ANDROID MALWARE DETECTION USING GENETIC ALGORITHM

Dheeraj S[1], Goutham GS[2], Jayasurya R[3], Sanjay G[4], Rakshitha R[5]
[1,2,3,4] Student, [5]Assistant Professor
Department of Computer Science and Engineering
Vidya Vikas Institute of Engineering and Technology, Mysuru

Abstract - Android platform due to open source characteristic and Google backing has the largest global market share. Being the world's most popular operating system, it has drawn the attention of cyber criminals operating particularly through wide distribution of malicious applications. This paper proposes an effectual machine-learning based approach for Android Malware Detection making use of evolutionary Genetic algorithm for discriminatory feature selection. Selected features from Genetic algorithm are used to train machine learning classifiers and their capability in identification of Malware before and after feature selection is compared. The experimentation results validate that Genetic algorithm gives most optimized feature subset helping in reduction of feature dimension to less than half of the original feature-set. Classification accuracy of more than 94% is maintained post feature selection for the machine learning based classifiers, while working on much reduced feature dimension, thereby, having a positive impact on computational complexity of learning classifiers.

Key words - Android malware analysis, Genetic algorithm, Feature selection, Support vector classifier, artificial neural network.

I. INTRODUCTION

Android Apps are freely available on Google Playstore, the official Android app store as well as third-party app stores for users to download. Due to its open source nature and popularity, malware writers are increasingly focusing on developing malicious applications for Android operating system. In spite of various attempts by Google Play store to protect against malicious apps, they still find their way to mass market and cause harm to users by misusing personal information related to their phone book, mail accounts, GPS location information and others for misuse by third parties or else take control of the phones remotely. Therefore, there is need to perform malware analysis or reverse-engineering of such malicious applications which pose serious threat to Android platforms. Broadly speaking, Android Malware analysis is of two types: Static Analysis and Dynamic Analysis. Static analysis basically involves analyzing the code structure without executing it while dynamic analysis is examination of the runtime behavior of Android Apps in constrained environment. Given in to the ever-increasing variants of Android Malware posing zero-day threats, an efficient mechanism for detection of Android malwares is required. In contrast to signature-based approach which requires regular update of signature database, machine-learning based approach in combination with static and dynamic analysis can be used to detect new variants of Android Malware posing zero-day threats. In [1], broad yet lightweight static analysis been performed achieving a decent detection accuracy of more than 94% using Support Vector Machine algorithm.

II. LITERATURE SURVEY

There are various methods available for android malware detection classification & prevention in literature it has been observed that mainly three approaches were considered which are as follows:

Signature-based detection: may be a widespread technique supported looking for antecedently outlined virus signatures in input files . Signature detection has the advantage of detecting malicious activity before the system is infected by the malicious code.

Behaviour checking: is another standard technique supported a behaviour checker that resides within the memory longing for uncommon behaviour.During this case, the user is alerted. Behaviour checker encompasses a disadvantage that by the time a malicious activity is detected, some changes have already been done to the system.

Integrity Checker: is the technique that maintains a log of all the files that area unit gift within the system. The log could contain characteristics of files just like the file size, date/time stamp and substantiation. Whenever associate degree integrity checker is run, it'll check the files on the system and compares with the characteristics it had saved earlier. [14] Depending upon types of malware detection method/technique. The experimentation results validate that Genetic algorithm gives most optimized feature subset helping in reduction of feature dimension to less than half of the original feature-set. Classification accuracy of more than 94% is maintained post feature selection for the machine learning based classifiers tested the following classification algorithms: Artificial neural network, support vector machine obtaining the best results with Functional Trees. Their work is restricted to identifying malicious apps. Already extracted the permissions from the Android apk files, and then performed feature selection with information gain algorithm, and finally compared with ANN and SVM to classify Android apk files as malware or good¬ware. The algorithm achieved the highest precision of more than 94% accuracy.

III. EXISTING SYSTEM

In the existing system, signature-based detection methodology is used for malware detection. This is one of the most popular and common method in malware detection. Signature is unique feature for each file, something like fingerprint for an executable file. Signature based methods use patterns extracted from various malwares to identify them. These signatures are often extracted with special sensitivity for being unique, so those detection methods have small error rates. Signature-based detection detects malware by comparing the application signature or pattern captured with database of known attacks or threats.

a. DISADVANTAGES OF EXISTING SYSTEM

Signature based detection make it possible to detect known attack accurately and using less computational resources but it is less effective to unknown or new malware.
Moreover, it is hard to keep the signature up to date and constant update can consume the limited storage the mobile device has.
It requires a high amount of manpower, time, and money to extract unique signatures.

IV. PROPOSED SYSTEM

The main work in the Proposed is reduction of feature dimension to less than half of original feature-set using Genetic Algorithm such that it can be fed as input to machine learning classifiers for training with reduced complexity while maintaining their accuracy in malware classification. In contrast to exhaustive method of feature selection which requires testing for 2N different combinations, where N is the number of features, Genetic Algorithm, a heuristic searching approach based on fitness function has been used for feature selection. The optimized feature set obtained using Genetic algorithm is used to train two machine learning algorithms: Support Vector Machine and Neural Network. It is observed that a decent classification accuracy of more than 94% is maintained while working on a much lower feature dimension, thereby, reducing the training time complexity of classifiers.

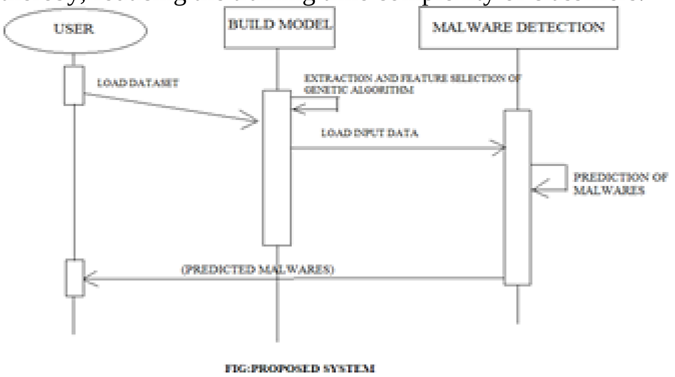

FIG 1: PROPOSED SYSTEM

V. METHODOLOGY

Two set of Android Apps or APKs: Malware/Goodware are reverse engineered to extract features such as permissions and count of App Components such as Activity, Services, Content Providers, etc. These features are used as feature vector with class labels as Malware and Goodware represented by 0 and 1 respectively in CSV format.

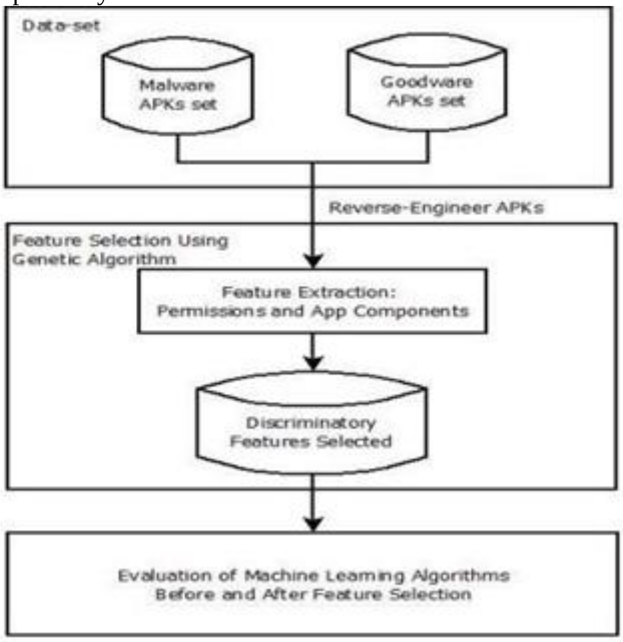
Data-set
Malware APKs set
Goodware APKs set
Reverse-Engineer APKs
Feature Selection Using Genetic Algorithm
Feature Extraction: Permissions and App Components
Discriminatory Features Selected
Evaluation of Machine Learning Algorithms Before and After Feature Selection

FIG 2: PROPOSED METHODOLOGY

VI. SYSTEM DESIGN

System design is a process of defining architecture and implementing the interfaces product design and data for a system specify the requirements and it could be seen as the application of product development.

a. APK FILES ALGORITHM FOR UPLOADING
Input: Uploading APK file
Output: Classify the APK file as malware or safe
Step 1: Load the CSV data set which has different features
Step 2: Extract the features of APK file
Step 3: Evaluate before optimized feature selection
Step 4: Apply genetic algorithm to these features and generate optimized feature attributes
Step 5: Train the ANN and SVC using Optimized features
Step 6: generate modal
Step 7: Apply the normalized feature vector to ANN or SVC machine according to the users selection
Step 8: Using result generated by output ANN or SVC machine declares the results an APK file is malware or safe

b. ARTIFICIAL NEURAL NETWORK
Artificial neural networks are a family of algorithms loosely based on the architecture and functioning of the biological brain. An artificial neural network is formed by several layers of nodes or neurons. The neurons on each layer are connected to the neurons on the next layer, with a connectivity strength associated to each connection.

c. SUPPORT VECTOR MACHINE
A support vector machine is a type of Artificial Neural Network which is widely used in classification problems. The goal of the SVM is to find a hyper plane in an n dimension space that separates the data points of different classes.

VII. RESULT ANALYSIS

The malware detection based on the permission features results are represented in fig 3. The (ANN) and (SVM) classifiers have the highest detection accuracy rates in experiment. The results exhibits the (ANN) precision 89.20% to recall rate 95% and the (SVM) precision 83% to recall rate 87%. This results show that outperforms all classifiers in terms of almost all evaluation measurements.

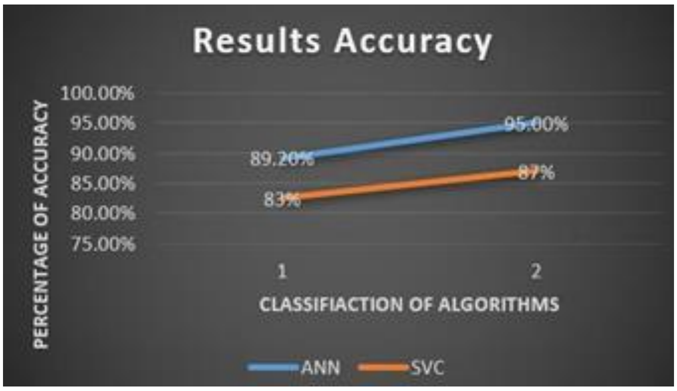

FIG 3: ACCURACY CHART

VIII. CONCLUSION

As the number of threats posed to Android platforms is increasing day to day, spreading mainly through malicious applications or malwares, therefore it is very important to design a framework which can detect such malwares with accurate results. Where signature-based approach fails to detect new variants of malware posing zero-day threats, machine learning based approaches are being used. The proposed methodology attempts to make use of evolutionary Genetic Algorithm to get most optimized feature subset which can be used to train machine learning algorithms in most efficient way. From experimentations, it can be seen that a decent classification accuracy of more than 94% is maintained using Support Vector Machine and Neural Network classifiers while working on lower dimension feature-set, thereby reducing the training complexity of the classifiers. Further work can be enhanced using larger datasets for improved results and analyzing the effect on other machine learning algorithms when used in conjunction with Genetic Algorithm.

REFERENCE

[1] “Global mobile statistics 2014 part a: Mobile subscribers; handset market share; mobile operators,” http://mobiforge.com/ research-analysis/global-mobile-statistics-2014-part-a mobilesubscribers-handset-market-share-mobile-operators, 2014.

[2] “Sophos mobile security threat reports,” 2014, last Accessed: 20 November 2014. [Online]. Available: http://www.sophos.com/en-us/threat-center/mobile-security-threat-report.aspx

[3] A. Reina, A. Fattori, and L. Cavallaro, “A system call-centric analysis and stimulation technique to automatically reconstruct android malware behaviors,” EuroSec, April, 2013.

[4] M. Backes, S. Gerling, C. Hammer, M. Maffei, and P. von StypRekowsky, “Appguard fine-grained policy enforcement for untrusted android applications,” in Data Privacy Management and Autonomous Spontaneous Security, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2014, pp. 213–231.

[5] Y. Zhou, X. Zhang, X. Jiang, and V. W. Freeh, “Taming information-stealing smartphone applications (on android),” in Proceedings of the 4th International Conference on Trust and Trustworthy Computing, ser. TRUST’11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 93–107. [Online]. Available: http://dl.acm.org/citation.cfm?id=2022245.2022255

[6] W. Enck, P. Gilbert, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth, “Taintdroid: An information-flow tracking system for realtime privacy monitoring on smartphones,” in Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation, ser. OSDI’10. Berkeley, CA, USA: USENIX Association, 2010, pp. 1-6.[Online].Available: http://dl.acm.org/citation.cfm?id=1924943.1924971

[7] S. Bugiel, L. Davi, A. Dmitrienko, S. Heuser, A.-R. Sadeghi, and B. Shastry, “Practical and lightweight domain isolation on android,” in Proceedings of the 1st ACM workshop on security and Privacy in Smartphones and Mobile Devices, ser. SPSM ’11. New York, NY, USA: ACM, 2011, pp. 51–62. [online].Available:http://doi.acm.org/10.1145/2046614.20466 24

[8] A. P. Felt, E. Ha, S. Egelman, A. Haney, E. Chin, and D. Wagner, “Android permissions: user attention, comprehension, and behavior,” in Symposium On Usable Privacy and Security, SOUPS ’12, Washington, DC, USA - July 11 - 13, 2012, 2012, p.3.

[9] Y. Zhou and X. Jiang, “Dissecting android malware: Characterization and evolution,” in Proceedings of the 2012 IEEE Symposium on Security and Privacy, ser. SP ’12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 95–109. [Online]. Available: http://dx.doi.org/10.1109/SP.2012.16

[10] “Contagio mobile, mobile malware mini dump.” [Online]. Available: http://contagiominidump.blogspot.com

[11] GoogleGroups, “Virustotal,” 2015. [Online]. Available: https://www.virustotal.com/

[12] Dr.Web, “Android malware review,” 2015. [Online]. Available: http://news.drweb.com/show/review/?lng=en&i=9546

[13] K. S. Labs, “Kindsight security labs malware report h1 2014,” 2014. [Online]. Available: http://resources.alcatel-lucent.com/?cid=180437

[14] A. Developer, “Android smsmanager api referencepage,” 2015. [Online]. Available: http://developer.android.com/reference/android/telephony/SmsManager.html