Android Malware Detection: A Hybrid Approach using Machine Learning

Thesis Proposal

Submitted By
Asadullah Hill Galib
MSSE 0718
IIT, University of Dhaka

Supervised By
Dr. B. M. Mainul Hossain
Associate Professor
IIT, University of Dhaka

Institute of Information Technology
University of Dhaka

10th August, 2019

Asadullah Hill Galib

Author
Asadullah Hill Galib
MSSE-0718
IIT, University of Dhaka

Supervisor
Dr. B. M. Mainul Hossain
Associate Professor
IIT, University of Dhaka

Contents

List of Figures

List of Tables

1 Introduction

Android Malware is an application running on Android operating system that implicitly or explicitly performs malicious activities. With the enormous growth of the Android System [1], malware also has grown significantly as well as upgraded its nature and activities [2]. Currently malware is smart enough to evade the typical Anti-Virus system or the obsolete approaches of malware detection. As Android malware generally tries to preserve the facade of a benign application, it is worthy and necessary to take an perceptive approach to defend them.

Basically, researchers nowadays have focused on machine learning approach for detecting Android malware, as signature-based approach is being proved outdated by the disreputable but brainy malware authors. The researchers analyze Android Malware with the following three approaches: *Static, Dynamic and Hybrid Analysis. Static Analysis* approach uses the static features of the application such as Permissions, API Calls, Intents, Call-Graph, Op-code, Hardware Usage Analysis, Meta-data etc. These static features are extracted from source code and meta-data. *Dynamic Analysis* approach investigates the dynamic behaviour of the application running on an emulated environment or on a real device. These dynamic features/behaviours include System Call, Network Traffic, File Operations, Running Services, Network Operations etc. *Hybrid Analysis* tries to incorporate both the static and the dynamic approach into a common ground.

Static Analysis faces many troubles such as code obfuscation, data obfuscation, encryption, over-privileged permissions etc by the shrewd malware authors. On the other hand, *Dynamic Analysis* also have some drawbacks like as small code coverage etc. It is quite possible to overcome these limitations by assembling static and dynamic approach into *Hybrid Analysis* [12, 13, 14, 15, 16]. But *Hybrid Analysis* prevails over those obstacles with the cost of complexity and time.
Researchers perform *Static Analysis* mainly focusing on the static feature 'Permissions', as almost all works in Static Analysis have incorporated that feature. A state-of-the-art work in this regard: Drebin [3] used 8 static features to detect malware using Support Vector Machine(SVM), though it fails to detect unknown families. Yerima et al. [4] employed 48 static features to detect malware using Bayesian Classifier. Other approaches incorporate Markov Chains of Behavioral Models [5], Neural Networks [6], Op-code and Binary Code Sequences [7]. In *Dynamic Analysis*, a state-of-the-art work: DroidScope [8] investigated native code and events that based on three layers such as hardware, Dalvik and Linux for classifying malware. Amos et al.[9] utilized 6832 dynamic features for different machine learning approaches for detection. DroidDolphine [10] used time logs and code traverse path, TaintDroid [11] used taint analysis for malware detection. Mostly *Dynamic Analysis* approach lack success with respect to *Static Analysis* by means of performance.

In *Hybrid Analysis*, Mobile SandBox [12] used 7 static features and investigated Native Code and Network Traffic as dynamic features to classify malware in spite of lacking solid performance metrics. Marvin [13] employed a lot of features both static and dynamic to detect malware, though it ignores famous dynamic feature- 'System Call' in its approach. Samadroid [14] incorporated 'System Call' and outperform Drebin, but it used old data-set. Kapratwar et al. [15] used 'Permissions' and 'System Calls' for Hybrid Analysis, even though its performance was poor. Talha et al. [16] revealed unknown characteristics of Android malware, however it did not integrate any machine learning approach. Hadm [17] subsumed Deep Neural Network for feature extraction from a set of static and dynamic features. All of these research works have their own limitations.

2 Research Question

As *Static* and *Dynamic Analysis* techniques have drawbacks individually, it is worthy to focus on *Hybrid Analysis* approach to mitigate those drawbacks. In Hybrid techniques, there are several works exists, but the limitations in those approach justify further investigation in this regard. This works aims to answer the following research question.

- **RQ1**: How do you automatically detect Android malware with *Static* and *Dynamic Analysis* using machine learning?
  To detect Android malware accurately, combining both static and dynamic approach may be beneficial according existing research. This question can be answered more precisely by investigating the following sub-questions.
  - **SQ1**: What are the important static and dynamic features for Android malware detection?
    Discovering important static and dynamic features can help to build an effective model, which leads to better malware detection.
  - **SQ2**: How do you extract and select the important static and dynamic features for Android malware detection?
    By answering this question, we can define a criteria for selecting important features from the extracted features and develop a process for extracting those features automatically. Finding out the effective combination of important features is the primary goal.

3 Research Methodology

Detecting Android malware using *Hybrid Analysis* needs assembling of both *Static* and *Dynamic Analysis.* To do so, this section points to the methodology of the proposed research in Table 1.

Step No	Activity title	Activity description and relation to research questions
1	Literature Review	Reviewing related works is the very first job of any research. Related works help to analyze, justify and modify researcher's idea. Besides investigating existing literature aids to sort out possible drawbacks and problems of one's research. This task is a continuous activity through out the research.
2	Data-set Collection	A good number of malware samples and benign samples are needed because, performance and effectiveness of machine learning approach depend on the magnitude of the data-set. So, collecting malware samples and benign samples is the first job. There exists some prominent data-sets in this regard, though most of them are not up to date.
3	Static Feature Extraction	Extracting static features automatically from the application(apk) file is to be performed for preparing data-set. Converting apk files to source code files is one of the subtask of this activity.
4	Dynamic Feature Extraction	Extracting dynamic features from the installed application is to be done for preparing data-set. Observing and investigating dynamic behaviour of applications and logging them is the primary goal of this activity.
5	Feature Selection	This phase incorporates taking any effective approach for reducing feature-set for computational advantage. Related works suggests that a lot of features can be deduced. Among them which features are more significant - is a vital issue with respect to performance and complexity.
6	Classification	Training and testing model using an effective machine learning approach is the next step to be carried out. In this phase, a brief analysis of different machine learning approaches' performance is to be performed.
7	Evaluation	This step is all about evaluating the result: accuracy, precision, recall and other significant metrics regarding viability and existing work. This evaluation will be compared with the state-of-the-art techniques such as Mobile-sandbox [12] or Marvin [13]. In accordance with the evaluation, interpretation is also necessary.
8	Technical Report	A technical report will be written and delivered for each of the major activity
9	Thesis Compilation	Finally, thesis will be compiled in order to fulfillment of Master of Science in Software Engineering

Table 1: Research Methodology

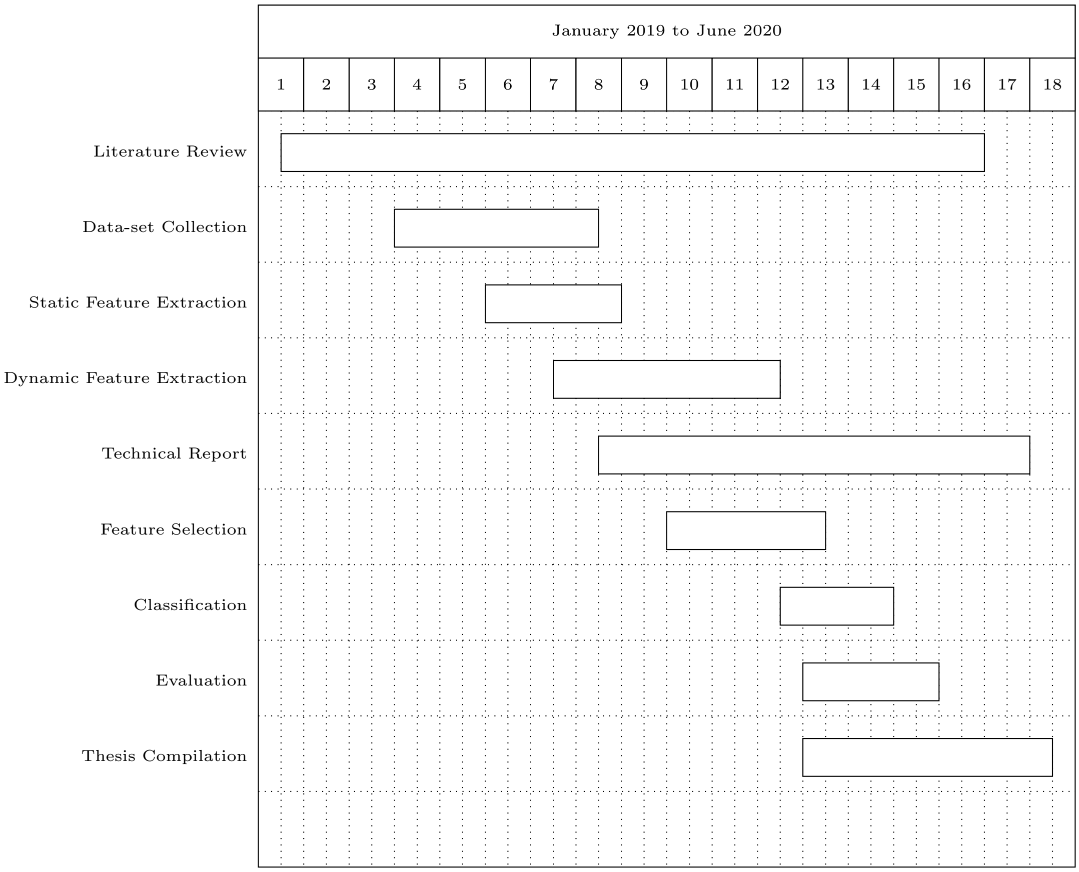

Figure 1: Timeline of the research

4 Research Time-line

Research time-line starts from January 2019 to June 2020. As Literature Review is a continuous process, this process will be carried out from the very beginning to the edge of the completion. And consequently Data-set Collection, Static Feature Extraction, Dynamic Feature Extraction and Feature Selection, Classification and Evaluation phases will be carried out. There are a plenty of overlapping among the phases because pipe-lining the whole process will be time efficient and viable. The overview of the time-line is depicted in Figure 1.

5 Rationale for the Research

Android Malware is the key factor for the most security breaches in android operating system. Moreover malware is growing exceedingly to keep pace with the immense growth of android applications. In each month, on average almost 10 million new malware is introduced [18]. Most alarming thing is that, nowadays malware authors also aware of the malware detection system and they use many novel and crafty evasion techniques to avoid them. So, to fight against these cunning black hats, we need to incorporate the most up-to-date and comprehensive detection technique. This work tends to alleviate those evasion techniques by integrating *Hybrid Analysis*. This research aims to improve the performance of malware detection process. By doing so, this research seeks to make a contribution to the academia as well as to the country.

Android is the most used mobile operating system (OS) in Bangladesh as the market share of it is 84.04% among other mobile OS by June, 2019 [19]. To assimilate the advantages of this vast popularity of android, the government of Bangladesh has taken many projects for developing android applications nationally. In 2015, the government launched 500 mobile apps costing 9.5 crore taka with an aim to achieve the new millennium development goal *Digital Bangladesh* by 2021 [20, 21]. Besides, ICTD (Information and Communication Technology Division) carries on various activities and projects such as Mobile Application Idea Generation Contest, Apps Sensitization – Boot Camp (7 Divisions), Apps Development Training (64 districts), Mobile application ready One office One app, National Mobile App Championship etc [22].

To accomplish the government's goal, secured mobile application is the most vital factor. Without maintaining the security concerns of these mobile applications, the whole plan might have been failed. Detecting malware in advance can guarantee a certain level of security for the latest applications developed by the government aided developers as well as local developers. This research aims to detect android malware in advance effectively. Thereby, this research can assist Bangladesh government's new millennium development goal: *Digital Bangladesh* by 2021.

References

[1] Popper, B. Google announces over 2 billion monthly active devices on Android. May 17,2017; Available from: https://www.theverge.com/2017/5/17/15654454/android-reaches-2-billionmonthly- active-users.

[2] Ahn, A. How we fought bad apps and malicious developers in 2017. 1/30/2018 3/20/2018]; Available from: https://android-developers.googleblog.com/2018/01/how-we-fought-bad-appsand- malicious.html.

[3] Arp, D., Spreitzenbarth, M., Hubner, M., Gascon, H., Rieck, K., & Siemens, C. E. R. T. (2014, February). Drebin: Effective and explainable detection of android malware in your pocket. In Ndss (Vol. 14, pp. 23-26).

[4] Yerima, S. Y., Sezer, S., McWilliams, G., & Muttik, I. (2013, March). A new android malware detection approach using bayesian classification. In 2013 IEEE 27th international conference on advanced information networking and applications (AINA) (pp. 121-128). IEEE.

[5] Mariconti, E., Onwuzurike, L., Andriotis, P., De Cristofaro, E., Ross, G.,

& Stringhini, G. (2016). Mamadroid: Detecting android malware by building markov chains of behavioral models. arXiv preprint arXiv:1612.04433.

[6] Ghorbanzadeh, M., Chen, Y., Ma, Z., Clancy, T. C., & McGwier, R. (2013, January). A neural network approach to category validation of android applications. In 2013 International Conference on Computing, Networking and Communications (ICNC) (pp. 740-744). IEEE.

[7] Jerome, Q., Allix, K., State, R., & Engel, T. (2014, June). Using opcode-sequences to detect malicious Android applications. In 2014 IEEE International Conference on Communications (ICC) (pp. 914-919). IEEE.

[8] Yan, L. K., & Yin, H. (2012). DroidScope: Seamlessly Reconstructing the OS and Dalvik Semantic Views for Dynamic Android Malware Analysis. In Presented as part of the 21st USENIX Security Symposium (USENIX Security 12) (pp. 569-584).

[9] Amos, B., Turner, H., & White, J. (2013, July). Applying machine learning classifiers to dynamic android malware detection at scale. In 2013 9th international wireless communications and mobile computing conference (IWCMC) (pp. 1666-1671). IEEE.

[10] Wu, W. C., & Hung, S. H. (2014, October). DroidDolphin: a dynamic Android malware detection framework using big data and machine learning. In Proceedings of the 2014 Conference on Research in Adaptive and Convergent Systems (pp. 247-252). ACM.

[11] Enck, W., Gilbert, P., Han, S., Tendulkar, V., Chun, B. G., Cox, L. P., ... & Sheth, A. N. (2014). TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones. ACM Transactions on Computer Systems (TOCS), 32(2), 5.

[12] Spreitzenbarth, M., Freiling, F., Echtler, F., Schreck, T., Hoffmann, J. (2013, March). Mobile-sandbox: having a deeper look into android applications. In Proceedings of the 28th Annual ACM Symposium on Applied Computing (pp. 1808-1815). ACM.

[13] Lindorfer, M., Neugschwandtner, M., Platzer, C. (2015, July). Marvin: Efficient and comprehensive mobile app classification through static and dynamic analysis. In 2015 IEEE 39th annual computer software and applications conference (Vol. 2, pp. 422-433). IEEE.

[14] Arshad, S., Shah, M. A., Wahid, A., Mehmood, A., Song, H., Yu, H. (2018). Samadroid: a novel 3-level hybrid malware detection model for android operating system. IEEE Access, 6, 4321-4339.

[15] Kapratwar, A., Di Troia, F., Stamp, M. (2017). Static and dynamic analysis of android malware. In ICISSP (pp. 653-662).

[16] Kabakus, A. T., Dogru, I. A. (2018). An in-depth analysis of Android malware using hybrid techniques. Digital Investigation, 24, 25-33.

[17] Xu, L., Zhang, D., Jayasena, N., Cavazos, J. (2016, September). Hadm: Hybrid analysis for detection of malware. In Proceedings of SAI Intelligent Systems Conference (pp. 702-724). Springer, Cham.

[18] Malware Statistics Trends Report — AV-TEST. (2019). Retrieved 4 August 2019, from https://www.av-test.org/en/statistics/malware/

[19] Operating System Market Share Bangladesh. (n.d.). Retrieved from http://gs.statcounter.com/os-market-share/all/bangladesh

[20] Report, S. O. (2015, July 26). Govt launches 500 mobile apps for better service. Retrieved from https://www.thedailystar.net/bytes/apps/govt-launches-500-mobile-apps-better-service-116977

[21] Report, S. B. (2015, March 07). Govt to make 500 mobile apps in Bangla. Retrieved from https://www.thedailystar.net/govt-to-make-500-mobile-apps-in-bangla-39757

[22] ACTIVITIES. (n.d.). Retrieved from https://www.nationalappsbd.com/?page$_id =$ 690