ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

Survey on Policy Based Framework for Smartphone Application's Privacy Using Multiple Profiles

Vinothini.S

Dept. of IT, IFET College of Engineering, Villupuram, India

ABSTRACT: Smartphone is the latest technology ruling the professional world. It plays vital role in accessing the documents on travel. Security and privacy related issues for smartphone application are still under research. By keeping the personal, private application and data separately using multiple profiles with in a single smartphone will reduce those issues. This paper delivers a survey about, the proposed system security and privacy on smartphone applications for the entire smartphone user. This is achieved through, a policy-based framework for enforcing software isolation of applications and data on the Android platform. It defines distinct security profiles with in a single smartphone. Time constrain for each profile will automatically change over to another profile showing only the selected applications. Dynamic switching from one security profile to another will control and access to applications and data.

KEY WORDS: security profile, android, framework, application control, multiple profiles.

I. INTRODUCTION

A smartphone is a mobile phone with an advanced operating system. Mobile phone users are rapidly switching over to smartphones as they offer personal digital assistant, media player and GPS navigation unit, touchscreen interface. Smartphones are very popular among the users because of its multiple smart features related to education, entertainment, business, etc. Companies design professional smartphones to complete most of the official works on travel like Windows phone with all MS Office feature. More and more companies nowadays provide mobile version of their desktop.Google Android is a Linux-based mobile platform developed by the Open Handset Alliance (OHA). Most of the Android applications are programmed in Java and compiled into a custom byte-code that is run by the Dalvik virtualmachine (DVM).

In particular, each Android package is executed in its own address space and in a separate DVM. Android supports an all-or-nothing approach, meaning that the user has to either grant all the selected application permissions or abort the installation of the application. However, security and privacy concerns about data and application leakage and loss have hindered the adoption of smartphones for corporate use.Securingan open platform requires robust security architecture and rigorous security programs. Android was designed with multi-layered security that provides the flexibility required for an open platform, while providing protection for all users of the platform.A policy-based framework for enforcing software isolation of applications and data on the Android platform provide the privacy. It provides an abstraction for separating data and apps dedicated to different profiles that are installed in a smartphone. For instance, corporate data and apps can be separated from personal data and apps within a smartphone.

II. RELATED WORK

In this section, we are focus on the different methods of losing the smartphone application and sensitive data. Android has highest market share of any mobile operating system. Most Android markets allow applications to be published without any verification. Count on self-regulating system of rankings and user reporting of malicious content.Application run inside sandboxed Dalvik VM. Application must request access to privileged information Presented to user at install-time. ButPermissions only tell user what kind of information can be accessed not how it is used.

[1] Thus Android Leaks – static analysis framework which Set a mappings between Android API and necessary permissions. The system provides Security analysis of 24,350 Android applications. Also implement taint-aware slicing

for Android callbacks and Compare prevalence of ad libraries and data leakage. [2] A key feature of modern smartphone platforms is a centralized service for downloading third-party applications. Smartphones are resource constrained but Third-party applications are entrusted with several types of privacy sensitive information. Context-based privacy information is dynamic and can be difficult to identify even when sent in the clear form as Applications can share information. Dynamic taint analysis is a technique that tracks information dependencies from an origin. TaintDroid is a system-wide integration of taint tracking into the Android platform which Extends tracking between applications and to storage. TaintDroid uses a combination of heuristics and method profiles to patch VM tracking state in which Applications are restricted to only invoking native methods in system-provided libraries. TaintDroid provides efficient dynamic taint tracking and analysis for Android.[3] Apex informs the user when it blocks an access as Runtime policy enforcement. Android Permission Extension (Apex) framework, a comprehensive policy enforcement mechanism for the Android platform. Apex gives a several options for the user restricting the usage of smartphone resources by different applications. Apex also allows the user to impose runtime Constraints on the usage of resources. Apex and the extended installer are both implemented in the Android source code with a minimal and backward compatible change in the existing architecture and code base of Android for better acceptability in the community. Apex allows users to specify detailed runtime constraints to restrict the use of sensitive resources by applications. The framework achieves this with a minimal trade of between security and performance. The user can specify her constraints through a simple interface of the extended Android installer called Poly. The extensions are incorporated in the Android framework with a minimal change in the codebase and the user interface of existing security architecture.

[4] Although Android’s permission system is intended to allow users to make informed decisions about their privacy on application but it is often ineffective at conveying meaningful information on how a user’s privacy might be impacted by using ansmartphone application. An alternate approach to providing users the knowledge on applications they install. There are four significant aspects to the design of the system. First, create a knowledge base of mappings between API calls and fine-grained privacy-related behaviors. Second, explain how this knowledge base is used to generate behavior profiles. Third, discuss how to develop a scalable system to produce such profiles for the available applications in Google Play. Finally, discuss how to make these profiles readily available to end users. Thus wishing to identify a set of privacy-relevant API calls and map them to appropriate behavior types which summarize the privacy implications of those calls. Start by extracting the API calls in our knowledge base from the application source code. In order to do this, use a tool called the Fortify Static Code Analyzer which is able to use the rules in our knowledge base to identify code in each application that matches our rules. [5] With their computing power and widespread adoption, smartphones promise to materialize the idea of ubiquitous computing. That made major companies such as Google, Apple, and Microsoft invest a lot of money and resources to increase the rate of innovation. One emerging trend is using Near Field Communication (NFC) for authentication, ticketing and secure banking. These applications have high demands on security. It is the task of the operating system (OS) to enforce the security of the system. Android has become the most popular mobile OS in terms of market share. L4Android: A Generic Operating System Framework for Secure Smartphones. This allows for highly secure applications to run side-by-side with the virtual machine. It is based on a state-of-the-art microkernel that ensures isolation between the virtual machine and secure applications. This framework can be used to solve four problems in current smartphone security.

III. CONCLUSION AND FUTURE WORK

Thus concludes that my approach provides compartments where data and apps are stored. These compartments are called Security and Privacy Profiles. Generally speaking, a security and privacy profile is a set of policies that regulates what applications can be executed and what data can be accessed. Profiles are not predefined; they can be specified and applied at any time. The ordinary system that exists currently will provide security and privacy only for limited number of application in the smartphone. When user using that application, each and every time it will ask the security code. But in our system separating the application using multiple profiles will provide the security and privacy for the user application. User can restrict the third party user not to access their important, private data or application. System asks the security code only at the time of changing the profile manually. But normally time is allocated for each profile and thus it change automatically. The system enhances the user comfort zone. Automatic profile change will consume the user time and their work.. In future try to implement this application in iPhone, also implement as the important default application while manufacturing. Smartphone is a private device, each user having some privacy on their smartphone applications and data. Simply saying that this survey is on privacy and security protection in smartphone application. Desktop with multiple users having permission to access only limited data enable on the screen. Likewise, smartphone having multiple profiles with the specified application to access with in a single smartphone according to the user wish. In future try to implement this application in iPhone, also implement as the important default application while manufacturing.

REFERENCES

1. C. Gibler, J. Crussell, J. Erickson, and H. Chen, "AndroidLeaks: Automatically Detecting Potential Privacy Leaks in Android Applications on a Large Scale,"
2. 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," Proc.Ninth USENIX Conf. Operating Systems Design and Implementation(OSDI '10), pp. 1-6, 2010. Demo for taintroid" http://appanalysis.org/demo/"
3. M. Nauman, S. Khan, and X. Zhang, "Apex: Extending Android Permission Model and Enforcement with User Defined Runtime Constraints," Proc. Fifth ACM Symp. Information, Computer and Comm. Security (ASIACCS'10), pp. 328-332, 2010.
4. https://web.eecs.umich.edu/~zmao/Papers/codaspy018-rosen.pdf
5. T.U. Dresden, and U. of Technology Berlin, "L4Android," http:// l4android.org/ 2014.] http://www.isti.tuberlin.de/fileadmin/fg214/liebergeld/spsm03-lange.pdf
6. A.R. Beresford, A. Rice, and N. Skehin, "MockDroid: Trading Privacy for Application Functionality on Smartphones," Proc. 12th Workshop Mobile Computing Systems and Applications (HotMobile '11), pp. 49-54, 2011.
7. P.B. Kodeswaran, V. Nandakumar, S. Kapoor, P. Kamaraju, A. Joshi, and S. Mukherjea, "Securing Enterprise Data on Smartphones Using Run Time Information Flow Control," Proc. IEEE 13th Int'l Conf. Mobile Data Management (MDM '12), pp. 300-305, 2012.
8. Unisys Establishes a Bring Your Own Device (BYOD) Policy, http://www.insecureaboutsecurity.com/2011/03/14/unisys_establishes_a_bring_your_own_device_byod_policy/, 2014.

BIOGRAPHY

Ms.S.VINOTHINI Currently pursuing B.Tech, in the stream of Information Technology at IFET College of Engineering, Villupuram, India. Her area of interests includes computer networking, Cryptography, Android application and java programming.