Mobile Application For Malware Detection

Pranjali Deshmukh, Pankaj Agarkar

[1] *M.E, Research Scholor, P.G. Department Of Computer Engineering , D.Y.Patil School Of Engineering Maharashtra,India*
[2] *Assistant Professor, P.G. Department Of Computer Engineering , D.Y.Patil School Of Engineering Maharashtra,India*

Abstract - *Mobile* Operating System, such as android, apples ios supported features like rich mobile applications, So understanding security risks of mobile application adventure. Andriod mobile platform provide both system evaluation and usability study. When upload and download an .apk file in android with there are some malicious activity are operated with android. On the other hand, for acquire a protection of android application, some proposals are executed. It is the methodology that performs apps security and provide user friendly interface on a mobile phone. So we targeted for achieve mobile application security in android mobile.

Key Words: Android ,Mobile malware detection, Hadoop distributed file system, Data mining

1. INTRODUCTION

In recent years, mobile computing is revolutionary part with different categories, different features and different applications. The mechanism with investigation for find out easier way to develop the application in mobile system. Mobile phone became pocket size, little device and apps are stored in mobile. Now a day's mobile application can be developed with latest technology, using some earlier methods. A typical mobile application with mobile computing device with advanced technology that access internet, email or web application run and install application. The popularity of mobile apps related to internet surfing is increased more and more downloading and uploading that becomes easy ,more number of downloading apps is from google or apple. The google provides free service for downloading apps. Mobile malware threats have recently turn into a actual fear. We evaluate the state-of-the-art marketable mobile anti-malware yield for android and check how anti they are beside different common difficulties to understand techniques (even with well-known malware). [12] Such an valuation is important for computing the accessible defense against mobile malware threats, as well as suggesting effective, next-generation solutions.

1.1 Downloading Rate

In 2011 downloading of mobile apps was 23 billion. Most of computer software are executed on mobile device. In 2012 downloading rate was double than the 2011, it was 45 billion. In 2013, downloading of mobile apps was approximate 81 billion apps. In same year 800 000 android apps are included in google play market. Maximum 48 billion downloading was performed in May 2013 at the same time apps were stored from apple. In 2014 it was double than 2013.While downloading apps from different sources also comes the mobile threats from different sources .There are many methods for testing and detecting malware from apps ,but that are run off and spread in android market.

1.2 Malware Origin

Different technique are follows many user while downloading apps in mobile, android provide easy way for that. But while downloading that automatically, brings mobile threats. It also means with this many more techniques are available for detection of malware. There are different ways of android malware origin.

1. First way is to set up apps from other party market set.

2. Special market set has special security service.

3. It is simple way to seaport an accessible window based botnet client to android platform.

4. An android application programmer are upload their application without inspection.

5. A number of applications have been filled in and broaden.Malicious performance android malware.

1. Freedom appreciation to root.
2. Leak confidential data.
3. Face finest numbers.
4. Botnet activity.
5. Backdoor generate via SMS.

There are many already discovered malware some of them are:

Android.PDaspy, android.Opfake, android.Obad, droidDream trojen, android.luckycat, Hippo Walk and Text android.Godwon, android.Gingerbreak, Dog War, android.BiigeZsone, Exploid, plankton etc.

2. RELATED WORK

Today's sizzling concept is to achieve security of mobile andriod apps. Most of researchers used static, dynamic analysis method, permission based method and cloud computing platform. Many researchers perform some analysis methods

Barrera et al.[2] have proposed analysis based on permission based safety form and proposed some that is self organizing map algorithm provides two-dimensional idea of high dimensional data and also proposed crowoid which uses linux system calls for locate malware system calls are open( ) to open it, read( ) for reading file, access ( ) to accessing it , chmod ( ) for changing mode, chown ( ) for changing owner. Next one is presented SAAF offer program. It consider 136 000 gentle apps and 6100 malicious app. SOM algorithm which conserve immediacy and furnish a simplified and relational view of a greatly complex dataset.

Portokalidis et al.[3] have proposed a methodology paranoid android that is complete malware analysis. It used to perform security analysis android that is based on mobile replicas and cloud platform. It was dynamic behavior analysis of system so it difficult to detect on run time.

Zhou et al. [4] also proposed droidMoss which takes fuzzy hashing technique. It used to perform security analysis with technique NFS storage and ZFS file system. Fuzzy hashing technique is more difficult to perform analysis of detecting malwares.

Enck et al. [5] have proposed that gives suggestions on to users concerning application and that appeal blacklisted sets of permission. Their outcome demonstrate the spacious use wrongly of privacy perceptive information, the verification of telephone mistreat, spacious counting of ad libraries in Android application, and the weakening to securely use Android APIs of many developers.

Felt et al. [6] have proposed stowaways to test over privilege in android application and used to calculate 940 applications from market of android, they identified and quantified developers pattern leading to over privilege. They determine androids access control policy through automotive testing technique. Their outcome nearby a fifteen crease enhancement more than the android documentation and expose that most deplorer are trying to go after the rule of least benefit but not succeed due to the lack of trustworthy permission information.

Elish et al.[7] employed a study means make data dependence graphs statically with interprocedural call connectivity information that captures the data utilization dealings in programme through identifying the directed paths between user inputs and entry points to process given that serious system services. Some malwares may try to avoid their data dependence inspection by misusing the user's inputs while performing malicious behavior, so their work needs to be better in these situations.

Kantiya Junhom [8] have proposed Cloudbroid is the application depend on android apps & cloud stack, it is an crossing point between mobile and cloud users. In the cloudbroid the application that administer cloud stack management & which is depend on android application using REST Principles. The main idea behind this is accessing the cloud stack system anyplace & anytime with many apply to construct an own business occasion.

Cloudbriod give features like dashboard, storage, template, account, infrastructure, event, project, package & report & also gives simple to read interface than CloudManagerAdvanced. It means we can say that the Cloudbriod rise above the problem of Cloud Manager Advanced which is not easy to deal or understand management of Cloudstack. Cloudstack is platform which holds information as service and also handle, assemble the large number of network virtual machines. In this technique we can make use of most accepted and functional REST Principle support on java enterprise edition tool or beans. The full implication of the REST is a Representational State Transfer. REST presents the communication between client & server is in a tricky ways. The protocol is used for the interaction with HTTP. HTTP has port number eighty. There are some problems with this it is more difficult to manage cloud system on mobile. Another application that is implemented UI for cloudstack. Web. Another application that is implemented UI for cloudstack.

Jaykumar Karnewa[9] have proposed get security by using data mining conception also k-means and clustering algorithm. Usability and feasibility is increased day by day of mobile apps, for the purpose, increasing safety of mobile apps explain the k-means technique beside the malicious activity of android apps. In this mechanism, arrange numerical data, or training gets are ordered in vectors with a dimension equal to number of features to estimate. K-means based on calculations. Sometimes results in inaccuracy. It also implemented for collection of more and more data.

R.C. Shivamurthy[10] has validate all mobile apps and to clear out malware from mobile app market by using cloud computing platform. It defines system mechanism & which provides plan from which the system developed. Whenever we collect more number of app's laboring and network behavior data, in this situation it not suitable to implement this system.

Zhen Chen[11] have proposed method by means of cloud storage to maintain composed traffic data and then giving out it with cloud computing platform to find the malicious attack presented and required computing of storage that depend on real trace data. In this method phishing operation are performed which are totally based on cloud platform, parallel processing was appraise which is practical sweeping to forensic analysis of other network attacks.

Rastogi [12]developed DroidChameleon, a efficient framework with various transformation techniques, and used it for only study. The results only commercial anti-malware applications for Android are having some trouble none of these tools is opposing beside common malware conversion techniques. In addition, a common of them can be irrelevantly crushed by applying minor transformation over known malware with little try for malware authors. Finally, in light of their results, they proposed possible remedy for improving the current state of malware finding on mobile devices.

Wei Peng[13] have proposed a common behavioral description of immediacy malware which depend on naive Bayesian model, which has been effectively useful in non-DTN settings such as clean out email spam's and discover botnets. The main problem is for widen Bayesian malware detection to DTNs i.e. inadequate facts versus indication gathering risk" and "clean out false evidence sequentially and circulately and recommend a simple method, that became , to tackle the defy. They also proposed two extensions to appear in front, strict filtering, to tackele the defy of malicious nodes sharing false evidence. This method is suitable for Bayesian network and also depends on Bayesian concepts not for the mobile applications.

In modern years, Many authors works with security of wireless communication networks, specially for malwares. Those are just for amusing to hidden attacks on devices, but are more risk able factor along with wireless device such as mobiles. The problem comes with the actual accessing the mobiles. Many authors gives [14] two examples one is Bluetooth spate and second is wireless LAN. The authors wind up that mobile security, and, in picky, worm transmission over wireless networks, are an motivating and original concept. Still, it must be definite to make sure the models beside actuality, and after expect threats that unsuccessful to show up, it must be able to recognize where it went erroneous.

3. IMPLEMENTATION DETAIL

3.1 System Architecture

The architecture that gives us structure of a system. It provides the idea regarding how the system is going deploy. The architectural design process is troubled with establishing framework of a system. In this we not only analyze but also checking security of android apps. When uploading any .apk file for the purpose analysis, apply hash function to store the key value. If the key is matched then result is return to submitter, if the key is not matched it means that new apk file. In this situation the .apk file result is stored in hadoop structure. So invokes tools ASEF and SAAF this is for the store in hadoop database. The reason behind use HDFS is that while analyzing framework related to clouds and storage the helpfulness by assessment of hadoop distributed file system ,is more suitable than others. The many experiment on amazons elastic cloud [15] carry outed successful application of hdfs.it has very good performance for storage of big data on mobile device.

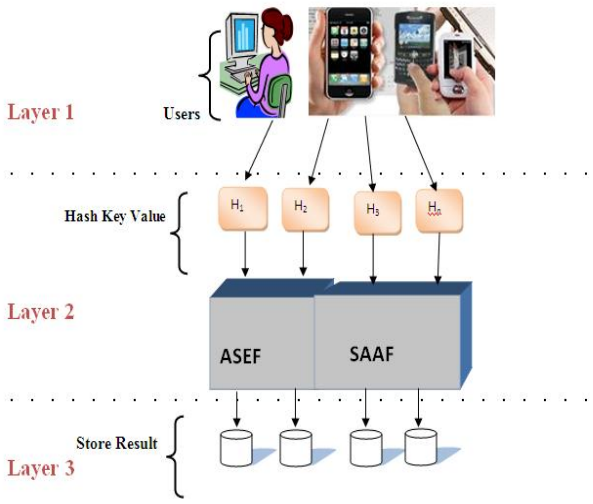

Fig-1. System Architecture

In this, architecture, to calculate mobile apps for improving safety of mobile apps, It combines static and dynamic technology to briefly that calculate android applications and decrease the sum of time to an suitable level. It invokes ASEF and SAAF that calculate apps and calculate time for estimate time. In Machine learning relate the tool information held by executables and used the names of the function and calls appearing at the output that to form manner training set and malicious set. Machine learning is a technique of training algorithms to enhance our perceptive of a firm data set. It gives computers the ability to learn that makes explicitly programmed.

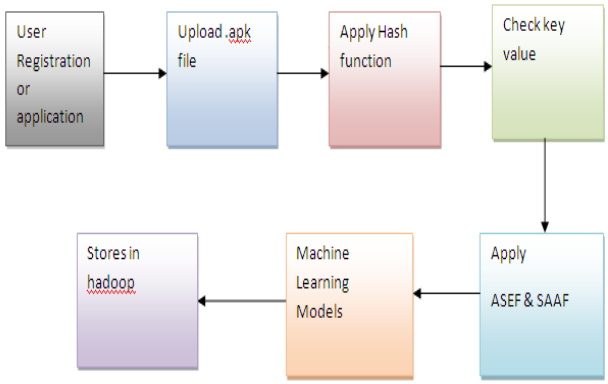

Fig.-2. Block Diagram

3.2 Work Break Down Structure

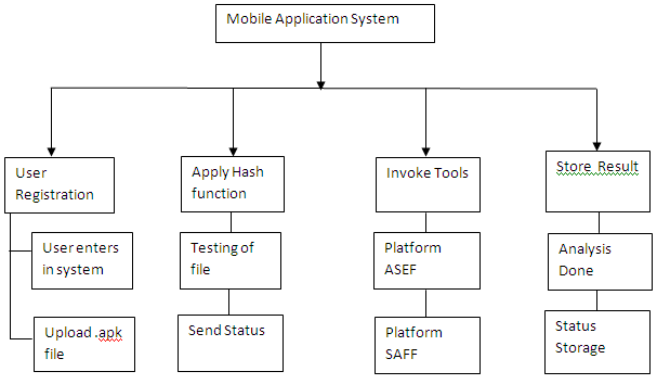

Fig-3. Work Breakdown Structure

The work break down structure mainly focuses on following areas:

Module 1:- In this module upload the .apk file on Hadoop distributed file system as cloud. Using mapper and Reducer.

Map & Reduce the input file. Assign the key, value to input file content.

Module 2:- In this module invoke the Android Security Evaluation Framework (ASEF) & Static Android Analysis Framework (SAAF) evaluate the android applications & total time required for evaluation.

Module 3:-. Use Hadoop distributed file system for keeping android applications verify android applications routinely to filter out malware applications.

Module 4:- Use machine learning to conduct automotive forensic analysis of mobile apps based on the generated multifaceted data in this stage.

3.2 Mathematical Model

1. Let S be the MobSafe system

$$S = \{I, O, F, Fs, FI, \Phi\}$$

2. Identify the inputs

$$I = \{A,M,R,P\}$$

Where:-

A=.apk file

M=assign Key, Value pair

R= Reduce Duplicate Process

P=Put result on HDFS

3. Identify set of Function

Let F be the set of Functions

F = {F1, F2,F3}

Where:-

F1 =Verify Information

F2 =Map and Reduce the input file

F3 =Apply ASEF & SAAF

4. Identify the Outputs

Let O be the set of outputs

O = {O1,O2}

Where:-

O1 =Map & Reduce Successfully

O2 =Successfully Find Vulnerability

5. Final State

Fs = Find Vulnerability using ASEF & SAAF

6. Failure case

Fl = Errors in measuring the input parameters

7. Constraints

Let Φ be the constraints Φ = Cl

Where: - Cl = Accuracy in measuring the input parameters.

Venn Diagrams:

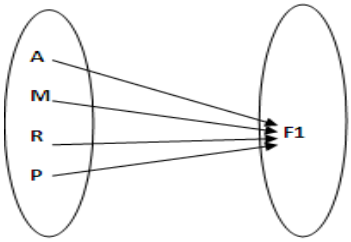

Fig-4. Verify the Information

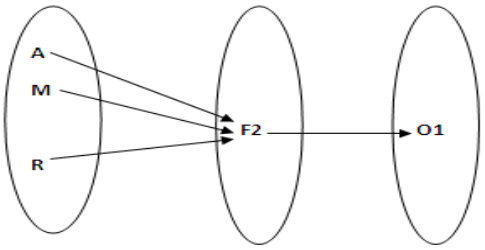

Fig-5. Map and Reduce the input file

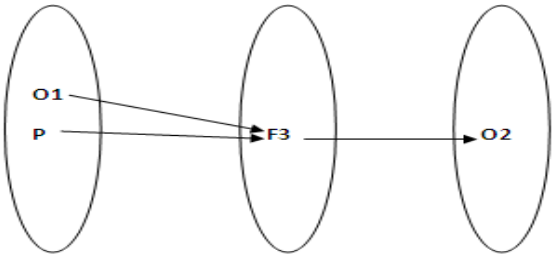

Fig.-6. Apply ASEF & SAAF

4. Result and Discussion

4.1 Dataset

In this paper we are considering application file which is already collected and saved in the input database. This application file is considered for experiment.

Fig.-7. Login the file

Fig-8. Uploading Of file

4.2 Resultset

In the first module we are going to consider application file is collected as input data. In second module original file from the dataset invokes the tool ASEF and SAAF. It evaluates application file and estimate the time required for evaluated. Later it verifies android application and to clear out malwares from the application file. At last we are expecting the secure application run on android by using the machine learning to conduct automotive forensic analysis of mobile apps. Finally, it detecting and testing the malwares from the mobile apps.

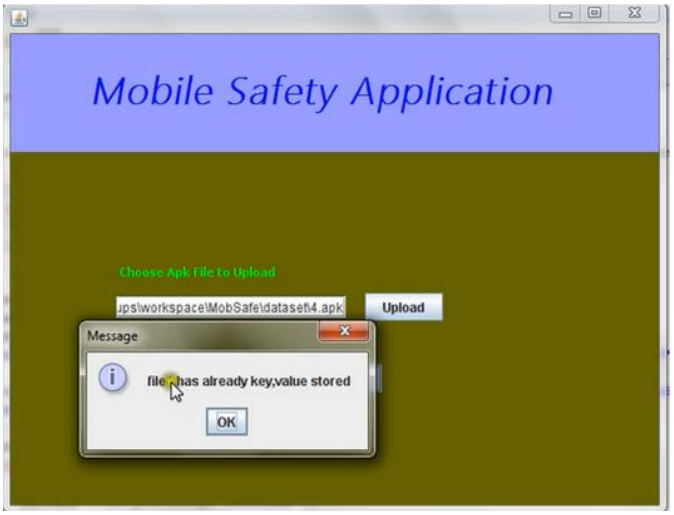

Fig.-9. Check Key Value

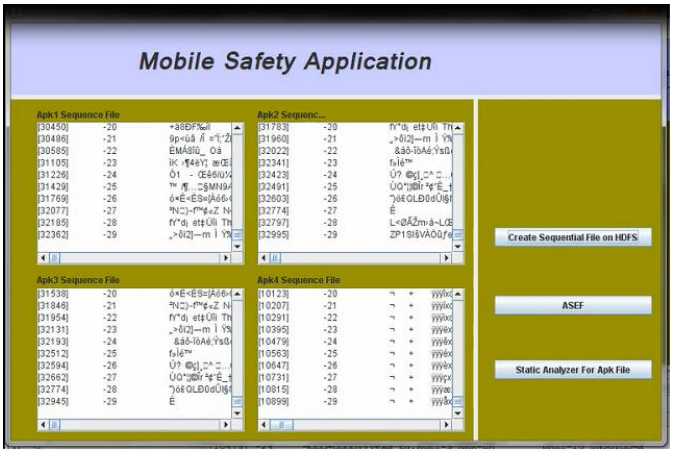

Fig.-10. Create Sequential file On HDFS

5. CONCLUSIONS

We can evaluate the security of android apps; we proposed methodology for estimation of analysis and design pattern of android apps. It is well performed on the basis of cloud and data mining. For the purpose, to perform the approximate calculation of active android apps to achieve security. It is proportion of mobile apps popularity, developed mechanism ASEF and SAAF. We can calculate approximately that the amount of dynamic android mobile apps and the normal apps set up in one mobile Android device, and the rising proportion of mobile apps. as mobile app market give out as the major line up of guard beside mobile malwares, it is convenient to employ data mining stage to security malware in mobile app promote.

ACKNOWLEDGEMENT

It is my pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced our thinking, behavior, and acts for the success of this.

I express my sincere gratitude to Dr. S. S. Sonvane, Director for providing me an opportunity to undergo this dissertation work. I am thankful to my guide and M.E Coordinator of Computer Engineering Prof. Pankaj Agarkar, for exemplary guidance, cooperation and motivation provided to us during the seminar for constant inspiration and presence. I take this opportunity to express my profound gratitude and deep regards to Prof. S. S. Das, HOD of Computer Engineering for support, monitoring and constant encouragement throughout the course of this paper.I am obliged to all staff members of my department, for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my dissertation work . Lastly, I would like to thank my parents for their moral support and my friends.

REFERENCES

[1] Zhen Chen Bin Cao Wenyu Dong Yu Guo Junwei Cao Jianlin Xu, Yifan Yu. Mobsafe: Cloud computing based forensic analysis for massive mobile applications using data mining. TSINGHUA SCIENCE AND TECHNOLOGY, 2013.

[2] D. Barrera, H. G. Kayacik, P. C. van Oorschot, and A. Somayaji, A methodology for empirical analysis of permission-based security models and its application to Android, in Proc. 17th ACM Conference on Computer and Communications Security, Chicago, USA, 2010, pp. 7384.

[3] G. Portokalidis, P. Homburg, K. Anagnostakis, and H. Bos, Paranoid Android: Versatile protection for smartphones, in Proc. 26th Annual ACM Computer Security Applications Conference, Austin, USA, 2010, pp. 347-356.

[4] Y. Zhou, Z. Wang, W. Zhou, and X. Jiang, Hey, you, get off of my market: Detecting malicious apps in official and alternative Android markets, in Proc. 19th Annual Network and Distributed System Security Symposium, San Diego, USA, 2012.

[5] Enck, D. Octeau, P. McDaniel, and S. Chaudhuri, A study of android application security, in USENIX Security Symposium, San Francisco, USA, 2011.

[6] A. P. Felt, E. Chin, S. Hanna, D. Song, and D. Wagner, Android permissions demystified, in Proc. 18th ACM Conference on Computer and Communications Security, Chicago, USA, 2011, pp. 627-638.

[7] K. O. Elish, D. Yao, and B. G. Ryder, User-centric dependence analysis for identifying malicious mobile apps, in Workshop on Mobile Security Technologies (MoST), San Francisco, USA, 2012

[8] Paphawee Lumlert Phapan Niampoonthong Vasaka Visoottiviseth Kantiya Junhom, Sirada Semkham. Cloudbroid: An android mobile application for cloudstack management system. Third ICT International Student ProjectConference, 2014.

[9] Jaykumar Karnewa Snehal Umratkar. K-means algorithm for selective _ltration of malicious android mobile applications. International Journal of Advent Research in Computer Electronics, 2014

[10] Rastogi, V. ,Yan Chen , Xuxian Jiang, Catch Me If You Can: Evaluating Android Anti-Malware Against Transformation Attacks, Information Forensics and Security, IEEE Transactions on (Volume:9 , Issue: 1)2013

[11] Zhen Chen, Fuye Han, Junwei Cao, Xin Jiang, and Shuo Chen Cloud Computing-Based Forensic Analysis for Collaborative Network Security Management System TSINGHUA SCIENCE AND TECHNOLOGY,February 2013

[12] Rastogi, V. ,Yan Chen , Xuxian Jiang, Catch Me If You Can: Evaluating Android Anti-Malware Against Transformation Attacks, Information Forensics and Security, IEEE Transactions on (Volume:9 , Issue: 1)2013:25 , Issue: 1),2014

[13] Wei Peng , Feng Li, Xukai Zou , Jie Wu, Behavioral Malware Detection in Delay Tolerant Networks, Parallel and Distributed Systems, IEEE Transactions on (Volume:25 , Issue: 1),2014

[14] Zanero, S., Politec. di Milano, Milan, Italy, Wireless Malware Propagation: A Reality CheckSecurity & Privacy, IEEE (Volume:7 , Issue: 5),2009.

[15] Yongwei Wu ,Beijing, China,Feng Ye,Kang Chen,Weimin Zheng,Modeling of Distributed File

Systems for Practical Performance Analysis Parallel and Distributed Systems, IEEE Transactions on (Volume:25 , Issue: 1),2013.

[16] Cloudstack project, http://cloudstack.apache.org, June, 2013

[17] CloudStack, A. Understanding Apache CloudStack. 2012 20/11/2013]; Available from: http://cloudstack.apache.org /software.html

BIOGRAPHIES

Pranjali Vilas Deshmukh1
M.E. research scholar, P G Department of Computer Engineering,D.Y.Patil School Of Engineering,Lohegaon,Pune. B.E. in IT (2009-10)From Shivaji University, Maharashtra. 4 years of working experience in teaching as lecturer.

Prof.Pankaj Agarkar2
is working as an Asst Prof in P G Department of Computer Engineering,D.Y.Patil School Of Engineering,Lohegaon,Pune. He has 19 years of teaching experience and he has published many research papers in international and national journals

