International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Survey Paper on Proactive Personalization through Abstraction for Smart Phones

Snehal Pundkar[1], Poonam Railkar[2], Parikshit N. Mahalle[3]

[1]Pune University, Smt.Kashibai Navale College of Engineering, Vadgaon(BK), Pune411041, India

[2]Pune University, Smt.Kashibai Navale College of Engineering, Vadgaon(BK), Pune411041, India

[3]Pune University, Smt.Kashibai Navale College of Engineering, Vadgaon (BK), Pune411041, India

Abstract: ***Recent years have witnessed the explosion in the use of smart phones. Many Applications are available in our AppStore of apple or Google Play Store. These Applications make the use of sensitive information of the user. Users do not have control over how their sensitive information is accessed by the Applications. Also in case the mobile search engine mobile users tend to submit shorter and more ambiguous queries. Current Android system is not capable of providing security to the user information. An enforcement system on the current android system is needed that will personalize the user web search and the Applications.***

Keywords: Android, AppGaurd, Appstore, Preference Manager.

1. Introduction

The goal of personalization is to deliver information that is relevant to the user. And the term proactive personalization refers to enhancing the personalization method without user intervention. It refers to studying user's behavior and automating the personalization process. In the recent years third party Applications for smart phones have become popular. In order to install the Application users are required to grant these Applications both the permission to access information on device as well as access the network. The network can be used to leak information to other Applications and advertising companies. The Application requesting permissions may use it for its core functionality or use it to share with advertising network or social network. For example a simple music player may ask access to your location at installation time and it can use this information to send advertising network. Users do not have control over how their information is used by the apps and to whom it is shared. Moreover there is requirement of the mobile search engine that will rank the search results according to user's requirement. Smart phones have powerful hardware with much functionality like camera, Bluetooth, microphones, GPS and can be used via APIs. Applications take use of this APIs to perform tasks to perform convenient but privacy sensitive tasks such accessing user's phone state, call log or location information. To personalize the user Applications and secure information Android and other mobile OSs implements security mechanism such as permission system. These mechanisms in practice proved to be insufficient with increasing no of malicious Applications targeting smart phones.

The paper is organized as follows: Related work in Section II which gives the details about existing system available for privacy preferences , Section III gives proposed system an enforcement system for android personalization and section IV conclude the paper.

2. Literature Review

In this section we analyze the android system working and its vulnerabilities to attack. Analyze App's privacy related behaviors and most frequently asked sensitive information. And study existing approaches to privacy preservation in android.

2.1 Android Permission System

Android is a linux based operating system. It is open source system designed for mobile platform. Android Applications are written in Java and compiled to DalvikExecutable byte-code format (DEX) [12]. Android system uses Application permission to protect information from other Applications. As seen in Fig 2.1, while installing an Application Smart Phone users has to grant some permissions user Application want to access. Permissions are given with little description but the purpose behind the permission is not specified. From the point of view of android programmer each permission gives access to one or more android APIs. Permissions are string defined by the system eg. Android permission INTERNET[1].An Application can access the functionality only if Application holds the permission. But this system is not sufficient to protect information from malicious Applications that reaches the component indirectly through chain of calls to innocent Applications. Fig 1 shows that before downloading an Application user has to grant permissions to the Application. Only then the user is able to download the Application.

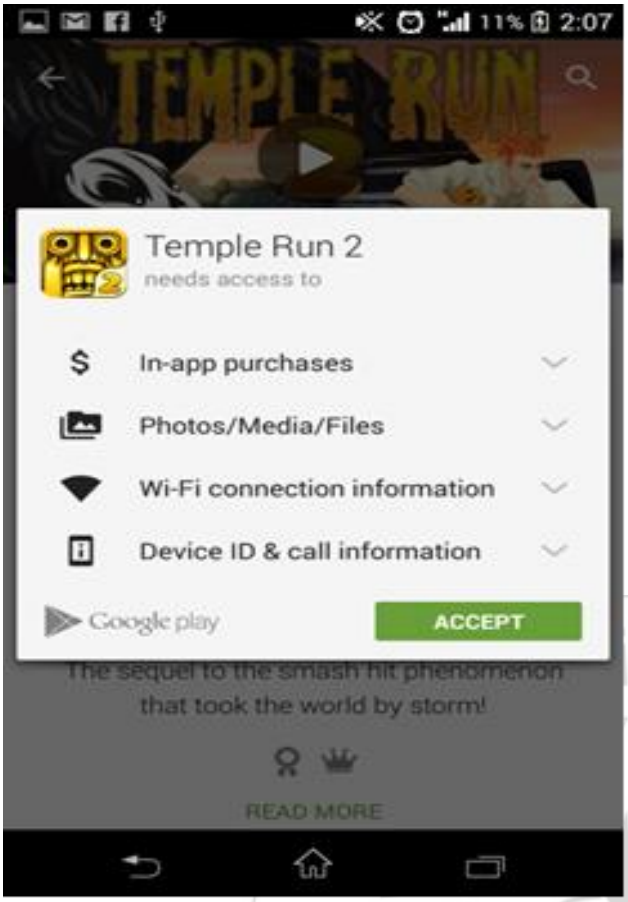

Figure 1: Before installing an Application user have to grant application with list of permissions it wants to access.

2.2 Analyzing Most Frequently Asked Sensitive Information

Android hardware has much functionality like GPS, camera, network etc. Application makes use of one or more of these functionalities to carry out tasks. For this Application requires permission to access these facilities at the installation time. The top 11 most frequently used sensitive information are [13]

INTERNET, READ_PHONE_STATES, BLUETOOTH, ACCESS_COARSE_LOCATION, CAMERA,ACCESS_FINE_LOCATION,GET_ACCOUNTS, SEND_SMS,RECORD AUDIO, READ_CONTACTS

2.3 Attacks on Permission Protected Information

Android Applications have access to much information that user may consider as private or sensitive. Android permission system is a powerful mechanism to protect this sensitive information. But the malicious Applications installed on the device can violet these permission policies.[5] One such method can be compromising communication mechanism such as Intents. Intents are used to exchange information between components. The attacks that compromise protected information can be:

Confused Deputy Attack: It relies on misconfigured Applications; components that interact with other Applications are invoked by unauthorized callers and allow them to access protected information.

Intent Spoofing: It affects the Applications not meant to communicate with other Applications. Malicious Applications are able to invoke the internal Activities, if the Application does not have necessary configuration.

Permission Collusion Attack: In this attack Application having access to limited permissions try to access protected information by sending and receiving intents to other Applications.

All these attacks are permission leak attacks. A malicious Application installed in a smart phone try to access the protected information by compromising android features like communication mechanism (called intents), or content manager.

2.4 Existing Approaches to Privacy Preferences

MockDroid [2] is a permission preference system which allows user to mock resources. This means system allows user to send fake information to the Applications to which they do not want to give access. For example user can give different information about the status of phone, call log or fake location details. For this it has to modify the Package Manager of the android. Package Manager store data and is the main way to share information between Applications. Data in the Package Manager is mocked or duplicated. The Application not having permission to access the information is provided with the mocked data at run time. eg. if Device ID is mocked then a random constant value is returned.

TaintDroid [3] is an information flow tracking system. It helps to track how third party Applications shares user's private information. The private stored data is labeled as 'taint'. The system monitors how third party Applications access third party Applications in real time. Sensitive information is identified as taint source. Dynamic taint analysis tracks how tainted data impacts other data in a way that may leak sensitive information.

SORBET[1] is an enforcement system that enhances the android permission system. The system can be retrofitted in the androids current architecture. Android uses the permissions which are in a string format (eg android.permission.INTERNET) to protect the components and APIs. It has included additional properties in the android permission system. The model has defined desired security properties which hold on SORBET and can be implemented on the top of android. Sorbet extends Android's permission labels to make them suitable for specifying coarse-grained information-flow policies, and enforces such policies at component and application boundaries.

PMSE[6] personalized mobile search engine has given a method to rank search results according to user preferences. The model can be used to profile user's behavior and personalize the search results accordingly. It has classifies

the user queries in content preference and location preference. It also takes into account users physical location GPS location to enhance the search result. User's query is first submitted to commercial search engines like Google, Yahoo etc. The clickthroughs of user are saved and further given to classification algorithm that will help to rerank the search result. This approach works in client server architecture. We can extend this approach to incorporate in our proposed proactive personalization framework.
Table 1 shows the comparison among existing approaches.

Table 1: Comparison between existing approaches to privacy preferences

	Personalized permission request	Permission purpose	Inter-app permission leak	Personalized web search	Privacy preservation in web search	URL based permission preference
SORBET	yes	Not effective	yes	no	no	no
MockDroid	yes	Not effective	no	no	no	no
TaintDroid	no	no	yes	no	no	no
PMSE	no	no	no	yes	yes	no
AppOps	Not effective	Not effective	Not effective	no	no	no
PPAS	yes	yes	yes	yes	yes	yes

3. Conclusion

This paper explains the existing approaches to privacy preferences. Some of them are not able to give the purpose behind the permission is. We believe that the user's permission preference is strongly depends on the purpose associated with permission. Thus we suggested the enforcement framework that can be retrofitted on the android smart phones. The suggested system will help user to personalize his Applications as well as web search while preserving privacy of the user.

References

[1] Elli Fragkaki,Lujo Bauer and Limin Jia.2014."Modelling and enhancing androids permission system" .
[2] Alastair r. Beresford, Andrew Rice,Nicholaas Skehin.2.11.MockDroid:Treadind privacy for application functionality on smartphones
[3] William Enck and Peter Gilbert.2010."Information Flow tracking system for realtime privacy monitoring on smart phones"
[4] DragosSbirlea, Michael Burke.2014."Automatic detection of inter application permission leaks"
[5] Kenneth Wai-Ting Leung,DikLun Lee and Wang-Chien Lee,2013."PMSE:A personalized mobile search engine"
[6] Reza Entezari-Maleki and ArashRezaei" A comparison of classification methods based on type of attributes and sample size"
[7] Androguard.Available: http://code.google.com/p/androguard/
[8] E.Agichtein,E Brill and S.Dumais,.2006."Improving Web search ranking by incorporating user behavior information"
[9] P.Hornyack, s. Han and J. Jung"These aren't the android your looking for: retrofitting android to protect data from imperious applications"
[10] M. Egele, c. Kruegel and E. Kirda .2011.PiOS:Detecting privacy leaks in ios applications"
[11] Google Inc.Android developers: Content Providers. http://developer.android.com/guide/topics/providers/content-providers.html
[12] Android. http://www.android.com
[13] Felt,A.P.,Chin,Hanna.2011."Android permissions demystified"