Registry* , 2016 , pp. 1-35.