OBFUSCATED COMPUTER MALWARE CLASSIFICATION BASED ON SIGNIFICANT OPCODE

YU CHII HENG

UNIVERSITI TEKNOLOGI MALAYSIA

OBFUSCATED COMPUTER MALWARE CLASSIFICATION BASED ON SIGNIFICANT OPCODE

YU CHII HENG

A project report submitted in fulfilment of the
requirements for the award of the degree of
Master of Engineering (Computer and Microelectronic Systems)

School of Electrical Engineering
Faculty of Engineering
Universiti Teknologi Malaysia

JULY 2022

ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my supervisor Dr. Ismahani Ismail for her guidance, advices and motivation. Without her continued support and interest, this thesis would not have been the same as presented here.

My sincere appreciation also extends to all my colleagues and others who have provided assistance at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family member.

ABSTRACT

Computer malware has greatly impacted the computer network securities and even personal computer users. Signature-based detection is incapable to recognize the obfuscated computer malware since it is being covered by the obfuscation techniques. Therefore, machine learning is being explored and equipped in the malware detection to withstand the threaten of malware. In fact, there are many features available, i.e., text string to be implemented for malware classification. Nevertheless, opcode could be one of the features owing to its relative smaller data size compared to the text string. In this project, the significant opcodes from the executable malware files are extracted and several machine learning classifiers are compared in terms of classification accuracy and speed, as well as the comparison is done with text string-based detection and signature-based detection. Only significant opcodes are extracted from the malware assembly code whereas the obfuscated malware code is used as testing dataset to observe the performance of classifier models. From the finding, machine learning classification using significant opcode is able to detect obfuscated malware with less time taken as compared to text string feature.

ABSTRAK

Malware telah menyebabkan kesan yang teruk bagi rangkaian securiti komputer dan juga komputer persendirian. Manakala, teknik pengesanan malware tradisional tidak dapat mengenalkan malware yang telah disamarkan dengan teknik penyamaran. Teknik penyamaran boleh mengubahkan kod binary tanpa menpengaruhi fungsi asal malware. Seterusnya, komputer malware yang berkembang dengan cepat akan menyebabkan stor data tandatangan malware tidak dapat menyimpan tandatangan yang terkini. Teknik pembelajaran mesin telah dilengkapkan bagi membantu pengesanan malware mengenalkan malware yang telah disamarkan. Malah, terdapat banyak ciri yang tersedia untuk melatih pengelas pembelajaran mesin. Rentetan teks adalah salah satu ciri biasa yang dilaksanakan untuk pengesanan malware. Namun begitu, kod operasi juga boleh menjadi salah satu penggantian rentetan teks kerana saiz datanya yang lebih kecil berbanding dengan rentetan teks. Kod operasi yang penting dari fail perlaksana telah diekstrakkan untuk melatih pengelas pembelajaran mesin. Manakala, malware yang disamarkan telah digunakan sebagai data pengujian untuk menguji pretasi pengelas pembelajaran mesin. Akhirnya, pengelas pembelajaran mesin dapat mengenalkan malware komputer yang disamarkan dengan kod operasi dalam masa yang lebih cepat berbanding dengan rentetan teks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

SGD	-	Stochastic Gradient Descent
GD	-	Gradient Descent
SMO	-	Sequential minimal optimization
API	-	Application Programming Interface
LDA	-	Latent Dirichlet Allocation
TF-IDF	-	Term Frequency Inverse Document Frequency
MLP	-	Multi-layer Perceptron
SVM	-	Support Vector Machine
GMDH	-	Group method of data handling
DT	-	Decision Tree
RF	-	Random Forest
*k*NN	-	*k*-Nearest Neighbors
XML	-	Extensible Markup Language
ARFF	-	Attribute-Relation File Format
PE	-	Portable Executable
IG	-	Information Gain
DLL	-	Dynamic-link Library
RIPPER	-	Repeated Incremental Pruning To Produce Error Reduction
QP	-	Quadratic Programming
ARFF	-	Attribute-Relation File Format

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Malware is a malicious software that could hijack the infected computer or system and disabling its firewall via the network connection. Based on the statistic report that made by Google, 70% of the malwares are discovered from the well-known websites. The computer could just be infected by opening the website if the website is infected by the malware [1]. According to the report by Accenture, the estimated financial loss that caused by the malware attack is around $2.6 million [2]. Based on the documented history, the earliest recorded malware was found in 1970s. The malware that first discovered was Creeper Worm which capable of self-replication and perform remote access from the attacker terminal. Looking back to the history of the computer, malware threat was coming along with the born of computer. Malware could be categories into many types which are virus, worm, trojan house, rootkit, spyware, adware, botnet, keylogger, ransomware and so on.

The threats of the malware were never ended with the existence of the anti-malware software. The malware threats have influenced and impacted not only the field of computer but also the field that required the computation with computer. The malware has evolved so that they could hide or cover themselves from being detected by the anti-malware software. Furthermore, the computer malware is evolving constantly where the outdated malware signatures database would barely to include all the latest malwares. The obfuscated computer malware is a malware that able to change its binary code while

preserving the malware functionality so that it would not be detected by the anti-malware software. Much more advanced obfuscation techniques have been invented by the hackers to protect their malwares from being captured. In order to detect the malware and protect the computer or system being attacked by the hackers, machine learning is equipped to the malware detection methodology to enhance the detection ability [3].

According to the study of [4], its result proven that by using machine learning could enhance the robustness in malware detection application. In the research [5], text string feature was involved to train machine learning classifier. Text string was selected in this research owing to its informative and small memory size. Instead of using text string as the features to train the machine learning classifier, other features such as byte code and opcode are available for this purpose. In research [6], text string is also used as the feature to train and test the machine learning classifier for detecting obfuscated malware.

New malware can carry some prevalent content from the previous malware. Based on this hypothesis, this work is proposed where the significant opcode is referring to the prevalent content. In this proposal, opcode is chosen because of it relatively smaller data size and significant opcode is proposed to be used as the features to replace text string. Opcode are extracted from the assembly code, that originally from the malware executable files.

1.2 Problem Statement

In [7], most of the anti-virus software vendor could have excess of 200 million malware signatures stored in their database and keep on growing by 2 to 3 million per

month. The hacker who with the skill set of anti-virus technology would be able to generate the malware that could escape the detection of signature-based anti-virus software. Signature based detection approach is incapable to recognize the obfuscated computer malware when the signatures that could be found in the signature database. Hence, the signature-based detection that highly relying on the frequent signatures update by the vendor could be a vulnerable to malware attack [8]. The implementation of machine learning on computer malware detection could be able to overcome the limitation of signature-based detection technique. There are several features available to train the machine learning classifier to recognize the malware, such as byte code, opcode, and text string. However, byte code and text string are relatively larger than opcode in term of data size [5]. Therefore, the opcode is chosen as the primary feature to obtain the information from it.

1.3 Objectives

1. To extract significant opcode from malware executable files.
2. To compare machine learning classifiers which are able to detect the malware based on the significant opcode in term of accuracy and speed.
3. To classify the obfuscated malware based on the best classifier chosen.
4. To compare the result with text string-based detection and signature-based detection.

1.4 Scope of the research

First, this research is mainly focus on the computer malware detection where the other types of malwares would be excluded. Next, the feature that would involve in the research is significant opcode neither byte code nor text string. Based on the chosen feature, it will be extracted from the assembly code and is used to train the classifiers. The best classifier in term of speed and classification accuracy would be chosen to compare with the classifier that trained with text string classifier. Among several common malware analysis techniques, static analysis is chosen as the primary analysis technique in this research to detect the malware. In addition, the supervised machine learning classifier with two classes classification is being implemented in the research.

1.5 Thesis organization

In this thesis, five chapters are organized and arranged in order and clearly as follow. In chapter 2, the background knowledge and literature review that related to the research topic will be discussed and elaborated. Then, the research methodology will be clearly explained in chapter 3. Next, the result and finding of the research would be presented in chapter 4. At last, the conclusion of the research will be demonstrated in chapter 5.

REFERENCES

[1] R. Rehman, D. G. C. Hazarika, and G. Chetia, "MALWARE THREATS AND MITIGATION STRATEGIES: A SURVEY," *J Theor Appl Inf Technol*, vol. 31, no. 2, 2011, Accessed on: Dec. 20, 2021. [Online]. Available: www.jatit.org

[2] A. Darem, J. Abawajy, A. Makkar, A. Alhashmi, and S. Alanazi, "Visualization and deep-learning-based malware variant detection using OpCode-level features," *Future Generation Computer Systems*, vol. 125, pp. 314–323, Dec. 2021, doi: 10.1016/j.future.2021.06.032. Accessed on: Dec. 21, 2021. [Online]. Available: https://www.sciencedirect.com20/science/article/abs/pii/S0167739X21002272

[3] D. Gibert, C. Mateu, and J. Planes, "The rise of machine learning for detection and classification of malware: Research developments, trends and challenges," *Journal of Network and Computer Applications*, vol. 153. Academic Press, Mar. 01, 2020. doi: 10.1016/j.jnca.2019.102526. Accessed on: Dec. 21, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804519303868

[4] W. Fleshman, E. Raff, R. Zak, M. Mclean, and C. Nicholas, "Static Malware Detection & Subterfuge: Quantifying the Robustness of Machine Learning and Current Anti-Virus." Accessed on: Dec. 21, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804519303868

[5] T. H. Xin, I. Ismail, and B. M. Khammas, "Obfuscated computer virus detection using machine learning algorithm," *Bulletin of Electrical Engineering and Informatics*, vol. 8, no. 4, pp. 1383–1391, Dec. 2019, doi: 10.11591/eei.v8i4.1584. Accessed on: Dec. 17, 2021. [Online]. Available: https://www.beei.org/index.php/EEI/article/view/1584

[6] Ahmed Ali, Mohammed Hasan Ali, "Metamorphic malware detection using machine learning". Master's thesis, Faculty of Engineering - School of Electrical Engineering, 51 Universiti Teknologi Malaysia, 2020. Accessed on Dec 02, 2021. [Online]. Available: http://eprints.utm.my/id/eprint/93122/

[7] "Is Anti-virus Really Dead?". Computers and Security, vol. 44. Elsevier Ltd, p. iv, Jul. 01, 2014. doi: 10.1016/S0167-4048(14)00082-0. Accessed on: Dec.

15, 2021. [Online]. Available: http://dx.doi.org/10.1016/S0167-4048(14)00082-0.

[8] C. Wressnegger, K. Freeman, F. Yamaguchi, and K. Rieck, “Automatically inferring malware signatures for anti-virus assisted attacks,” in ASIA CCS 2017 – Proceedings of the 2017 ACM Asia Conference on Computer and Communications Security, Apr. 2017, pp. 587–598. doi: 10.1145/3052973.3053002. Accessed on: Nov. 30, 2021. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3052973.3053002

[9] “What is Opcode_”. Accessed on: Jan. 29, 2022. [Online]. Available: https://www.engineersgarage.com/what-is-opcode/

[10] A. Yewale and M. Singh, “Malware detection based on opcode frequency,” in Proceedings of 2016 International Conference on Advanced Communication Control and Computing Technologies, ICACCCT 2016, Jan. 2017, pp. 646–649. doi: 10.1109/ICACCCT.2016.7831719. Accessed on: Dec. 25, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7831719

[11] R. Rehman, D. G. C. Hazarika, and G. Chetia, “MALWARE THREATS AND MITIGATION STRATEGIES: A SURVEY,” J Theor Appl Inf Technol, vol. 31, no.2, 2011, [Online]. Available: www.jatit.org

[12] “Computer Virus_ What are Computer Viruses_”. Accessed on: Dec. 06, 2021. [Online]. Available: https://www.malwarebytes.com/computer-virus

[13] “What is a Computer Worm_ _ Malwarebytes”. Accessed on: Dec. 07, 2021. [Online]. Available: https://www.malwarebytes.com/computer-worm

[14] “What is a Trojan_ Is It Virus or Malware_ How It Works _ Norton”. Accessed on: Dec. 15, 2021. [Online]. Available: https://us.norton.com/internetsecurity-malwarewhatisatrojan.htm

[15] “What is a Rootkit & How to Remove it_ _ Avast”. Accessed on: Dec. 15, 2021. [Online]. Available: https://www.avast.com/c-rootkit#gref

[16] “What is Spyware_ _ Veracode”. Accessed on: Dec. 15, 2021. [Online]. Available: https://www.veracode.com/security/spyware

[17] “What is adware_ _ Kaspersky”. Accessed on: Dec. 15, 2021. [Online]. Available: https://www.kaspersky.com/resource-center/threats/adware

[18] “What is a Botnet_ _ Kaspersky”. Accessed on: Dec. 15, 2021. [Online]. Available: kaspersky.com/resource-center/definitions/what-is-a-packet-sniffer

[19] “What is Keystroke Logging and Keyloggers_ _ Kaspersky”. Accessed on: Dec. 15, 2021. [Online]. Available: https://www.kaspersky.com/resourcecenter/definitions/keylogger.

[20] “What Is Ransomware_ _ McAfee”. Accessed on: Dec. 15, 2021. [Online]. Available: https://www.mcafee.com/enterprise/en-us/security-awareness/ransomware.html

[21] N. Idika and A. P. Mathur, “A Survey of Malware Detection Techniques,” 2007. Accessed on: Dec. 15, 2021. [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.75.4594&rep=rep1&type=pdf

[22] R. Tahir, “A Study on Malware and Malware Detection Techniques,” *International Journal of Education and Management Engineering*, vol. 8, no. 2, pp. 20–30, Mar. 2018, doi: 10.5815/ijeme.2018.02.03. Accessed on: Dec. 18, 2021. [Online]. Available: http://www.mecs-press.net/ijeme/ijeme-v8-n2/IJEME-V8-N2-3.pdf

[23] M. G. Schultz, E. Eskin, E. Zadok, and S. J. Stolfo, “Data mining methods for detection of new malicious executables,” *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, pp. 38–49, 2001, doi: 10.1109/secpri.2001.924286. Accessed on: Dec. 02, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/924286

[24] G. G. Sundarkumar, V. Ravi, I. Nwogu, and V. Govindaraju, “Malware detection via API calls, topic models and machine learning,” in *IEEE International Conference on Automation Science and Engineering*, Oct. 2015, vol. 2015-October, pp. 1212–1217. doi: 10.1109/CoASE.2015.7294263. Accessed on: Dec. 27, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/5675808

[25] I. Firdausi, C. Lim, A. Erwin, and A. S. Nugroho, “Analysis of machine learning techniques used in behavior-based malware detection,” in *Proceedings - 2010 2nd International Conference on Advances in Computing, Control and Telecommunication Technologies, ACT 2010*, 2010, pp. 201–203. doi: 10.1109/ACT.2010.33. Accessed on: Dec. 15, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5675808

[26] Kaspersky, “Machine Learning for Malware Detection Learn more on kaspersky.com #bringonthefuture Contents.”. Accessed on: Dec. 15, 2021. [Online]. Available: https://media.kaspersky.com/en/enterprise-security/Kaspersky-Lab-Whitepaper-Machine-Learning.pdf

[27] Z. Markel and M. Bilzor, “Building a machine learning classifier for malware detection,” Jan. 2015. doi: 10.1109/WATeR.2014.7015757.

Accessed on: Dec. 15, 2021. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7015757

[28] U. Baldangombo, N. Jambaljav, and S.-J. Horng, “A STATIC MALWARE DETECTION SYSTEM USING DATA MINING METHODS.” Accessed on: Dec. 15, 2021. [Online]. Available: https://arxiv.org/abs/1308.283 [29] “What is Machine Learning_ _ IBM”.

[30] A. Aldahiri, B. Alrashed, and W. Hussain, “Trends in Using IoT with Machine Learning in Health Prediction System,” *Forecasting*, vol. 3, no. 1, pp. 181–206, Mar. 2021, doi: 10.3390/forecast3010012. Accessed on: Feb. 23, 2022. [Online]. Available: https://www.mdpi.com/2571-9394/3/1/12

[31] “What is Supervised Learning_”. Accessed on: Feb. 22, 2022. [Online]. Available: https://www.ibm.com/cloud/learn/supervised-learning

[32] “What is Unsupervised Learning_ _ IBM”. Accessed on: Feb. 22, 2022. [Online]. Available: https://www.ibm.com/cloud/learn/unsupervised-learning

[33] M. Horný, J. Morgan, M. S. Bori, M.-Y. Lin, and K. Min, “Bayesian Networks,” 2014. Accessed on: Feb. 24, 2022. [Online]. Available: https://www.bu.edu/sph/files/2014/05/bayesian-networks-final.pdf

[34] “Naïve Bayes Algorithm_ Everything you need to know - KDnuggets”. Accessed on: Feb. 24, 2022. [Online]. Available: https://www.kdnuggets.com/2020/06/naivebayesalgorithmeverything.html

[35] “Naive Bayes Classifier in Machine Learning - Javatpoint”. Accessed on: Feb. 24, 2022. [Online]. Available: https://www.javatpoint.com/machine-learning-naivebayes-classifier.

[36] “Multinomial Naïve Bayes’ multinominial”. Accessed on: Feb. 24, 2022. [Online] Available: https://towardsdatascience.com/multinomial-na%C3%AFve-bayes-fordocuments-classification-and-natural-language-processing-nlp-e08cc848ce6

[37] “Introduction to Logistic Regression”. Accessed on: Feb. 24, 2022. [Online] Available: https://towardsdatascience.com/introduction-to-logistic-regression-66248243c148

[38] “Simple Logistic Regression - StatsTest.com”. Accessed on: Feb. 25, 2022. [Online] Available: https://www.statstest.com/simple-logistic-regression/

[39] “ML _ Stochastic Gradient Descent (SGD) - GeeksforGeeks”. Accessed on: Feb. 25, 2022. [Online] Available: https://www.geeksforgeeks.org/ml-stochastic-gradientdescent-sgd/

[40] J. C. Platt, “Sequential Minimal Optimization: A Fast Algorithm for TrainingSupport Vector Machines ClearType View project Support Vector Machines ViewProject Sequential Minimal Optimization: A Fast Algorithm for Training Support 54 Vector Machines,” 1998. Accessed on: Feb. 25, 2022. [Online]. Available: https://www.researchgate.net/publication/2624239

[41] “J48 Classification (C4.5 Algorithm) in a Nutshell _ by Nilima Khanna _ Medium”. Accessed on: Feb. 22, 2022. [Online]. Available: https://medium.com/@nilimakhanna1/j48-classification-c4-5-algorithm-in-anutshell-24c50d20658e

[42] “Random Forest _ Introduction to Random Forest Algorithm”. Accessed on: Feb. 25, 2022 [Online] Available: https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/

[43] “VxHeaven”. Accessed on: Dec. 15, 2021. [Online]. Available: https://vxug.fakedoma.in/archive/VxHeaven/

[44] “DasMalwerk”. Accessed on: Dec. 15, 2021. [Online]. Available: https://dasmalwerk.eu/

[45] “VirusSign _ Malware Research & Data Center, Threat Intelligence, Free Downloads”. Accessed on: Dec. 15, 2021. [Online]. Available: https://www.virussign.com/downloads.html

[46] “command-not-found.com – objdump”. Accessed on: Dec. 15, 2021. [Online]. Available: https://command-not-found.com/objdump

[47] “NirSoft - freeware utilities_ password recovery, system utilities, desktop utilities”. Accessed on: Dec. 15, 2021. [Online]. Available: https://www.nirsoft.net/

[48] “Weka 3 - Data Mining with Open-Source Machine Learning Software in Java”. Accessed Jan. 15, 2022. [Online] https://www.cs.waikato.ac.nz/ml/weka/

[49] “VMware Workstation Player _ VMware _ ASEAN”. Accessed on: Dec. 02, 2021. [Online]. Available: https://www.vmware.com/asean/products/workstationplayer/workstationplayerevaluation.html

[50] “Crypto Obfuscator For .Net - Obfuscator With Code Protection, Exception Reporting, Optimization For .Net Assemblies, WPF, Silverlight, Windows Phone 7 and ASP.Net Websites”. Accessed Jan. 15, 2022. [Online] https://www.ssware.com/cryptoobfuscator/obfuscator-net.htm

[51] “How it works – VirusTotal”. Accessed Jan. 15, 2007. [Online] https://support.virustotal.com/hc/en-us/articles/115002126889-How-it-works