Android Malware Detection System

Jasrinkaur Boyal[1], Rani Malode[2], Madhavi Patil[3], Shreya Patil[4]

[1,2,3,4]*Student, Dept. of Computer Engineering, MET's Institute of Engineering, Maharashtra, India*

Abstract - *Mobile devices have became popular in our lives since they offer almost the same functionality as personal computers. Among them, Android-based mobile devices had appeared lately and, they were now an ideal target for attackers. Android-based smartphone users can get free applications from Android Application Market. But, these applications were not certified by legitimate organizations and they may contain malware applications that can steal privacy information for users. In this project, a framework that can detect android malware applications is proposed to help organizing Android Market. The proposed framework intends to develop a machine learning-based malware detection system on Android to detect malware applications and to enhance security and privacy of smartphone users. This system monitors various permission based features and events obtained from the android applications, and analyses these features by using machine learning classifiers to classify whether the application is goodware or malware.*

Key Words: Malware, Goodware, Machine Learning, Permission based feature.

1. INTRODUCTION

In the past few years, mobile devices, like smartphones, tablets, etc. have become popular by increasing the number and complexity of their capabilities. Present mobile devices offer a large number of services and applications as compared to the one's offered by personal computers. Thus the number of security threats to the mobile devices has increased. And hence the hackers and malicious users are taking advantage of causing threat to the users personal credentials due to the lack of security mechanisms. In 2016, malware attacks increased to a count 3,246,284 malware samples. Android is the platform with the highest malware growth rate by the end of 2016. To overcome these security threats, various mobile specific Intrusion Detection Systems (IDSes) were proposed. Most of these IDSes are behavior-based, i.e. they dont rely on a database of malicious code patterns, as in the case of signature-based IDSes. In this project, we describe a machine learning based malware detection system for android based smartphones users. This system exploits machine learning techniques to distinguish between normal and malicious applications. We propose Android Malware Detection System to identify malware with efficiency and effectiveness. To develop a machine learning-based malware detection system on Android to detect malware applications and to enhance security and privacy of smartphone users.

2. LITERATURE SURVEY

In this chapter we will see the various studies and research conducted in order to identify the current scenarios and trends in Android Malware Detection System using static analysis and dynamic analysis.

E. Mariconti et.al[4] Mamandroid: Detecting android malware by building markov chains of behavioral models.arXiv preprint arXiv:1612.04433, 2016,has been proposed that this system build MAMADROID builds a behavioral model, in the form of a Markov chain, from the sequence of abstracted API calls performed by an app, and uses it to extract features and perform classification.

Y. Fratantonio et.al[3] has been describe that in this system Triggerscope: Towards detecting logic bombs in android applications. IEEE Security and Privacy, pp. 377–396, 2016 they just find malicious application logic that is executed, or triggered, only under certain condition as a logic bomb. The static analysis is the new technique use for that seeks to automatically identify triggers(logic bombs)in Android applications.

V. Rastogi et.al[2] Catch me if you can: Evaluating android anti-malware against transformation attacks. This IEEE Transactions on Information Forensics and Security, vol. 9, no. 1, pp. 99–108, 2014. in this system evaluate the state-of-the-art commercial mobile anti-malware products for Android and test how resistant they are against various common obfuscation techniques (even with known malware).

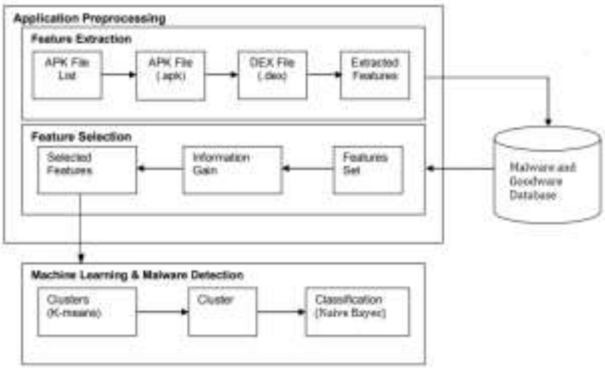
3. PROBLEM DEFINITION

As there is advancement in the usage of Android apps it becomes a need to identify the malicious behaviour of android apps to ensure the end user's security and privacy. The proposed system is a machine learning based malware detection system that classifies the apps as malicious and benign.

4. WORKING

The admin login's to the system by providing valid user name and password. The apk of the application that is to tested for malicious behavior is loaded from the file containing list of apks. The apk is extracted using the apk extractor. The extracted features contain XML features and

DEX file further the XML features are compared with the existing database of apks. The system clusters the similar features into number of clusters specified by the tester. Clustering done using k-means algorithm and for the classification of the apk into malicious or good ware Naïve Bayes algorithm is used, this is the first layer for detection using only XML features. If the apk remains unclassified in the first step then the DEX features are used in the second layer. The same procedure of clustering and classification are repeated for the second layer and finally the apk is classified.



5. DESIGN

In this Section describes the System Requirement Specification (SRS) to be implemented for Android Malware Detection system. It also explains the architecture of the system and external interface requirements.

5.1 Software Requirement Specifications

The Software Requirement Specification describes the scope of the project, functional and non-functional requirements, System requirements, Analysis Models. It also elaborates the system architecture of the Android Malware Detection System.

5.1.1 Project Scope

The proposed system identifies malware with efficiency and effectiveness. The novel techniques enabling effectiveness is the machine learning and data mining to automatically extract features to detect malware on both xml files and bytecode.

5.1.2 Functional Requirements

5.1.2.1 User Classes and Characteristics

User 1 : Admin This type of user will be users who will train system with new malware types.
User 2 : User This type of user will be users who detect malware.

5.1.2.2 Assumptions and dependencies

- System will assume that dataset is properly trained.
- Database has been properly maintained to detect malware.

5.1.3 Non-Functional Requirements

5.1.3.1 Performance Requirements

The application should be provided with a trained data set and apk file should be provided in order to check it's malicious nature.

5.1.3.2 Safety Requirements

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.

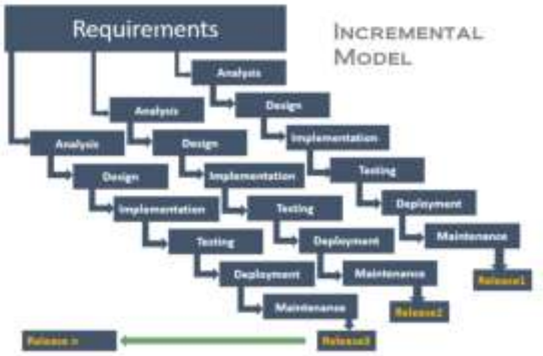
5.1.2.3 Security Requirements

To prevent the access from unauthorized user to database. Also we can provide login system for security purpose.

5.1.4 Analysis Model

5.1.4.1 Incremental Model

The incremental model combines the elements of waterfall model applied in an iterative fashion. As in Fig 3.1 the incremental model applies linear sequences in a staggered fashion as calendar time progresses. Each linear sequence produces deliverable increments of the software. When an incremental model is used, the first increment if often a core product that is basic requirement is addressed but many supplementary features remain undelivered. The core product is used by the customer. As a result of use and/or evaluation, a plan is developed for the next increment. This addresses the modification of the core product to better meet the needs of customer and delivery of additional feature and functionality. This process is repeated following the delivery of increment, until the complete product is produced. Fig below depicts an incremental model that contains five phases:



6. TECHNICAL SPECIFICATIONS

6.1 Advantages

- Real time Malware Detection System

6.2 Limitations

- Limits detection of Android Malware.
- Internet connection is required.

6.3 Applications

- Secure Application.
- Malware Detection.

6.3.1 Hardware Requirements

- Processor Core 2 Duo or Higher
- RAM 4GB or Higher
- Hard Disk/Secondary Memory Min 500 GB

6.3.2 Software Requirements

- Windows XP /7 onwards
- C# Programming

7. CONCLUSION

We have developed a system for classifying Android applications as malicious or benign applications using machine-learning techniques and Naïve Bayes algorithm. To generate the models, we have extracted several permission features from several downloaded applications from android markets. Some of the malware applications are used from malware sample database and both malware and normal applications are classified by using machine learning techniques.

REFERENCES

[1] Juniper Networks: 2011 Mobile Threats Report(February 2012).

[2] Burguera, U.Z., Nadijm-Tehrani, S.: Crowdroid: Behavior-Based Malware Detection System for Android. In: SPSM11, ACM,October 2011).

[3] G. Holmes, A. Donkin, and I.H. Witten, Weka: a machine learning workbench, August 1994, pp. 357-361.

[4] Xie,L.,Zhang,X.,Seifert, J.P.,Zhu, S.: pBMDS: a behavior-based malware detection system for cellphone devices. In: Proceedings of the Third ACM Conference on Wireless Network Security, WISEC 2010, Hoboken, New Jersey, USA, March 22-24 2010, ACM(2010) 37-48.