Implementation of Security Policies based on Security Profiles in Android System

Shakuntala P. Kulkarni[1], Sachin Bojewar[2]

[1]*PG Scholar, Department of Computer Engineering, ARMIET, Maharashtra, India*
[2]*Associate Professor, Department of Information Technology, VIT, Maharashtra, India*

Abstract - *In this modern era, users are shifting from computers to smart handheld devices and Smartphone to fulfill their computational needs. Smartphone are fruitful tools for increasing productivity of business. Now-a-days Smartphone come with added computational power and storage capacity. Smartphone let users perform various important tasks and help users to stay updated while on the move. Smartphone are proving to be very important tools for accessing messages, calls, emails, website browsing, etc.,. Companies provide employee owned Smartphone, so that he remains connected to office through his phone. Due to this productivity of companies increase as employees can work while on move. Smart phones connect to internet to access many web services. Smartphone face similar security problems while connecting to internet as the problems faced while connecting the computers to internet. If a corrupted application is downloaded by user then the entire phone may get damaged. Issues like data sharing, data leakage and loss have hindered the use of Smartphone for corporate use. For these security reasons many users carry different phones to meet their workplace and personal needs. In proposed system, Implementation of Security Profiles based on Security Policies in Android System is a policy based framework called MOSES used for enforcing isolation of application and data on android platform. Here multiple security profiles are declared in a single Smartphone. Each security profile has a set of policies which leads to control of access to applications and data. Profiles are not predefined and can be changed and applied any time. The proposed system also has automatic location wise activation of profiles.*

Key Words: **Smartphone, Security, MOSES, Security Profiles, Isolation, Android**.

1. INTRODUCTION

1.1 Overview

Smartphone provide users with a lot of functionality while users are on the move. Smartphone also provide similar features of that a modern day computer. Due to this reason the employees connect their Smartphone to their IT infrastructure where maximum companies now a day provide mobile version of their desktop applications. A lot of companies these days are encouraging concept of BYOD (Bring Your Own Device) policy where the employee is provided with an access to company's applications through their own smart phone. Several security concerns arise as user can download and install applications provided by third party on their smart phones. For example, malicious application can access SMS, mails, etc. from user's smart phones which might be containing company's confidential data. There are many malicious applications which cause data leakage from the phone without user's consent. This possesses a serious security risk to sensitive data where current security mechanisms are not effective for protection of data.

1.2 Mobile Virtualization

Virtualization technique was firstly proposed in 1960 by IBM [1]. Virtualization was solution to problems such as security and utilization of hardware resources. Virtualization is a framework which divides the resources of a computer into execution environments by applying concepts such as hardware or software partitioning, time sharing, partial or complete machine simulation, emulations, quality of service, etc. Virtualization will prove to be very useful in Smartphone as multiple Virtual Phones (VP) keep running where instance of one VP does not affect the performance of other. If one VP gets damaged and stops running then the user can easily switch to other VP and complete their work. Here environments are isolated from each other and are indistinguishable from the" bare" hardware. Hypervisor is responsible for providing such isolation and coordination of virtual machine activities. Virtualization used in traditional computers provide the following functionalities

- Increase in security
- Reduction in cost of application deployment

Virtualization in mobile phones will provide following benefits

- Separate communication subsystems from high-level application code.

1.3 Objectives

- A context is associated with each security profile by which it can be determined when the profile needs to be activated.
- End users need to specify contexts and profiles dynamically and for performing this, a GUI is used to create these security profiles and differentiate those to the level of single application and single objects.
- Switching between security profiles can be manual and in case of switching according to location or time, it can be automatic.

2. LITERATURE SURVEY

2.1 Taint Droid: An Information-Flow Tracking System for Real Time Privacy Monitoring On Smart Phone.

Currently, Mobile operating systems provide coarse-grained controls to check whether an application is accessing their private data. For example, if a user has allowed an application to access his location then that application may even send his location to some location based application which user may not have allowed. There is no way the user might know how is private data is being used in the mobile phones. TaintDroid [5] tracks the flow of private data in the mobile phones. TaintDroid assumes that downloaded data is not trusted data and it monitors how the applications access and manipulates user's private data. TaintDroid labels (taints) private data and monitors its flow. When tainted data is sent over network or leaves the system, TaintDroid keeps log of applications which sent tainted data over network. It presents feedback to user. It is hence useful for user to identify which applications are leaking data. [5]

2.2 Cells: A Virtual Mobile Smartphone Architecture

'Cells' [3] is virtualization architecture. It allows one to create multiple virtual Smartphone in one single physical phone. These virtual phones run simultaneously as if the other virtual phone does not exist. Even if one of the virtual phones is running any malicious application then other virtual phones remain unaffected. In Cells model, physical Smartphone has one foreground application and multiple background application. Cells use VoIP services. For security reasons, each virtual phone is isolated from the other phone. Foreground application has direct access where as background application has shared access to hardware. If foreground application requests Bluetooth connection then background virtual phones will not request Bluetooth connectivity. Different virtual phones on Cell run unmodified in different Android applications. [3]

2.3 The ThinVisor Mobile Device Virtualization Architecture

Smartphone are essential part of everyday life. Users carry multiple Smartphone; one for personal use and other for work. This is to safeguard the work related sensitive data from loss. ThinVisor [7] is a light-weight mobile virtualization technique. ThinVisor runs multiple Smartphone in a single physical phone. These multiple phones are called as personas. Personas act as individual phones. They have full access to the hardware like a single physical phone. There are different access rights to each persona. These access rights can be customized. For example,
1. No Access:- Application having no access rights, cannot access that feature of the mobile.
2. Shared access:- Foreground persona and background persona share an application.
3. Exclusive Access:- Only foreground persona has access to some hardware and background personas cannot access it.
ThinVisor was security measure from data loss. [7]

2.4 Taming Information-Stealing Smartphone Applications (on Android)

New privacy mode is provided by Taming Information-Stealing Smartphone Applications (TISSA) [4]. These modes are flexible and user controlled. Modes tell which application can access what data. Granted access can be revoked any time at run-time. [4]
Privacy modes in Smartphone protect private information of the user. User can install any untrusted application; user can also control its access permission. User can specify what an application can access. These access rights can be changed at run-time. User can also change the access right at the time of installation. Smartphone has constraints such as less memory etc. TISSA is very memory efficient as well as energy efficient. [4]

3. EXISTING SYSTEM

Smartphone allow users to perform many tasks while on move. Using Smart phones user can even work outside office. It is observed that, productivity of the company increases if its employees use smart phones and work outside office also. To work outside office, employee has to be connected to their IT database and infrastructure. Users mostly use two phones, one for personal use and other one for work. Several device manufacturers allow using of two SIM (Subscriber Identification Module) at the same time in same device. Since, users can install third party applications; there is a risk of data leakage. Work related sensitive data is present in the phone. Any third party installed application may leak sensitive data. Malicious application may access SMS, MMS, and E-mails etc. Even legitimate applications may access data that is not necessary for its functioning.

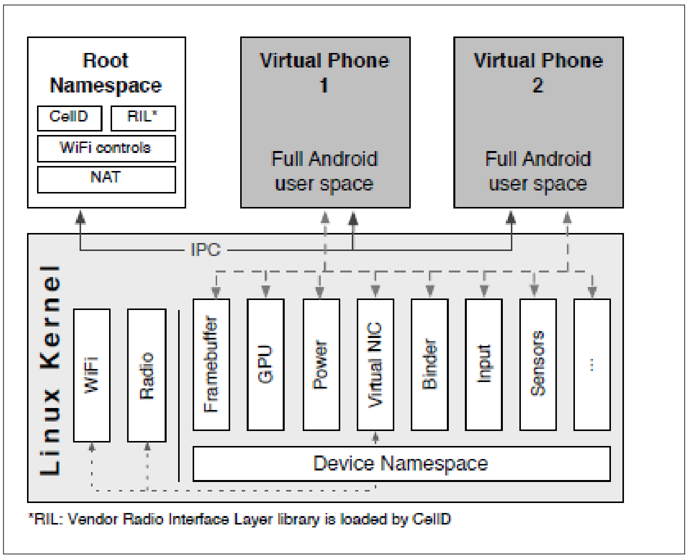

Figure -1 Existing System Architecture [3]

There is difference between mobile devices and traditional computers. Cells mechanism uses virtualization to isolate foreground application from background. Isolation is brought about by using device namespace and device namespace proxies. Device namespace provides Kernel-level based abstraction. Virtualization of hardware devices such as frame buffer and GPU is brought about by Kernel-level abstraction. Device namespace virtualizes Power Management framework. Cells proxy libraries provide User-level mechanism which virtualizes closed and proprietary device infrastructure. [3]

3.1 Disadvantage of Existing System

- Environments are hardcoded. Environments cannot be defined by users; they are predefined in the virtual machine.
- Switching among the users requires user interactions.
- Switching among environments requires time and power.

4. PROPOSED SYSTEM

In the proposed system, a mechanism is provided to separate data and applications related to different context installed in single device. Work related data and application can be separated from personal data and applications. Compartments are created in which data and applications are stored separately. Data and applications in one compartment are completely isolated from the other compartment. These compartments are called Security Profiles. Security profiles are set of security policies which explain what data and application can access. First environment may be for work related applications, trusted application and sensitive data. Second environment may be for personal use which may have personal data, recreational applications and untrusted application. The risk of leakage of sensitive information can be greatly reduced as long as applications from the second environment are denied the access to data of the first environment, the risk of leakage of sensitive information can be greatly reduced. Each security profiles (SP) are associated to one or more contexts that determine when the profile becomes active. Contexts and profiles can be easily defined by end users. Switching between security profiles can be done by user interaction or can be automatic. Switching between profiles can also be based on location. Automatic activation of Security profiles is the main feature of the proposed system. There can be a change in security profile. That particular Security profile is activated for which the context definition evaluates to true. Each security profile is given priority between 1 and 99. If for a particular context, more than one security profile becomes active then the security profile with higher priority becomes active.

4.1 Advantages of Proposed System

- There is automatic activation of Security Profiles.
- Contexts and Security Profiles can be dynamically and easily specified by users.
- Context Switching can be done by user interaction, automatically or location based.
- Loss of data is reduced.

5. MODULE DESCRIPTION

5.1 About

This module describes about proposed system. In proposed system, it is possible to create different contexts which are governed by security policies in a Smart Phone. It defines policies that control the access to applications and data. Profiles are not predefined. They can be specified by the user. Main feature of proposed system is dynamic switching between security policies.

5.2 Create Security Profiles

In this module user can create security profiles which are governed by security policies. Security Profile has security profile name, time, location and priority. Types of Security profiles that can be created are work, home, default etc. Priority value of security profiles lie between 1 and 99. When at a given instance two or more Security Profiles become active then the Security Profile with highest priority becomes active.

5.3 View Profiles

Profiles which are created are stored in the database. View Profile Module is used to view the stored Security Profiles. One can also view the time and location given as input while creation of the profile**.**

5.4 Applying Security Profiles

Security Profiles are created based on security policies. Contexts are the environment where there is a change in the Security Policy. When a particular context is encountered then Security Profiles corresponding to that particular context is activated. Context can be time or location. One of the contexts can be that at particular time, there should be a change in the security profile. Another context can be that when the user crosses a particular location, there should be a change in security profile. As Context change, another security profile is applied.

5.5 Permissions

This module assigns permissions to the application. There are two types of permissions. One is ALLOW and another is DENY. "ALLOW" permission allows applications to run in the background whereas "DENY" permission denies the application to run in the background process.

6. SURVEY RESULTS AND ANALYSIS

A survey was done to know about the view of Human Resource people about the concept of using "employee-owned" devices in the company. 11 HRs from different companies were surveyed. The questions of the survey are presented below:-

Q1:- Do you support the idea of "Bring Your Own Device "to company so that you can perform many of the company works from your own smart phone or tablet or any handheld device?

Q2:- Do you understand the distinction between personal data and enterprise data?

Q3:- Do you keep personal data and enterprise data separate in your own device?

Q4:- Do you support the idea of company taking control of your device during office hours and allowing access to only enterprise data during office hours?

Q5:- Would you prefer to install company owned application to your device so that it can limit your access to only enterprise data during office hours?

Q6:- Do you support the idea of employer providing you stipends to work on your own device instead of company providing you a device for working?

Q7:- How likely is it that you recommend the BYOD approach to others?

The result analysis of the survey is displayed below in tabular form:-

Table -1: Tabular Analysis of the survey

Question No.	Yes	No
Q1	10	01
Q2	08	03
Q3	08	03
Q4	07	04
Q5	09	02
Q6	11	00
Q7	10	01

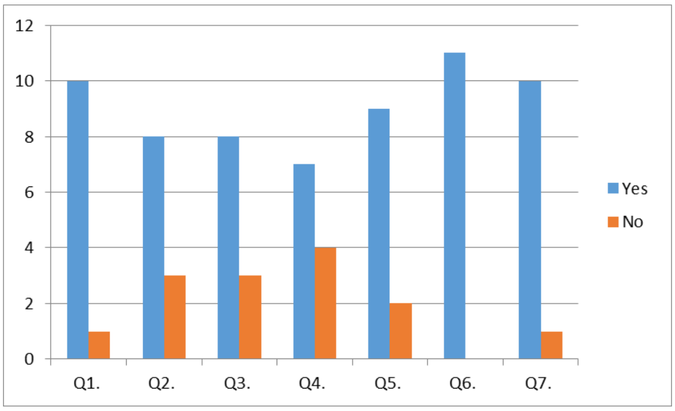

Chart -1: Graphical Analysis of the Survey

7. PERORMANCE ANALYSIS

To evaluate the performance of the application on the Android phone, different experiments were performed. All the experiments were performed on Lenovo K4 phone. To measure the energy overhead produced by proposed application, we performed following tests. We charged the battery of our device to 100%. We noted the change in battery for 100 minutes. 10 readings were taken in all. The readings were noted at an interval of 2 minutes. The values were noted and an average was taken for these 5 readings. During the experiment three system applications were run uniformly. Those were Calculator, Contacts and Browser and Email. For each application we performed common operations for all set of experiment. Addition of three digit number was done for Calculator, opening contacts and miss calling a number and opening Mumbai University home page

for browser. This experiment was conducted for two types of environment: one is without installing the application and other is after installing the application.
The readings were as follows:-

Table -2: Battery Reading

Time Interval (in mins)	Battery reading without application	Battery reading after installing the application
0 – 10	99	98
10 – 20	97	95.5
20 – 30	95.2	94
30 – 40	92.2	91.5
40 – 50	88.6	87.5
50 – 60	86.8	85.8
60 – 70	85.4	84
70 – 80	84	82.4
80 – 90	82	79
90 – 100	80.4	77
Mean	89.06	87.47

From the mean, it is obvious that there is no significant difference between Average Battery before installing the proposed application and average battery usage after installing the proposed application. Hence, we can conclude that the proposed application doesn't have any energy overhead. Graphical representation also shows almost overlapping line which denotes that there is no significant decrease in battery due to proposed application.

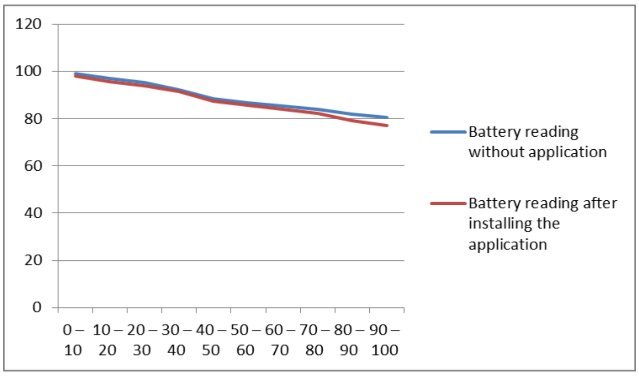

Chart -2: Graphical Analysis of Enegy Overhead

8. CONCLUSIONS

Smartphones play a vital role in increasing the business of the company as the employee can work for the company while on move also. But there are security issues like data leakage, data sharing etc. with the use of Smart phone. These issues were overcome by using the proposed Android application, Implementation of Security Profiles based on Security Policies in Android System. By using this application, user was able to create different profiles in the same smart phone. Different applications were governed by different security policies. Entertainment applications were governed by different profile and work related applications were governed by different profile. Due to this, there was no interaction between the applications and hence prevented data leakage and other security issues.

The proposed application was checked for energy overhead. It was observed that there is no considerable overhead, with application and without application. Survey was conducted to test the efficiency of the proposed application. Human Resource officer from different IT companies were asked to keep their views on the proposed application. 90% of the HRs responded positively and were ready to implement the application in their company. Thus, it can be concluded that, the proposed application is effective in preventing the security issues faced while using same Smart phone for corporate use and personal use.

REFERENCES

[1] Yury Zhauniarovich, Giovanni Russello, Mauro Conti, Bruno Crispo, Earlence Fernandes, "MOSES: Supporting and Enforcing Security Profiles on Smartphones", IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL 11, NO.3, MAY-JUNE-2014.

[2] G. Russello, M. Conti, B. Crispo, and E. Fernandes, "MOSES: Supporting Operation Modes on Smartphones," Proc. 17th ACM ymp. Access Control Models and Technologies (SACMAT '12), pp. 3-12, 2012.

[3] J. Andrus, C. Dall, A.V. Hof, O. Laadan, and J. Nieh, "Cells: A Virtual Mobile Smartphone Architecture," Proc. 23rd ACM Symp. Operating Systems Principles (SOSP '11), pp. 173-187, 2011.

[4] Y. Zhou, X. Zhang, X. Jiang, and V. Freeh, "Taming Information- Stealing Smartphone Applications (on Android)," Proc. Fourth Int'l Conf. Trust and Trustworthy Computing (TRUST '11), pp. 93-[1]107, 2011.

[5] 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.

[6] C. Gibler, J. Crussell, J. Erickson, and H. Chen, "Android Leaks: Automatically Detecting Potential Privacy Leaks in Android Applications on a Large

Scale," Proc. Fifth Int'l Conf. Trust and Trustworthy Computing (TRUST '12), pp. 291-307, 2012.

[7] Cellrox, "The ThinVisor Mobile Device Virtualization Architecture" Nov 2011.

[8] Shakuntala Kulkarni, Sachin Bojewar, "Survey on Smartphone Virtualization Techniques" International Research Journal of Engineering and Technology, Volume 02, Issue 04, July 2015.

BIOGRAPHIES

	Shakuntala P. Kulkarni received her Bachelor of Engineering degree in 2012 from University of Mumbai. She has been working towards Master of Engineering degree from University of Mumbai.
	Mr. Sachin Bojewar has 25 years of rich teaching experience and is currently working as an Associate Professor in Vidyalankar Institute of Technology