iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** IV **Month of publication:** April 2024

DOI: https://doi.org/10.22214/ijraset.2024.61198

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue IV Apr 2024- Available at www.ijraset.com

Malware Detection and Prevention Using Machine Learning

Prof. M. E. Sanap[1], Neha Jadhav[2], Samiksha Katore[3], Srushti Mahadik[4], Akanksha Shendage[5]
[1, 2, 3, 4, 5]*Computer Engineering, SAE – Kondhwa*

Abstract: ***The malware became a serious threat challenge the computing world that requires an immediate consideration to avoid financial and moral blackmail. So, there is a real need for a new method that can detect and stop this type of attack. Most of the previous detection methods followed a dynamic analysis technique which involves a complicated process. The present study proposes a novel method based on static analysis to detect ransomware. The significant characteristic of proposed method is dispensing of disassemble process by direct extraction of features from raw byte with the use of frequent pattern mining which remarkably increases the detection speed. The Gain Ratio technique was used for feature selection which exhibited that 1000 features was the optimal number for detection process. The current study involved using random forest classifier with a comprehensive analysis to the effect of both tree and seed numbers on the ransomware detection. The results showed that tree numbers of 100 with seed number of 1 achieved best results in terms of time-consuming and accuracy. The experimental evaluation revealed that the proposed method could achieve a high accuracy of 97.74% for detection ransomware.***
Keywords: Ransomware detection; Machine learning; Random forest; Cyber security

I. INTRODUCTION

These days, the attackers use intelligent techniques to gen- erate new profitable malware type. One of these attacks whichhighly spread recently is ransomware. Malware is irreversible and difficult to stop not like other security problems. The strategy of this malware is based on access restrictionto user files by encrypting them and demands a ransom in order to obtain the decryption key. According to Symantec Corporation 2016, hundreds of millions of dollars are enforcedto be paid by users as a ransom every year. In 2016, Osterman Research and Inc. has conducted a survey including about 290 organizations from various industrial sectors in Europe and United States. The survey revealed that 50% of themhad been victims of a ransomware during a year. About 40% of these victims have paid to attackers. In another hand, a statistics report from Virus Total described that on Feb 2017 around 1.37 million of new samples cyber-attack were submitted.

The essential difference between malware and ransomware by time taken for the attack and attack behavior.While mal- ware hide behind applications and then infect and damage the computer without asking for paying the ransom. According to Chittoo parambil et al., none of the existing methods can afford detection and stop this type of attack. Besides, Weckste´n, M., et al. and Kharaz, A.,et al., confirmed to the difficulty of stopping this type of attack. Therefore, there is an urgent need to introduce new technique that can be used to detect ransomware.

This article investigates the machine learning technique for the classification of ransomware using random forest and features extracted from raw byte of the file. Different size of seed and tree have been tested experimentally in order to design the best random forest classifier that can detect ransomware accurately.

The remainder of this paper is organized as follows: Section describes the previous works in the literature on ran- somware detection; Section 3 describes the proposed method, Section includes the description of the collected datasets which used in the experiment. Section 5 illustrates the experimental results and finally Section 6 concludes the paper.

II. RELATED WORKS

Ransomware attack is launched in September of 2013 usingRSA public-key cryptography. In 2016, this attack turned The problem has spread worldwide. More than 1,400,000 Kaspersky Lab users were attacked from various locations (Kaspersky Security Bulletin, 2017). In 2017, approximately 400,000 computers in 150 countries were infected with Wanna Cry (Crowe) ransomware in one day. Therefore, in the past few years, many cyber space researchers have paid great attention to ransomware research. Analysis and the third is an attempt to use a hybrid machine that combines dynamic and static analysis. Takeuchi et al. used SVM classifier to identify ransomware based on dynamic analysis. First, we retrieve specific resources called application programming interface (API) calls, and then use Cuckoo Sandbox to inspect the history and behavior of the API calls.

API calls are represented as q-gram vectors. They used 276 ransomware and 312 legitimate files. As a result, the ransomware identification accuracy using SVM was found to be 97.48%. Vinaykumar et al proposed a new method for writing sandbox API systems using dynamic analysis. During testing, they downloaded seven types of ransomware. Classifies ransomware and malware using a multilayer perceptron (MLP). The accuracy of this method is 98%. Kharaz et al.use dynamic scanning called UNVEIL to detect ransomware.

The system creates a factory and detects ransomware without fail. The accuracy of the system is approximately 96.3%. Homayou et al. presented a ransomware detection framework based on a combination of candidate features used as machine learning inputs (MLP, Bag, Random Forest, and J48) for classification. Results showed that ransomware identification accuracy was approximately 99%. analyzed the behavior of four types of ransomware on virtual machines installed on the Windows 7 operating system. The authors use process monitoring software, task logs, task files, and recordings to monitor business processes. They claim that the ransomware attack is essentially based on the vssadmin.exe file. Therefore, to prevent this attack, users should not have access to vssadmin.exe. Zeng et al. used deep learning methods to infer ransomware behavior from network file header information. Chenet al.Artificial Intelligence (GAN). Application processes can create dynamic functions. However, it takes a long time to process and review these reviews. During this time, malicious agents are transmitted at the same time, when ransomware fingerprints the environment, it cannot extract sensitive API sequences. A recent study by Zhang et al. is a function used as an opcode.in ransomware detection.

Their method included transferring opcode sequences to N-gram sequences then Term Frequency-Inverse document frequency (TF-IDF). Five machine-learning methods were used to distinguish between ransomware and goodware such as; Decision Tree, Random Forest, K-Nearest Neighbor, Naive Bayes, and Gradient boosting. The best accuracy of 91.43% was obtained using random forest. Baldwin and Dehghantanha used static analysis to detect ransomware. They extracted the opcode characteristics as thefeatures to be used as input to the machine learning technique represented by SVM classifier. The WEKA machine learning toolset has was used in this study.

The best accuracy rate for five crypto ransomware families is around 96.5%. Subedi et al. used dynamic analysis and static analysis at three different levels; Additionally, they developed CRSTATIC, an analysis tool that uses reverse engineering to create signatures to identify ransomware families. Shaukat and Ribeiro introduced a robust trap layer using a combination of dynamic and static analysis and machine learning. They used 74 samples from 12 crypto ransomware families. The results show that the detection rate using the gradient tree boosting algorithm is approximately 98.25%. Ferrante et al. proposed a hybrid approach for Android ransomware detection. System dynamic analysis is combined with static analysis. The dynamic detection method includes memory usage, call statistics, CPU usage, and network usage, while the static detection method uses the frequency of opcodes. Moore uses honeypot folders to monitor changes occurring in folders.

Some researchers have developed special tools to detect ransomware. Kolo denker et al.proposed a Pay Break tool that stores encryption keys in the store. These keys are used to decrypt affected files after a ransomware attack. In another study, Scafe et al. proposed the use of the Crypto Drop system, which uses a set of behavioral patterns to alert users during suspicious activity. Continella et al.introduced the Shield FS system, which examines the memory system and looks for encryption signatures. In this study, byte-level static analysis was used to overcome the shortcomings of dynamic analysis. Features are extracted directly from the raw bytes of the executable and then extracted using active models. During the classification process, a random forest generator was used to classify ransomware and data quality.

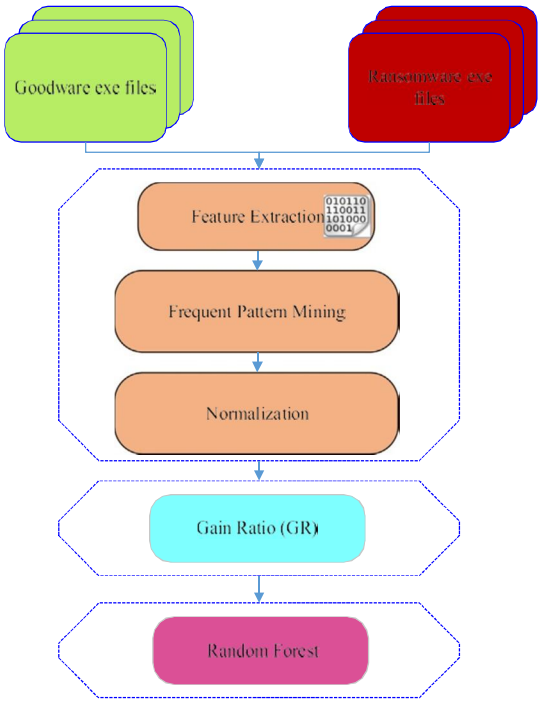
Random forest prediction is based on the majority vote of the combined predictions of multiple decision trees. First, the tree is determined and based on the best combination of variables, then the data set is dumped into subtrees of the tree. However, finding a combination of different variables simultaneously is not an easy task one).

III. DATASET

The dataset consists of 1680 executable files: 840 malware executable of different families, and 840 goodware files.

The Windows Portable Executable (PE32) ransomware files comprise three different families [9];(Cerber (267 samples), Tesla Crypt (315 samples), and Locky (258 samples)) which downloaded from Virus Total. The goodware files included two types of executable files; first type was collected from windows platform while the other type was collected from Portable Apps platform. Both ransomware and goodware are checked using Virus Total is a free tool that used to detect whether file is goodware or ransomware file.

The present method was implemented using computer of Core i7 CPU with 8 core, and 16 GB RAM with two systems; Windows 10, and Linux 4.1.



Evaluate the efficiency of our proposed method, as in following equations:

$$TRP = \frac{TP}{TP + FN}, \quad FRP = \frac{FP}{FP + TN},$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}, \quad Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$F - Measure = 2 * \frac{(Precision * Recall)}{Precision + Recall}$$

where: True Positive (TP): the number of ransomware that iscorrectly predicted as ransomware.
True Negative (TN): the number of goodware files that arecorrectly classified as goodware.
False Positive (FP): number of goodware files misclassifiedas ransomware.
False Negative (FN): number of ransomware which is mis-classified as goodware.

A. The effect of Features Dimension

To find the appropriate size for building classifiers, a set of features from 1000 to 7000 was tested with (1) number of seeds and a tree set of 100, because it found the price. too bad. Figure 2, Figure 3, and Table 1 show the accuracy of the classification operator, the machine learning process, and the classification confusion matrix, respectively. Figure 2 shows the difference between size and reality. From the results, it can be seen that the accuracy of 1000 dimensions is the best and reaches 97.74%. Meanwhile, Figure 2 shows that increasing the number of features does not improve the accuracy of classification. Figure 3 shows the ROC, recall, precision, and F-measure of the classification model. It is clearly seen that in case of 1000 dimensions the performance is best not only in terms of Recall but also in terms of Precision. For 1000 dimensions, the F1 index is higher than 97.8% and the ROC is about 99.6%. Table 1 shows the confusion matrix of this model and shows that the best values of FPR, FNR, TPR, and TNR for feature size 1000 are 0.043, 0.002, 0.998, and 0.957, respectively. Therefore, these results show that the best dimension to use in the classification model is the 1000 dimension. Therefore, all remaining tests will be of size 1000 features.

B. The effect of tree and seed numbers

Regarding the effect of the number of trees and seeds, this study attempted to increase the tree size from 10 to 1,000 and the seed size from 1 to 1,000. The process of this experiment is done by fixing the seed for a seed and changing the size of the tree from 10-1000 according to the elapsed time, as shown in Figure 4. This

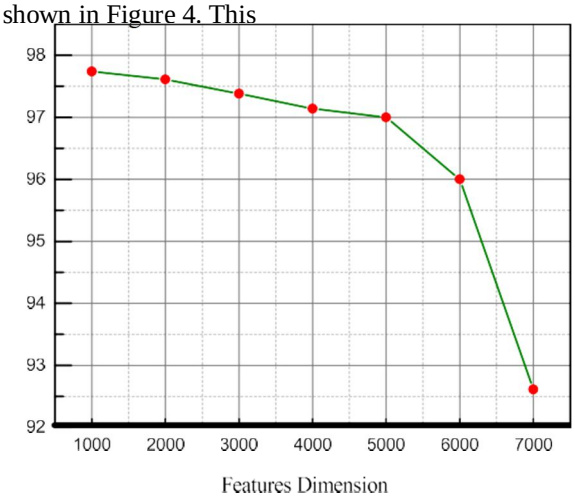


Fig. 2. The accuracy for different features dimension.

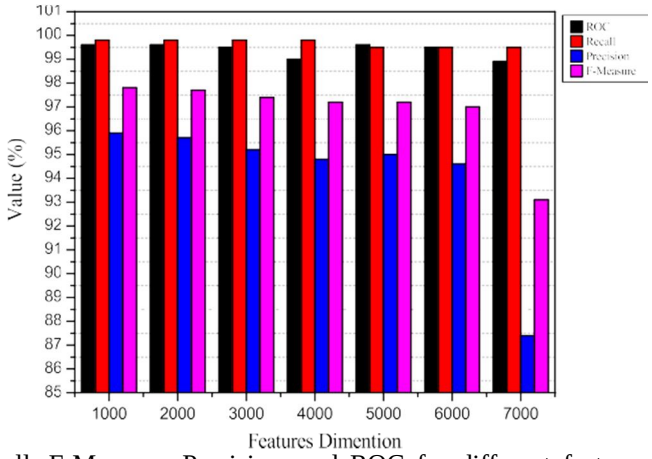


Fig. 3. Recall, F-Measure, Precision, and ROC for different featuresdimension.

The main issue here is which trees are best to provide high accuracy and time distribution.

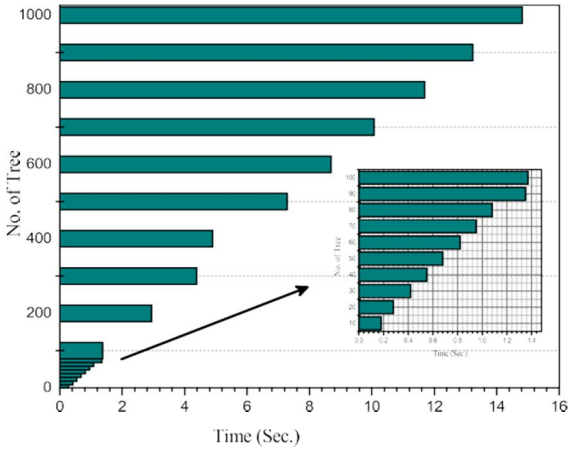


Fig. 4. The time for classification build using 1000 features diminution, the test dataset using different number of tree (10 to 1000) for the random forest classifier.

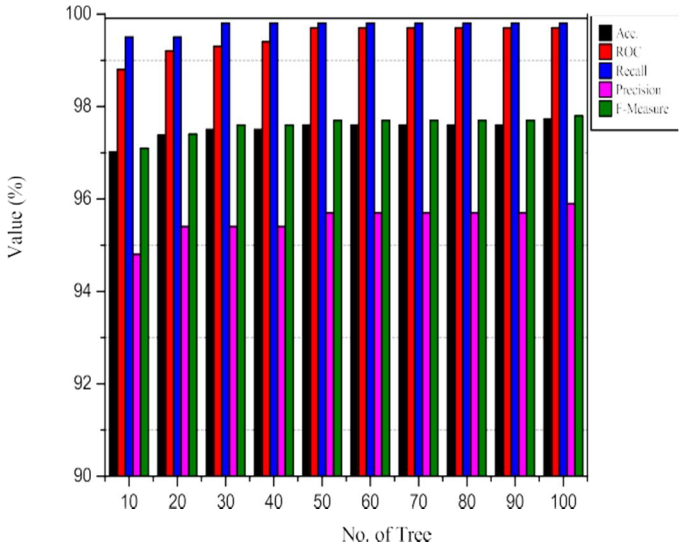


Fig. 5. Accuracy, Recall, F-Measure, Precision, and ROC for different tree number.

The number of trees (200 to 1000) is not included in these results as precision, recall, F-measure, accuracy, and ROC give the same results as 100 trees but again take more time. Effect on classifier performance. While the number of trees remains at 100, the number of seeds varies between 1 and 1000. The results show that when the number of seeds is 1, the most accurate result is 97.74%, as shown in Figure 6. works in most cases. The results show that the RF confusion matrix (FNR, FPR, TNR and TPR) and classification test models (accuracy, regression, precision, ROC and F-Measure) are better than Ada Boost M1 and Bagging, as shown in Figure 3 and Figure 7. It was also determined that the results of the Forest Competition were very close to the RF. But rotational forestry takes longer to develop structure

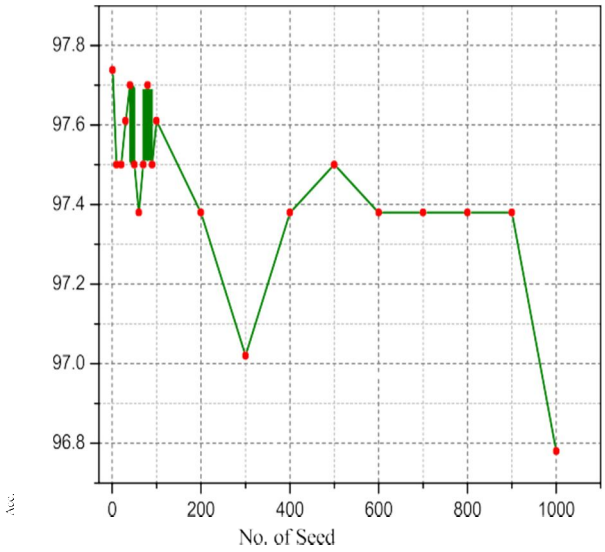


Fig. 6. The accuracy of the random forest classifier using different seednumber.

While the spin forest classifier takes about (25.63 seconds), RF only takes about (1.3 seconds) to build the model. Therefore, the RF classifier can still be considered more efficient in terms of time than the rotating forest. In this study, all data of 1680 samples were used using the 10-fold cross-validation method. This method reinstructs the operator on 90% of the information presented and analyzes the other 10%. Results are determined after 10 iterations using the average accuracy of each model. The test distribution standard and confusion results using 10-fold cross-validation are shown in Figure 8 respectively. uses opcode-based n-grams as features to represent them. This process requires programming the disassembler to achieve.

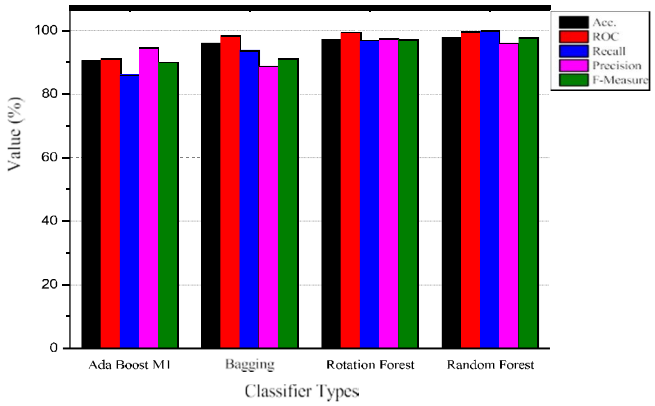


Fig. 7. The standard classification measures of different classifiers.

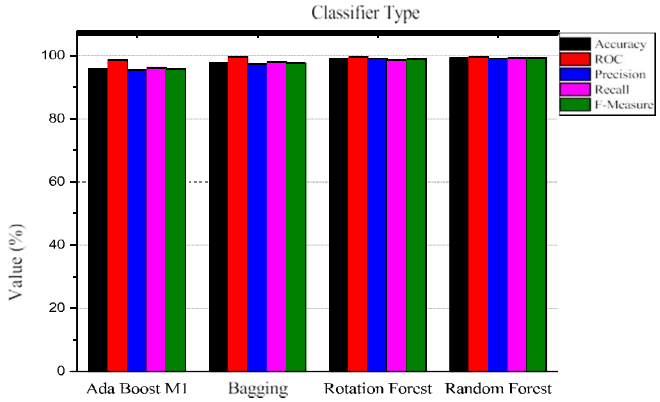


Fig. 8. The standard classification measures of different classifiers when10-fold cross validation has been used.

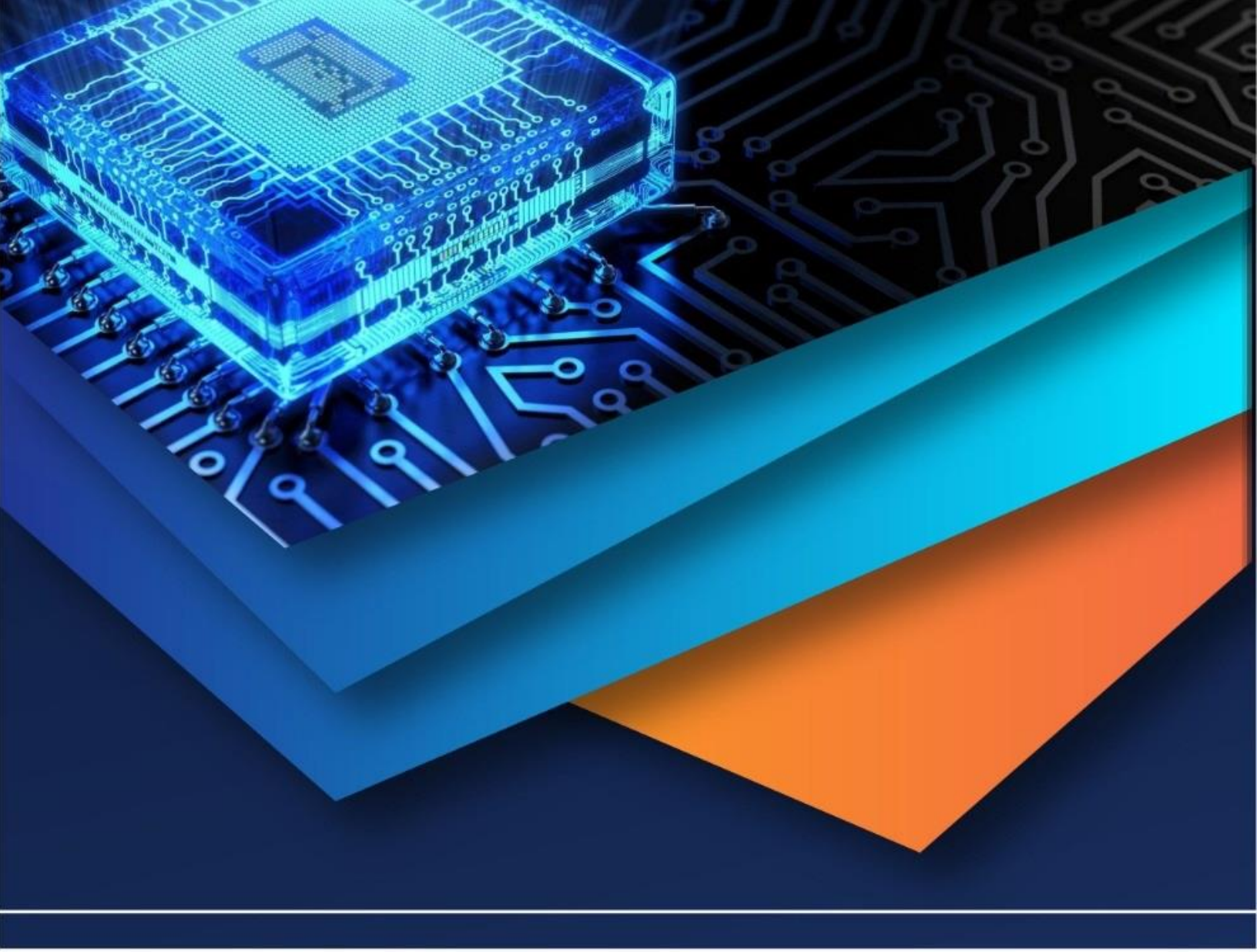
Opcodes are extracted from the data, and this method eliminates disassembly by extracting features directly from the raw data. Table 5 shows the comparison of the proposed method with Zhang et al. Technology. From the comparison, it can be seen that this research is more accurate than Zhang's prediction and the prediction time is shorter.

IV. CONCLUSION

This work presents a method based on machine learning techniques (random forest classifiers) to detect ransomware attacks. The current study tested different sizes of trees and seeds, such as 10-1000 and 1-1000 seeds, respectively.

REFERENCES

[1] H.J. Chittooparambil, et al., A review of ransomware families and detection methods, in: International Conference of Reliable Information and Communication Technology, 2018, pp. 588–597.
[2] I. Osterman Research, Understanding the depth of the global ran- somware problem, 2016, http://www.malwarebytes.com/pdf/whitepapers/Understanding The Depth Of Ransomware In theUS.pdf.
[3] P. Burnap, et al., Malware classification using self organising feature maps and machine activity data, Comput. Secur. 73 (2018) 399–410.
[4] X. Luo, Q. Liao, Awareness education as the key to ransomware prevention, Inf. Syst. Secur. 16 (2007) 195–202.
[5] M. Weckstén, et al., A novel method for recovery from crypto ransomware infections, in: 2016 2nd IEEE International Conference on Computer and Communications (ICCC), 2016, pp. 1354–1358.
[6] A. Kharaz, et al., {UNVEIL}: A large-scale, automated approach to detecting ransomware, in: 25th {USENIX} Security Symposium ({USENIX} Security 16), 2016, pp. 757–772.
[7] Y. Takeuchi, et al., Detecting ransomware using support vector machines, in: Proceedings of the 47th International Conference on Parallel Processing Companion, 2018, p. 1.
[8] R. Vinayakumar, et al., Evaluating shallow and deep networks for ransomware detection and classification, in: 2017 International Con- ference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 259–265.
[9] S. Homayoun, et al., Know abnormal, find evil: frequent pattern mining for ransomware threat hunting and intelligence, IEEE Trans. Emerg. Top. Comput. (2017).
[10] A. Tseng, et al., Deep learning for ransomware detection, IEICE Tech. Rep. 116 (2016) 87–92.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY