DATA MINING BASED MALICIOUS APPLICATION DETECTION OF ANDROID

Bala Naidu Barani sundram[1] **Swaminathan**[2]
[1]Associate Professor, College of Informatics, Department of Computer Science and Engineering,
, Bule Hora University,Bule Hora, Ethiopia, Africa
[2]Software Engineer, Vee Eee Technologies, Chennai, India
E-mail: bsunder2@gmail.com

Abstract

One of the most popular mobile phone platforms is an Android mobile phone platform owned by Google. The Android platform is open source to allow the developers to develop the full future application of the mobile operating system. Nowadays, malicious applications have been expanding in scale as an Android system. In this paper a data mining aided approach to detect malware applications in Android applications is presented. This approach capture the instant attracts that cannot be conclusively identified in past work. Static detection is one of the popular methods based on permissions detection of maliciousness in all the way through AndroidManifest.xml by classifiers. This paper suggests implementing a malicious application identify tool called Androidspy. Initially observe the relationship among system functions, sensitive permissions, and interface of responsive programming. Then, examine the system function grouping that has been clarifying the application behavior and characteristic vector. Following on the characteristic vectors, finding malicious android applications used to be naïve Bayesian, function decision algorithm, methodologies of j48decision tree. Androidspy is real-world applications as well as test sample programs. The test sample result confirms that Androidspy can be enhanced to detect malicious applications by using the system function group estimated with the previous work.

Keywords: Antenna Data mining, malicious application, android mobile, detection.

1. Introduction

The Android operating system for mobile phones has become an easy target for attackers because android market share value of android has expanded. Also, reverse engineering technique quickly targets android mobile applications. The android operating system permits the users can download applications from third-party application stores. This paper proposes a malicious android application detecting technique aided on data mining, and this technique detect immediate attracts via low charge. This technique builds a characteristic based vector on the function call system and classifies applications based on source, where it is used as information of the particular classifier. Compared via above techniques, the energetic fine-graineded applications solve the problems of static detecting against obfuscation and encryption capability. They are capable of detecting the instant attack with explaining about application behavior at the range of build characteristic vector. In the application testing, the application is classified with an agreement. The success rate of detection is increase based on useful types with decrease false positives through the statistics. To secure applications users from threats of malicious, more techniques have been suggested. But static detection is fully based on the code evaluation at concerned patterns. Although few static analyses have been succeeded, several obfuscation techniques have developed. Dynamic based analysis method is one of the popular methods of Performance – based identification method, which makes run the apk files in several platforms to analyze logs of run time.

2. Related Works

In [1] implementing our method on total applications is 100 (70 application benign, 30 malware application) as well as outcome prove that TRP is 96% through accuracy is 77 and F1 is 0.85. In [2] our method generalizes all applications. Sample outcomes on real-worldld applications with above 1000 malware application as well as 1000 benign applications samples validate the performance of the algorithm. In [3] our evaluation techniques of both methods show an F- score 95% as well as F- measure 89% for classification and permission-based on source code and classification model. In [4] compare between state – of – the art methods and SIGPID, the result shows the SIGPID is an effective way of identifying 93% of malware. As well as 91 % new malware/ unknown malware samples.

In [5] this method preferred for rapid examination of the application of Android files as well as information about suspicious applications. In [6] Sream framework, represent developed to allow rapid major scale verification of mobile application malware along with machine learning classifiers. In [7] a preliminary outcome shows that prediction reaches of accuracy is 86%, as well as f-Score, is 0.85. The dataset value is increased; the identifying accuracy increases simultaneously. In [8] purposely evaluate 8,000 applications; MD accuracy improves 10 % compared with the fate of the art antivirus protection systems. The parallel Storm Droid, and streaminglized process improves efficient rate by three times.

In [9] explained the latest feature sets, which resolve the problems of earlier studies in mobile application malware identification as well as analyze the performance of malware detection of machine learning classifiers. In [10] Androidetect test real-world applications as well as sample programs. The sample outcomes show that Androidetect can identify more malicious Android applications. In [11] this structure is used to improving the presentation of both FFT and MIMO – OFDM and also decrease the complexity of the hardware components. In [12] the main aim of the project is to reach the warning information about disaster quickly via the internet as well as made it available to these when need it. In [13], the filter segments of the focal scheme to restrict content message as well as the short classifier. Based on trail evolutions, this method performs well, recall as well as F1 score on the dataset. In [14] implement firewall policy based on visualization tool called firewall anomaly management environment. That approaches show how efficiently discover as well as solve anomalies in the firewall. In [15] user can interfere with the device using input touch buttons which is useful when the user doesn't have a mobile phone as well as connectivity of data.

In this paper proposed the framework of malware detection based on data mining for android applications devices. The main contributions of this paper are:

- The detailed description of extract permissions in apk files.
- The explanation of reduction as well as Information Gain(IG)
- Machine learning classifiers are used to perform empirical validations and make compare the performance of different weka based classifiers to different datasets.

Purposefully collected 170 applications from several android stores. The collection of applications consists of 100 benign applications as well as 70 malware applications. The malware detected applications from Contagio malware dump attached. Benign applications files were downloaded from the google application play store.

3. PROPOSED DESIGN

This paper proposes a malicious Android application identification method based on data mining, which implements in Androidspy tool to identify malicious applications. This tool is extracting the characteristics of the android applications using the hook technology, the process of injection technology, and interproceduralal communications that construct the eigenvectors. The algorithm for Androidspy tool specially designed to classify of normal applications and malicious applications called function class judgment algorithm. The data mining algorithms called the decision tree as well as naïve Bayesian. These are used to test classifiers and train set. The Androidspy tool contains two major modules namely analysis module and access module, the access module used to acquire character log for each and every eigenvector. The transformed eigenvector for security threatens to inculcate classifier, this type of classifiers used to identify the malicious application. The structure to represent the Androidspy identification system is shown in Fig.1.

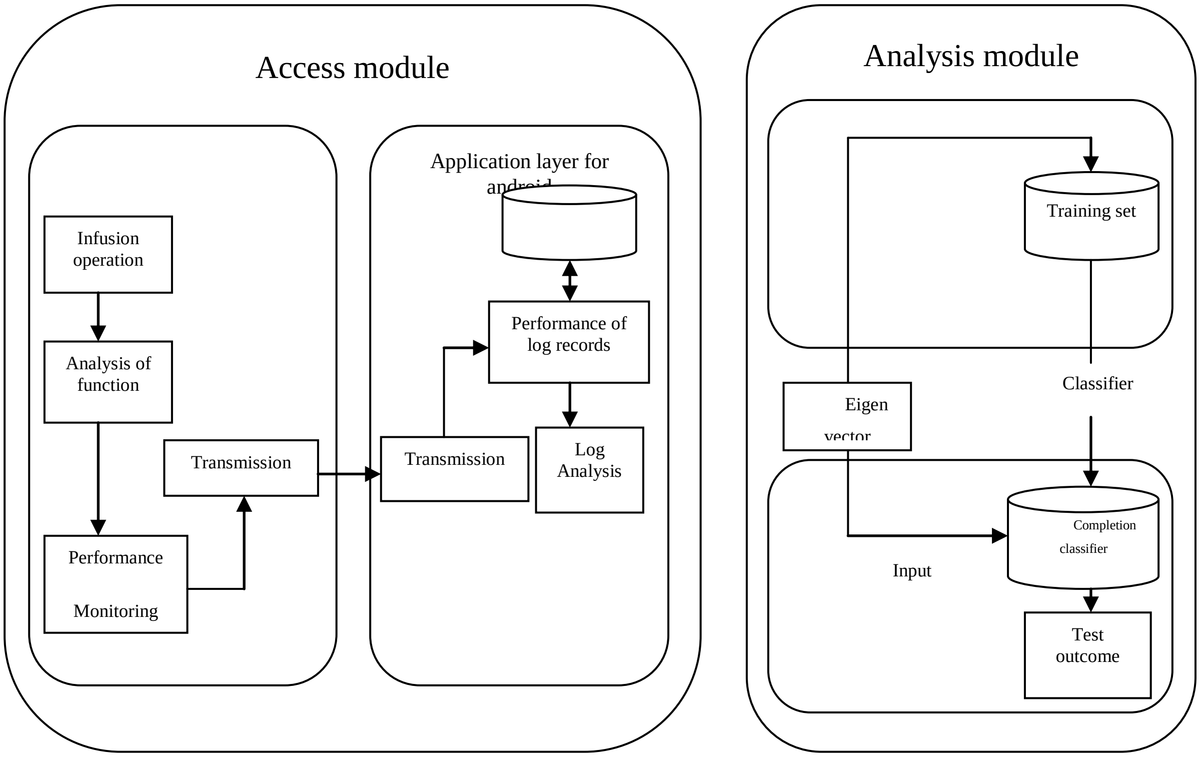

Figure. 1 Androidspy identification system structure

A. ACCESS MODULE

1) Infusion technique: We follow the hook technology as well as the injection of code technology to finish the interception system calling functions during the process of malicious application samples. To complete the interception function, the .so file attached with the server system to restore the IOCTL function.

2) Analysis: We must complete the analysis of system calls in Android system using Binder mechanism for the communication process. To check system sensitive as well as function calls system for the process using Binder communication system. The Binder communication system has a set of rules for analyzing the write and read command. It can get two types of parameters for intercepted function as well as realize the data analysis.

3) Monitoring: Behavior monitoring is used to check the outcomes of analysis data by joining the underlying data as well as extracting process also. In special programs gets access to open the monitoring service data and record operations in the application layer.

4) Record analysis: The customer rules to unite the outcomes of information, the java processing to import the data to analyze as well as getting data. To get data and analyzing process step by step of data adopt.

B. ANALYSIS MODULE

1) Construction of vector: The collections of more functions are employing to explain the behavior of applications. Consider the system permissions, permissions of sensitive, sensitive APIs and system function, that all collections of application behavior. At the time, the collections of behaviors of sensitive are getting multiple ways from record data to build more eigenvectors.

2) Classifier: In an analysis module, use decision tree algorithm as well as the naïve Bayesian algorithm to prepare for eigenvectors. The collections of record data in behaviors are measured and build eigenvectors. The eigenvectors are prepared for data of input.

3) Detection of malicious application:

The classifier can detect more applications. The classification of application based on 13 functional types as well as the neighbor algorithm. Therese are attached to build the classifiers, use this to detect security threaten.

5. RESULT AND DISCUSSION

The Android application behavior method describes the process of hook technology, as well as infusion technology, are together to process the features from various types of applications together with instant attack behavior. Bias principle included in naïve Bayesian to calculate the probability of each and every condition. Mobile performance is based on the behavioral interception, also the classification as well as the accuracy of Android application identification based on a behavioral interception. In this part, the performance of the behavioral interception impact is measured. In purposely install the malicious applications in android phone and send the message as well as to obtain private data via background. The complete process of injection through various files like .so files and .apk files based on behavior interception. The intercept of function as well as parameter analysis stored in the record for future process. The following table represents the proposed method effectiveness and proves some test samples compare with classic testing tools.

Table 1: characteristics of the proposed tool and other tool

ALGORITHMS	Select characteristics	Detection tools	Analytical method
C4.5,SVM	Authority API	Literature tools	Combination of dynamic and static
K-means,BN,J48	API	Andromaly	Dynamic
Random forest algorithm	Authority	PUMA	statics
No	System status	Literature tools	Dynamic
Application functional classification,C4.5,Naive Bayesian algorithm	System call functions	Androidspy tools	Combination of dynamic and static

Some features used to perform system behavior in a better way of malicious applications. The Androidspy can detect the instant attacks accurately based on system functions relationship, permissions as well as sensitive APIs operations. At the same time, other tools unable to detect irregular behavior produced by which Android applications. Moreover, Androidspy uses functional classifications of applications algorithm along with accurate identification.

Table: 2 Result of Andriodspy

IDENTIFICATION TOOLS	ACC(%)	TPR(%)	FPR(%)
Literature tools	87.1	81	13
Andromaly	78	57	14
PUMA	81	68	12
Literature tools	78	77	19.1
Androidspy tools	88	89	4

Evaluating with sample outcome shows Androidspy system to detect more malicious Android applications along with high classification accuracy as well as low positive rate. Androidspy system detects a better way compared to other tools. The sample result shows Andriodspy obtain better performance in the categories TPR, FPR as well as ACC.

6. Conclusion

In this paper, a system function to construct eigenvectors using dynamic analysis technique to extract the feature of the system is presented. The major classification model is reorganized by Android application function, decision tree, as well as naïve Bayesian to detect malicious application. First, identify an instant attack using system function. Second, the identification method based on Android application function algorithm and able to identify the irrelevant behavior. At last, compared with other related work, Androidspy tool performs better way about TPR, FPR as well as ACC.

References

1. Bhattacharya, Abhishek, and Radha Tamal Goswami. "DMDAM: data mining based detection of android malware." In Proceedings of the First International Conference on Intelligent Computing and Communication, pp. 187-194. Springer, Singapore, 2017.Peiravian, Naser, and Xingquan Zhu. "Machine learning for android malware

detection using permission and api calls." In 2013 IEEE 25th international conference on tools with artificial intelligence, pp. 300-305. IEEE, 2013.

2. Milosevic, Nikola, Ali Dehghantanha, and Kim-Kwang Raymond Choo. "Machine learning aided Android malware classification." Computers & Electrical Engineering 61 (2017): 266-274.
3. Li, Jin, Lichao Sun, Qiben Yan, Zhiqiang Li, Witawas Srisa-an, and Heng Ye. "Significant permission identification for machine-learning-based android malware detection." IEEE Transactions on Industrial Informatics 14, no. 7 (2018): 3216-3225.
4. Shabtai, Asaf, Yuval Fledel, and Yuval Elovici. "Automated static code analysis for classifying android applications using machine learning." In 2010 International Conference on Computational Intelligence and Security, pp. 329-333. IEEE, 2010.
5. Amos, Brandon, Hamilton Turner, and Jules White. "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, 2013.
6. Wu, Wen-Chieh, and Shih-Hao Hung. "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, 2014.
7. Chen, Sen, Minhui Xue, Zhushou Tang, Lihua Xu, and Haojin Zhu. "Stormdroid: A streaminglized machine learning-based system for detecting android malware." In Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, pp. 377-388. ACM, 2016.
8. Ham, Hyo-Sik, and Mi-Jung Choi. "Analysis of android malware detection performance using machine learning classifiers." In 2013 international conference on ICT Convergence (ICTC), pp. 490-495. IEEE, 2013.
9. Wei, Linfeng, Weiqi Luo, Jian Weng, Yanjun Zhong, Xiaoqian Zhang, and Zheng Yan. "Machine learning-based malicious application detection of android." IEEE Access 5 (2017): 25591-25601.
10. Pandiaraj, P. "Efficient Architecture of Combined Radix Dif Algorithm for MIMO-OFDM Application." International Journal of Advances In Signal And Image Sciences 2, No. 2 (2016): 9-13.
11. Amjath Ali, J., B. Thangalakshmi, and A. Vincy Beaulah. "IoT Based Disaster Detection and Early Warning Device." International Journal of MC Square Scientific Research (IJMSR) 9, no. 3 (2017): 20-25.
12. Prakash, Gyan, Nishant Saurav, and Venkata Reddy Kethu. "An Effective Undesired Content Filtration and Predictions Framework in Online Social Network." International Journal of Advances in Signal and Image Sciences 2, no. 2 (2016): 1-8.
13. Prakash, Gyan, Nishant Saurav, and Venkata Reddy Kethu. "An Effective Undesired Content Filtration and Predictions Framework in Online Social Network." International Journal of Advances in Signal and Image Sciences 2, no. 2 (2016): 1-8.

14. Khan, Azizuddin, and Gyan Prakash. "Design and Implementation of Smart Glass with Voice Detection Capability to Help Visually Impaired People." International Journal of MC Square Scientific Research 9, no. 3 (2017): 54-59.