IBIMA Publishing
Journal of Information Assurance & Cyber security
http://ibimapublishing.com/articles/JIACS/2018/836339/
Vol. 2018 (2018), Article ID 836339, 13 pages
DOI: 10.5171/2018.836339

Research Article

An Efficient Mining Based Approach Using PSO Selection Technique For Analysis and Detection of Obfuscated Malware

Zafar Ali[1] and Tariq Rahim Soomro[2]

[1]Department of Computer Science, SZABIST Dubai Campus, Dubai, UAE

[2]College of Computer Science & Inf. Sys., IoBM, Karachi, Pakistan

Correspondence should be addressed to: Tariq Rahim Soomro; tariq.soomro@iobm.edu.pk

Received date: 23 November 2017; Accepted date: 27 December 2017;

Published date: 13 February 2018

Academic Editor: Meysam Namayandeh

Abstract

Malware plays a threatening role to the security of the data and information systems, as they created in different forms targeting data and networks. Malware developers use obfuscation techniques to hide malwares structure from detection of Anti-Virus (AV) programs, which use signature based detection; it is almost hard to detect the zero day attack and ineffective to analyze the hidden structure of malware. Such malicious codes are categorized as Oligomorphic, polymorphic and metamorphic Malware. Malware writers use packing mechanism to keep the malicious code harder during the signature-based detection and bypass easily. Mining techniques are one of the promising methods to analyze and detect hidden malware based on clustering and classification. This research focuses on improving accuracy and reducing processing time in the classification phase. This research approach mainly focused on the optimal attribute selection for classification to get the desired output. The proposed model uses Particle Swarm Optimization (PSO) for best attribute selection from the features set extracted from the packed and non-packed Portable Executable (PE) file format of malware and benign dataset. Classification tests have been prepared on the optimal subset of PE features in which Random Forrest classification outperforms from the rest of the classification algorithm.

Keywords: Malware, PSO, Obfuscation, Mining Techniques.

Cite this Article as: Zafar Ali and Tariq Rahim Soomro (2018), “An Efficient Mining Based Approach Using PSO Selection Technique For Analysis and Detection of Obfuscated Malware", Journal of Information Assurance & Cyber security, Vol. 2018 (2018), Article ID 836339, DOI: 10.5171/2018.836339

Introduction

The history of the Malware is not new and none of the platforms of services is safe from the Malware attacks whether it is Operating System, Custom made builds, Mobile Platforms etc. According to the Symantec Report of 2008, the number of malicious code and unwanted software may be exceeding the number of legitimate programs.

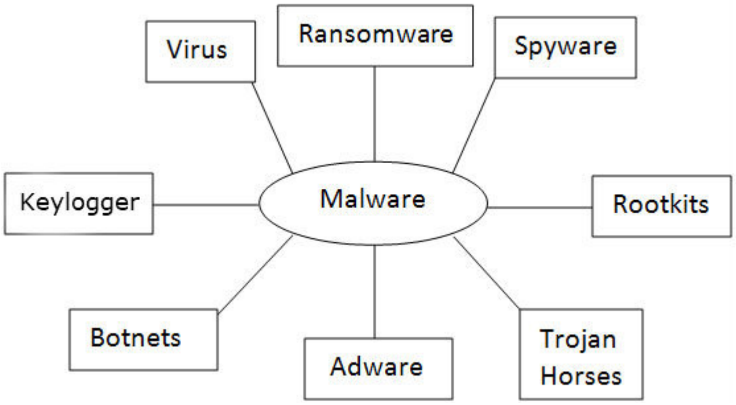


Fig 1.The figure shows malware types or classes of malware

The Malware consists of all types of Viruses, worms, Trojans, and bots are software called malware, as shown in Fig 1 above. Malware or malicious code (malcode) is short for malicious software. The core purpose of Malware code whether it is in the form of virus, worm, etc. is to damage, disrupt, steal, or some other "awful" or illegitimate activity on information, hosts, or systems. There are many types of classes of malware that have varying ways of infecting systems and propagating themselves. Sometimes Malware are combining with legitimate software to infect systems, or to attach as macros to files. Some are targeting and exploiting the vulnerability that exists in the system that can be in the custom based software or the operating system itself. Network devices are also prone to attacks because of the misconfiguration, such as a hole in a browser that only requires users to visit a Website to infect their computers. In the book, Masud et al. (2011) concluded that in most of the cases the users' actions are the ones causing the major drawback. The lack of knowledge e.g. when opening the email received from an un-trusted source; or browsing the Website, which contains malicious code available in the URL; or downloading the programs from un-trusted sites; or clicking the Ads, which contains illegal code; and many more. The commonly known types of malware are viruses, worms, Trojans, bots, back doors, spyware, and adware. Damages from malware are not limited to stealing confidential information, recording users' actions, and unavailability of services by way to Denial-of-Service (DoS) attack e.g. Distribute DoS (DDoS) and DoS such attack comes from the Botnets controlled by the Command and Control (C&C), destroying data, and compromising and/or entirely disabling systems and networks. It is hard to say that Malware cannot damage the physical hardware of systems and network equipment, but they almost target data and software residing on the equipment. Malware should not be confused with defective software, which is intended for legitimate purposes, but has errors or bugs that can harm the normal operation of the systems by having conflicts with system resources and kernel functions. Malware writers are targeting the Portable Executable (PE) to insert malicious code and then obfuscate PE format. In most of the cases malware writers' use packing tools or custom-build packed malicious code for infection in which the malicious code can be in the form of many variants, but the infection or action is the same. Now to fight against the malware attacks, Anti-Virus (AV) mechanisms are not the real time solution to detect and stop the zero-day attacks. Over the past few years, machine learning and mining based techniques are exploring and showing an effective approach in detection and analysis of malwares even with the zero-day attacks based on the behavior and features analysis of malwares. This research is focusing on the performance and accuracy of malware detection by selecting the best features using PSO technique and then applying classification of techniques. Random Forest (RF) shows 99.6% accuracy and outperforms as compared to other classification techniques applied on the selected features. This research paper is organized as follows; sections 2 will review the generation of malware along with obfuscation techniques; in section 3 the detection mechanism of malware and their types will be explored. In section 4 related works will be reviewed; in section 5 proposed approach will be discussed; in section 6 the experiment done in this research will be discussed; in section 7 results and findings of this study will be highlighted; and finally discussion and future work.

First & Second Generation of Malware

Malwares are of two generations, the first generation of the malware and the second generation of the malware. In the first generation, the structure of the malware does not change. While in the secondgeneration of the malware the internal structure of the malware changes in every variant while the action or the infection remains the same. This generation of malware uses obfuscation techniques to hide itself from the detection and do malicious activities as long as they reside in the system. The 2nd generation brings a huge responsibility on the white hat and the security companies to address the changing structure of the malware, which are hard to detect with the traditional signature based detection approach. The 2nd generation of the malware families use obfuscation techniques and they are further categorized as Encrypted, Oligomorphic, Metamorphic and Polymorphic Malwares.

Obfuscation Techniques

In general, the term obfuscation means making something harder to understand. In a programming point of view, it means to make written code harder to understand, which may reduce the vulnerability of being exploiting. The Obfuscation technique is adopted in the software industry to protect the copyrights of the legitimate software, now such techniques attracted the black hat hackers or the malicious code writer to obfuscate the malware families and produce new variants of the malware, which is harder for the signature based protection approach to detect the malware. Following are the various types of Malwares:

A. Encrypted Malwares

This was the first technique used in the 2nd generation malware development, it consists of two parts: the encryption body and the decryption code. Each time the Malware execute, it keeps the body unique and changes the key to hide its signature during detection. According to You and Yim (2010), the decryption process remains the same. The main motivation behind this technique is to avoid the malware detection done via static code analysis.

B. Oligomorphic Malwares

The encrypted malware creation let to the concealment of the Oligomorphic Malware by keeping a set of different decryptors. Usually decryptors from one variant to another are mutated. Signature based detection can be applied by keeping the signature for all the decryptors. According to the research of You and Yim (2010), signature based techniques in general are not effective on such types of malware to detect and provide protection.

C. Polymorphic Malware

In Polymorphic malwares, thousands and millions of decryptors can be created by modifying the instruction set of in the next variant of the malware to avoid signature-based detection. Polymorphic malware also consist of 2 parts performing the decryption of the body part. While executing this malware, the changing/mutation engine creates new malwares; which are combination with the encrypted malware body. Polymorphic malwares are created by using the obfuscation techniques, for example dead-code insertion, register reassignment, subroutine reordering, instruction substitution, code transposition/integration etc. Such challenges bring a question mark on the signature-based techniques in the process of detecting polymorphic malware.

D. Metamorphic Malwares

In the study of You and Yim (2010) Metamorphic malwares instead of generating new decryptors, they create new instances of the body without changing the infection behavior. Strong obfuscation techniques are used to deploy metamorphic malwares, which can harm not only the computers themselves, but the smartphones are the open target for such malware. As the vulnerability exploitation chances are more in smartphones, while the signature based detection mechanism is almost impossible, developing a true metamorphic malware is a challenging task and only few malware of this type are deployed. W32/NGVCK was created in 2001 with the help of Next Generation Virus Creation Kit (NGVCK).

Detection Mechanisms of Malware and their types

To provide defense against the threats/attack for the malware and its families; Anti-Virus programs are developed; most of the anti-virus programs are signature based keeping the assumption that the malware and its families are not changing its structure. However, due to the obfuscation techniques used in the 2nd generation of the malware, the signature based detection failed to capture malwares. Therefore software companies and academia are working hard to produce such techniques to combat against the 2nd generation of malware. The below overview on the signature based and heuristics based detection are presented to get some clear understanding of the detection approaches.

A. Signature based Detection

In the book authored by Masud et al. (2011), they mentioned that signature detection is the simplest and an effective way of detecting malware, which are known and can be easily identified. Once the malware sequences are identified, and features unique to the malware are extracted, which are then maintained in a signature database and should be manually updated on continuous basis, as new malware is discovered and analyzed. Signature-based malware detection generally enforces a static approximation of some desired dynamic (i.e., behavioral) security policy. Signature based detection always matches its signature patterns to the predefined policies of the binary executable and checking for any changes. However, due to the fact that polymorphic and the changing structure of the malware makes the signature-based detection less effective, an automated based detection can take over the detection approach of the malware variances.

B. Heuristic based Detection

This approach is considered as an effective technique for the detection of unknown malwares. The Heuristics method is a promising technique for the detection of encrypted malware; however such detection mechanism needs a completely sandbox environment to carry out the test. While heuristic technique can be combined with the machine learning techniques to get better results for detection.

C. Machine Learning Techniques

In recent times the machine learning techniques are gaining popularity not only in terms of unknown and known malware detection, but also learning from the environment, which may detect zero day attacks as shown in fig 2 below. The popular machine learning techniques as per the Masud et al. (2011) are Navie Bayes, Decision Tree, Support Vector Machine, Neural Networks, hidden Morkov modes. Such techniques are providing real time detection once trained. Such techniques may not be suitable for the end users, but can be deployed at the enterprise gateway level as such techniques are costly and computationally expensive.

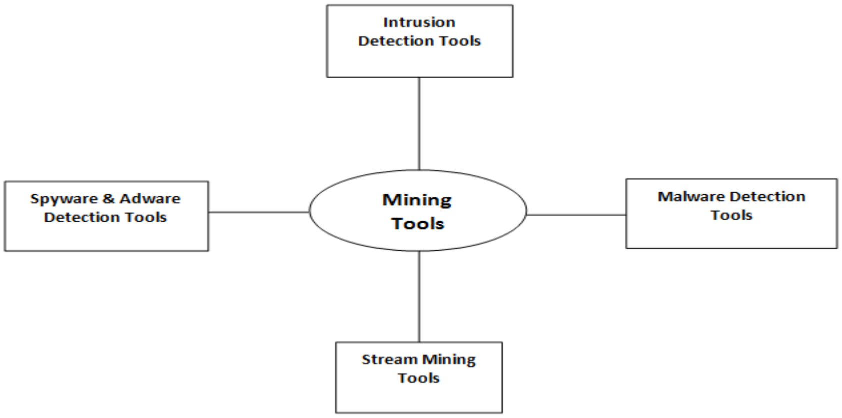


Fig. 2: Data Mining tools for malware detection

Related work

Perdisci et al. (2008) extracted 9 important features of the PE file format header and sections areas, which are based on the static information of the PE file; and using pattern recognition and threshold value to detect the packed files from the non-packed files. However, only files, which are packed, can be sent to the universal unpacker tool for further analysis and the non-packed files are sent directly to the anti-virus tool. This reduces time and improves efficiency of the analysis.

The proposed solution of Komashinskiy and Kotenko (2010) is focusing on the processing of position ally dependent of static information, such static information builds a model for detection of the elements gathered from the static information,, which element can be explored in the future by combining with already existing methods or doing a modular approach based on object oriented analysis.

In the study of Elhadi et al. (2012), the authors have addressed the problem that malware writers are using packing and obfuscation techniques to make the signature based detection impossible to catch; which also results zero-day detection failure. The aim of their research is to improve accuracy and the scanning time of the malware detection. The authors proposed a framework for malware detection using signature based and behavior based call graph. A hybrid model was developed based on signature and behavior based call graph while doing the static and dynamic analysis.

The proposed solution mentioned by Faruki et al. (2012) presented a behavioral model, which represents the abstraction of the binary by analyzing the API string available in the windows PE file. The focus is based on extracting the temporary snapshots of legitimate and malicious code executables, which are known as API call-grams.

The solution presented by Singhal and Raul (2012) is providing a model for protecting the enterprise network at firewall level. The presented models, extracting the PE header files for infected and benign code by using the Import Address Table (IAT), extracted various API call and stored them in a data mine repository, then Information Gain (IG) is calculated for each function. Then Random Forest classifier has been used for classification, which is working on a combination of decision tree predictors.

Wang et al. (2012) proposed prototype, which abstract the character of the malware by analyzing malware sample. This behavior semantics uses taint analysis approach; the critical behaviors are abstracted and graph based on data and control dependency is constructed and reconstructed using behavior logic of the malware. The proposed systems collect characters are adopted to detect malware variants, which reduces the delay between the new malware variants and features updating of the malwares.

In the research study by Cervante et al. (2012), feature selection filter based was approached. It is using Particle Swarm Optimization (PSO) to achieve performance, feature selection approach discard irrelevant and redundant features.

The research study of Saxe et al. (2013) proposed a Web mining technical documentation to identify malware automatically. The symbols used in the malware created are mapped, and extracted from the technical posts available on the Web. A simple algorithm for creating function relationship graph for malware samples from Web technical documents. This approach for detecting malware is a new direction instead of using machine learning (ML) or Data Mining classifiers, keeping in mind mining of the web knowledge.

Ding et al. (2013) proposed selection criteria for association rule and system calls for extra burden of processing of useless system call and association rules, which have no classification power for effective detection. Furthermore, the authors have proposed multiple association rules for classification accuracy for executable classification, in this case the result has been compared for classification accuracy rate.

In a research study by Markel and Bilzor (2014), machine learning classifiers have been applied to train the classifier on metadata of windows executable file for extracting valuable features to classify normal and malicious code using different machine classifiers to find the best classifier out of them. Test has been carried out for comparison purpose in which decision tree was used as compared to Naïve Bayes and logistic regression classifier.
Yadav et al. (2014) proposed an algebraic signature based technique for detecting all types of deployed malware available on the web acting as malicious URL for malware spreading.

A comparison for detection of different techniques has been introduced, which shows that the proposed system of Gunalakshmii and Ezhumalai (2014) for detection improves protection and confidentiality in smartphones. Support

Vector Machine (SVM) has been proposed in the architecture of this project, which observes the feature based on permission and analysis for categorizing weather the application is normal application or key logger application.

The proposed solution of Bai et al. (2014) uses format information of PE file by mining and in-depth static analysis of PE file. In this paper all the features of the PE files were extracted for the PE header file in which the classification method has been applied to reduce the dimensionality and enhance completeness of the features for accuracy. The selected features were trained using classification algorithm to distinguish between the benign software and malware.

Shi et al. (2014) presented a lightweight method, the growing hierarchical self-organizing map (GHSOM) for malwares detection and structural classification. This method provides the structural malware classification, and measures the similarities of structural classes that belong to different trees. The mining is performed on the DLL windows file for the detection of malware, which is being one of the targets of malware writers.

According to Wang and Wang (2015), behavior based malware detection in automatic based malware detection for unspecified malware using support vector machine to train the classifier, based on behavioral signatures, while a cross validation scheme is used for classification accuracy, as there are classification errors for generalization of detection. In addition, the detection mechanism of other machine learning techniques produces less accurate classification.

Chen et al. (2015) used file relation graph for malware detection and introduces novel belief propagation (BP) algorithm. Based on the previous detection techniques and the performance of the detection of mining and machine learning techniques, which may not be adequate for classifying and extracting all the features of malware, the proposed relation graph and the novel belief propagation classifier would be used for effectiveness and the accuracy of catching the malware variants.

Chemchem and Drias (2015) addressed the importance of performance and improvement in speeding up the process of reasoning engine. Their work used data mining techniques for searching interesting patterns, which led to the development of K-NN-IR and K-means-IR which is based on induction rules. The new Architecture called the miner intelligent agent is tested and evaluated on public large scale benchmark, which includes 25000 induction rules. Tests have been carried out by comparing with the classical cognitive agent in this MIA outperform in terms of performance.

Proposed Model

This research is focused on qualitative research approach as extensive literature review has been done along with experimental approach. This approach is to reduce the 9 PE features of Packed and non-packed files explained by Perdisci et al. (2008) and to get the same or better classification accuracy and to reduce processing time. A filter based Particle Swarm Optimization (PSO) approach by Cervante et al. (2012) is used for feature selection, which removes the redundant, irrelevant and noisy features from the extracted features. The authors proposed and extended the fellow of experiment used in the research study by Perdisci et al. (2008), as shown in Fig. 3 below. Feature selection plays an important role especially on a large scale, when processing time and improving efficiency really matters. This approach has minimized the number of features by selecting the optimal features from the set extracted from the PE File used by Perdisci et al. (2008), during pre-processing in this approach. Here filtered based Particle Swarm Optimization (PSO) has been used for attributes/features selection approach, of Cervante et al. (2012), and then classification techniques have been used. For example, decision tree (J48); Neive Bayes; Random forest; IBK, Support Vector Machine (SVM); Logistic Regression; and Multilayer Perception are applied on the optimal subsets of the attributes, which shows better results and the processing time may certainly improve, and False Positive Alarm on a Random Forest (RF) is extremely reduced on the minimized features set.

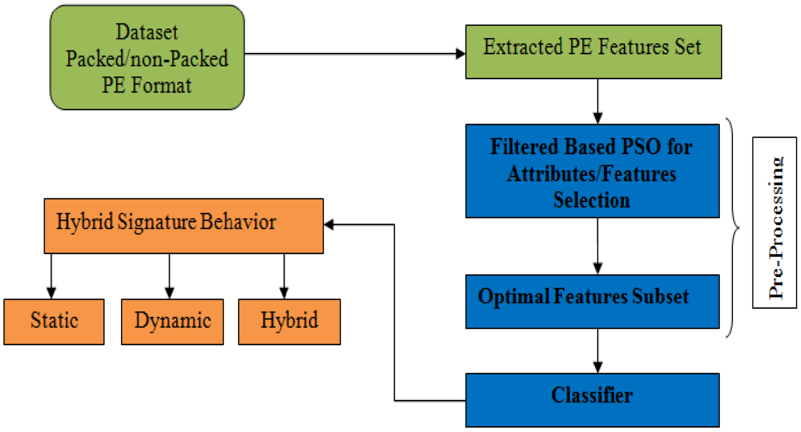

Dataset
Packed/non-Packed
PE Format
Extracted PE Features Set
Filtered Based PSO for Attributes/Features Selection
Optimal Features Subset
Classifier
Pre-Processing
Hybrid Signature Behavior
Static
Dynamic
Hybrid

Fig. 3: Proposed Model

Once the classification phase is based on both packed and non-packed executables, these are forwarded to the next phase of our model, which is part of the model presented by Elhadi et al. (2012), for further detection of the malware and benign files. Non-packed malware/benign will not take enough time and will be passed from anti-virus signature part of the Hybrid Signature Behavior Based Phase, while packed malware and benign will be checked for malicious code.

Experiment

Free dataset (training and test) have been downloaded and used in this study from the Website http://roberto.perdisci.com/projects/cpexe. It contains 9 PE features used by Perdisci et al. (2008), extracted from the packed and non-packed malwares, with the total of 5,498 executables, which contains 2,598 packed computer virus, 2,231 benign executables of clean Win XP executables and 669 are packed benign executables, which were packed with freely available packing and wrapping tools. Packed computer viruses were collected from Malfease Project discussed in Perdisci et al. (2008). The features' extraction was based on the information gain calculation of the entire PE file, data sections, code sections and PE header. While the number of standard and non-standard sections, executable, IAT, Read and Write sections are also considered for PE file. As mentioned by Perdisci et al. (2008), the information gain is always high of the packed files due to byte randomness and abnormalities in the function calls. A virtual machine has been installed and runs on a windows XP image on Intel based machine. WEKA (Waikato Environment for Knowledge Analysis), a data mining tool has been used to perform the features subset selection. Once the installation is done for WEKA, the study used PSO attribute selection for features/attribute selection. The PSO attribute selection package is separately installed from the package manager of the WEKA tool. The training dataset mentioned above downloaded from author's website has been loaded to WEKA in arff format, which is the native format for WEKA application to load the dataset and make it available for pre-processing phase. In the pre-processing step, PSO attribute selection is applied by using the filter subset approach. The features/attributes are minimized with the help of PSO attribute selection by selecting the optimal 5 features/attributes as a subset from the 9 features/attributes. The resulted features are standard sections (SS), Executable Sections (ES), Import Address Table (IAT), Code Entropy (CE) and File Entropy (FE). In the classification phase, the following classifiers are used with the 10 fold cross-validation:

- Neive Bayes (NB)
- Support Vector Machine (SVM)
- Multilayer Perception (MLP)
- IBK (Instance Based Classifier)
- Decision tree (J48)
- Random Forest (RF)

Results and Finding

Based on the experiment, the below Table 1 shows different classifiers for the below weighted average statistics on the supplied dataset using 10-fold cross validation on the reduced features set performed in the pre-processing with the help of filtered based PSO attribute selection. Table 1 shows that Random Forest classifier out-performs the rest of the classifiers in terms of all the classification results obtained as RF receive 99.6% accuracy of True Positive Rate (TPR). Receiver Operating Characteristics (ROC) curves are also produced for the packed files, which shows True Positive Rate (TPR) against the False Positive Rate (FPR). Here True Positive Rate (TPR) refers to the classifier evaluation of correctly detecting malware instances divided by the total number of malware files. Whereas False Positive Rate (FPR) refers to the number of benign executable misclassified as malware over the total number of benign files.

Table 1: Classifier results and finding from optimal features of PE files (weighted average)

	TP-Rate %	FP-Rate %	Precision %	Recall %	F-Measure %	MCC %	ROC Area %	PRC Area %
NB	97.6	2.4	97.7	97.6	97.6	95.3	99.0	98.9
SMO	97.8	2.2	97.8	97.8	97.8	95.5	97.8	96.7
MLP	98.5	1.5	98.5	98.5	98.5	97.1	99.7	99.7
IBK	99.5	0.5	99.5	99.5	99.5	98.9	99.4	99.2
J48	99.4	0.6	99.4	99.4	99.4	98.8	99.2	98.6
RF	99.6	0.4	99.6	99.6	99.6	99.1	100.0	100.0

In the below figures from 4 to 9, Receiver Operating Characteristics (ROC) curves are also produced for the packed files, which shows True Positive Rate (TPR) against the False Positive Rate (FPR).

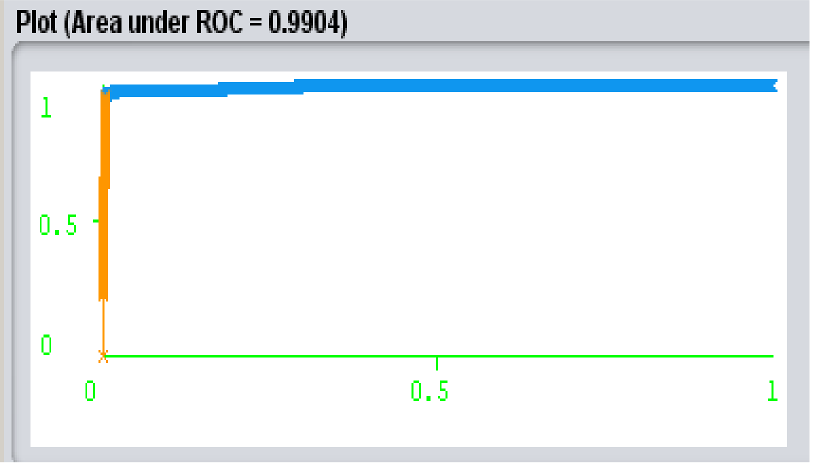


Fig. 4: NB ROC Curve for Packed Files

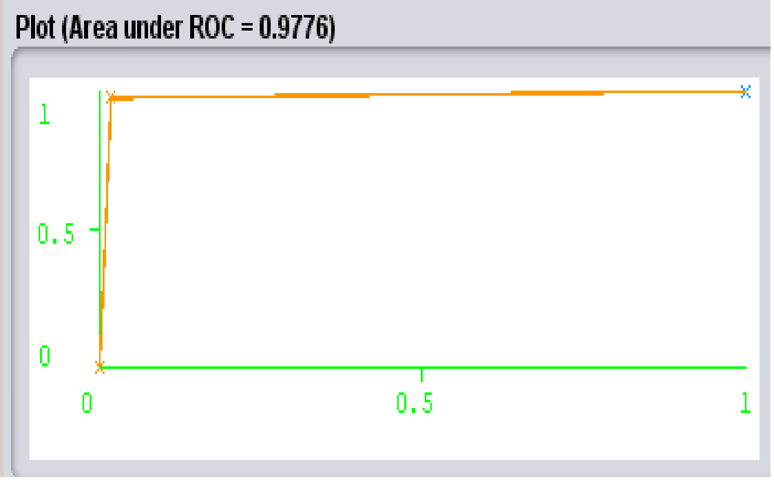


Fig. 5: SMO ROC Curve for Packed File

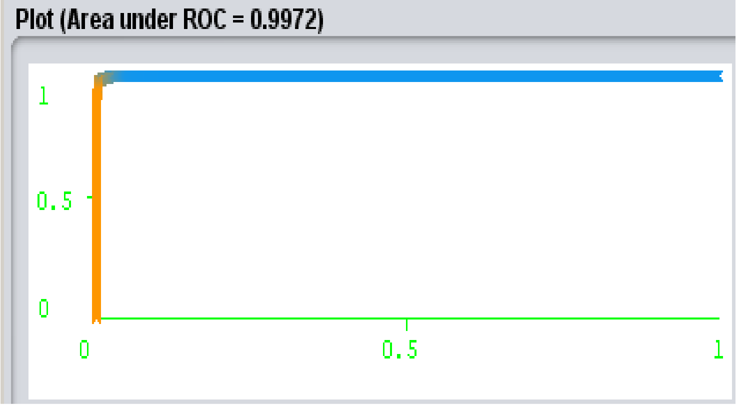
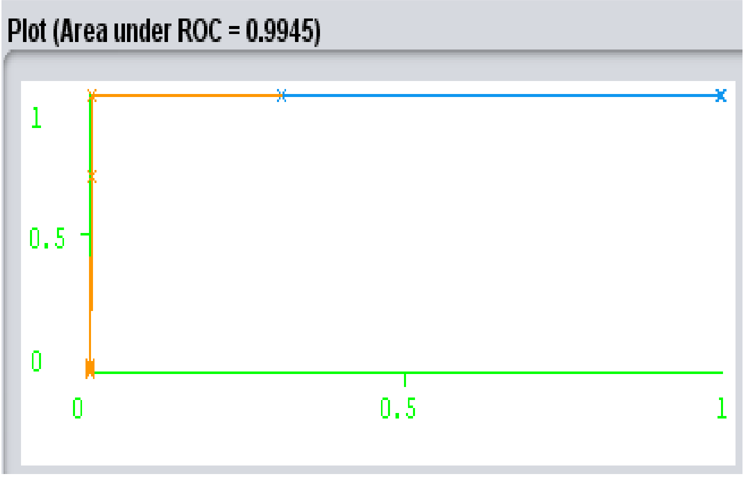
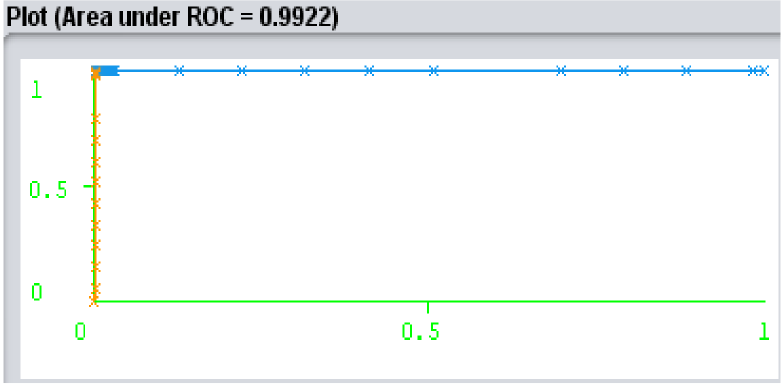


Fig. 6: MLP ROC Curve for Packed Files



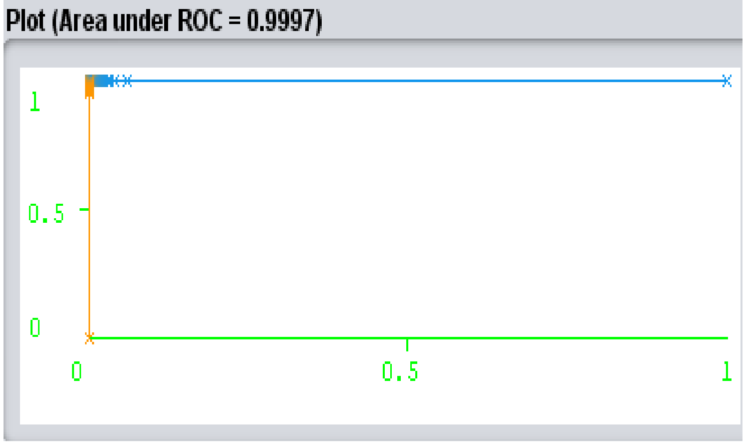
Plot (Area under ROC = 0.9945)

Fig. 7: IBK ROC Curve for Packed Files



Plot (Area under ROC = 0.9922)

Fig. 8: J48 ROC Curve for Packed Files



Plot (Area under ROC = 0.9997)

Fig. 9: RF ROC Curve for Packed Files

Discussion and Future Work

This research is focused on the detection of obfuscation malware via mining techniques. An extensive study has been done, with the help of literature review and available tools and it is figured out that malicious code is packed to hide from the signature-based detection and can be easily propagated into the system. This research approach identified PE files features of packed files whose entropy values are always high due to abnormalities and randomness in the IAT (Import Address Table) and other sections of PE files. This research approach is focused on classification accuracy and reducing the processing time, for this reason attribute selection is performed using filtered based PSO to get the optimal subset of PE features during the pre-processing of the proposed model to remove the noisy, irrelevant and redundant features from the PE features extracted from the packed and non-packed files. The results are carried-out using different mining classification techniques in WEKA tool in which Random Forest (RF) outperforms the rest of the classifiers obtaining 99.6% accuracy of True Positive Rate on the reduced features set. The classified files are forwarded to the next phase of our model which is the part of the model presented in Elhadi et al. (2012). The current dataset contains 5,498 instances of packed and non-packed files with 9 features set with class attribute mentioned in Perdisci et al. (2008), which is reduced to 5 features with proposed model; this gives a very good classification accuracy especially on Random Forest and the resultant classified sets are forwarded to the next phase of Hybrid behavior signature phase presented in Elhadi et al. (2012). In the future, researchers will try maximizing the number of instances on a large scale with some diverse dataset of malware and benign files and will try to check the performance of accuracy and processing time on these reduced features of the dataset. Furthermore, authors consider the Hybrid behavior signature phase for generalization of the packed and non-packed files so that correctly classified non-packed files should pass from the signature phase and packed files are further analyzed, for their structural and behavior characteristics for more useful patterns of zero-day attack.

References

1. Bai, J., Wang, J. and Zou, G. (2014), 'A malware detection scheme based on mining format information,' The Scientific World Journal, 2014.

2. Cervante, L., Xue, B., Zhang, M. and Shang, L. (2012), 'Binary particle swarm optimisation for feature selection: A filter based approach,' Evolutionary Computation (CEC), IEEE Congress on, 2012. IEEE, 1-8.

3. Chemchem, A. and Drias, H. (2015), 'From data mining to knowledge mining: Application to intelligent agents,' Expert Systems with Applications, 42, 1436-1445.

4. Chen, L., LI, T., Abdulhayoglu, M. and YE, Y. (2015), 'Intelligent malware detection based on file relation graphs,' Semantic Computing (ICSC), IEEE International Conference on, 2015. IEEE, 85-92.

5. Ding, Y., Yuan, X., Tang, K., Xiao, X. and Zhang, Y. (2013), 'A fast malware detection
6. algorithm based on objective-oriented association mining,' computers & security, 39, 315-324.

7. Elhadi, A. A. E., Maarof, M. A. and Osman, A. H. (2012), 'Malware detection based on
8. hybrid signature behaviour application programming interface call graph,' American Journal of Applied Sciences, 9, 283.

9. Faruki, P., Laxmi, V., Gaur, M. S. and Vinod, P. (2012), 'Behavioural detection with API call-grams to identify malicious PE files,' Proceedings of the First International Conference on Security of Internet of Things, 2012. ACM, 85-91.

10.Gunalakshmii, S. and Ezhumalai, P. (2014), 'Mobile keylogger detection using machine

11.learning technique,' Computer Communication and Systems, International Conference on, 2014. IEEE, 051-056.

12.Komashinskiy, D. and Kotenko, I. (2010), 'Malware detection by data mining techniques based on positionally dependent features,' Parallel, Distributed and Network-Based Processing (PDP), 18th Euromicro International Conference on, 2010. IEEE, 617-623.

13.Markel, Z. and Bilzor, M. (2014), 'Building a machine learning classifier for malware detection.'Anti-malware Testing Research (WATeR), Second Workshop on, 2014. IEEE, 1-4.

14.Masud, M., Khan, L. and Thuraisingham, B. (2011), Data mining tools for malware

15.detection, CRC Press.

16.Perdisci, R., Lanzi, A. and LEE, W. (2008), 'Classification of packed executables for accurate computer virus detection,' Pattern Recognition Letters, 29, 1941-1946.

17.Saxe, J., Mentis, D. and Greamo, C. (2013), 'Mining web technical discussions to identify

18.malware capabilities,' Distributed Computing Systems Workshops (ICDCSW), IEEE 33rd International Conference on, 2013. IEEE, 1-5.

19.Shi, H., Hamagami, T., Yoshioka, K., Xu, H., Tobe, K. and GOTO, S. (2014), 'Structural

20.classification and similarity measurement of malware,' IEEJ Transactions on Electrical and Electronic Engineering, 9, 621-632.

21.Singhal, P. and Raul, N. (2012), 'Malware detection module using machine learning

22.algorithms to assist in centralized security in enterprise networks,' arXiv preprint

23.arXiv:1205.3062.

24.Wang, P. and Wang, Y.-S. (2015), 'Malware behavioural detection and vaccine development by using a support vector model classifier,' Journal of Computer and System Sciences, 81, 1012-1026.

25.Wang, R., Jia, X. and Nie, C. (2012), 'A Behavior Feature Generation Method for Obfuscated Malware Detection,' Computer Science & Service System (CSSS), International Conference on, 2012. IEEE, 470-474.

26.Yadav, R., Baghel, R. and Tomar, D. S. (2014), 'An Approach to Detect Malware Snippets,'International Journal of Innovative Research and Development, 3.

27.You, I. and Yim, K. (2010), 'Malware obfuscation techniques: A brief survey,' International conference on broadband, wireless computing, communication and applications, 2010. IEEE, 297-300.