iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 5 **Issue:** IV **Month of publication:** April 2017

DOI: http://doi.org/10.22214/ijraset.2017.4206

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 5 Issue IV, April 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Privacy Preservation for Preventing Data Over-Collection in Smart Phones using Mobile-Cloud Framework

[1]M. Radhika, G. Bharathi[2], SK. Mastan[3]
[1, 2, 3] *Department of IT, L.B.R.C.E, Krishna, A.P*

Abstract: ***Smart city is meant to be the key for next generation urbanization process. In smart city all kinds of user data are stored in electronic devices to make everything intelligent and efficient. A smart phone is the most widely used electronic device in smart city because of its portability and also considered to be pivot of all smart systems. Thus all traditional systems in smart city, transform into smart systems and integrate their functions into smart phones. But, along with improvement of efficiency and reliability we are also facing several challenges with respect to privacy of user's data. The main challenge is data over-collection problem in smart phone apps. Data over-collection means collecting data more than its original function within permission scope given by user. To overcome this problem we are using a mobile-cloud frame work where user data is stored in cloud and apps work as data requesters and further the encryption/decryption operations also done with in cloud. In this we are using cloud storage service offered by Firebase. With this active approach we can reduce energy and also save storage space on smart phones which is very essential in today's mobile devices. By putting all user's data in cloud the security of data can also be greatly improved.***

Keywords: smart city, smart phone, data over-collection, firebase, mobile-cloud framework.

I. INTRODUCTION

Privacy and security are the new challenges of smart city to be solved. This is because, in smart city use all kinds of electronic devices instead of traditional approaches. Smart city consists of various smart things such as smart water, smart integration, smart public services etc. Moreover all user's data are stored in electronic devices to make everything smart and intelligent. Consequently, data has become the core of smart city as smart systems integrate their functions into smart phones. We entrust and provide private data to smart systems like smart phones to enjoy services we want. For example, If we want to buy book or products online, we need to provide them our name, address, email id and other personal details with the trust our data will be in secured. But without our permission it may be given to third party organizations. As a result, the security and privacy of smart phone data has become an important to achieve blue print of smart city.

Smart phones are providing wide variety of go-anywhere apps offering social, financial, and recreational services. Security risks happened in both android and ios systems. Now-a-days Smart phones are playing irreplaceable role in smart cities. They are used for various services as health assistants, entertainment, work secretaries etc. Smart phone apps bring enormous security problems. At present the most security hazard with the apps is that apps collect more data than required for its original function. This data is being collected within the permission scope given by the user to get the facilities provided by the app. This problem is stated as Data Over-collection problem. There are solutions for this problems but the existing solutions are passive measures which are used for monitoring and detecting the data over-collection problems. Furthermore, these solutions require tools to be run and also consume more energy on smart phones which is an essential feature of present smart phones. As a result to provide an active solution to data over-collection problem we are using mobile-cloud frame work where cloud services manage all user's data and also providing fine grained access permissions.

II. BACKGROUND

In this I am providing what type of data is being mostly collected by smart phone apps and what are the privacy threats that occur by sending this collected user's data to third party organizations. Although, smart phones are most widely used devices in blue print of smart city they are not competent of storing user's sensitive data. Further only coarse gained permissions are offered by the current mobile operating systems for regulating whether an app access users' private information but it is not going to consider how much

of private information collected is used. That is in coarse grained permissions a users' need to accept whatever app requesting in order to get the functionality of an app. Only few users are aware and understand the permission information during installation of app. After knowing that there data has been sent to third party app developers or others and they like to stop this the only option is to uninstall the app.

From the survey of AppThority, it is found that about 71% of free and 38% of paid android apps, about 32% of free & 16% of paid ios apps are sharing the collected user's data to network advertisements. And also 73% of free android apps and 43% of paid apps 61% of free ios apps and 38% of paid apps are sharing users' data to social networking sites. The often data collected by the apps are user's location, address book, IMEI number etc. The first and most direct risk involved with tracking user's location is physical security concerns. Users' tracks are easily exposed to someone who has their real-time and accuracy location data. Using some simple data mining methods, users' habits and customs are easy to be inferred. The risk in collecting address book is the contacts of user's have great worth in mobile devices. These data can be used by the app developers to expand their customer base. Thus these risks are causing problems to the citizens of smart city.

III. EXISTING SYSTEM

In this section, I would like to discuss the current solutions for solving data over-collection problem. Defence measures against hazards are mainly of two types active and passive approaches. Monitoring and detecting refer to passive methods[11] while prevention relate to active approach. At present the inherited techniques of security and privacy from distributed systems are used in smart city.

A. Monitoring and Detecting Methods

Egele et al. presented PiOS [2] to detect privacy leaks in iOS applications. In this static analysis is used to detect sensitive data flow to achieve the aim of detecting privacy leaks in applications in iOS. It mainly consists three steps. First, it reconstructs the control flow graph of the application to find code paths from sensitive sources to sinks. Second, it performs a standard analysis to find the paths in the control flow graph which connect nodes accessing sensitive information to nodes interacting with the network. Third, it performs data flow analysis along the paths to verify whether sensitive information is indeed flowing from the source to the sink.

The second method is TaintDroid, a system wide dynamic tracking system. It is an extension to the Android mobile-phone platform that tracks the flow of privacy sensitive data to third-party applications. TaintDroid will monitor applications and alert you when one tries to send personal identifiable data from your device. It won't tell you that this is good or bad, but just what is being sent and where it's being sent to.

B. User Aware Approaches

Enck et al. presented an approach named Kirin [10] where they automatically extract the security manifest of Android apps. This manifest is evaluated against logic invariants, before an app is installed. Users are only prompted for consent to install the app if these invariants are violated. Xiao et al. proposed a user-ware privacy control approach to reveal how private information is used inside applications [6]. They use static information flows and classify them as safe or unsafe based on a tamper analysis that tracked whether private data is obscured before escaping through output channels. Then the classified information enables platforms to provide default settings that expose users' private data only for safe flows, thereby preserving privacy and minimizing the burden of deciding for users.

These above two approaches reduce the operation burden for users who just have to choose whether to allow permissions to those apps or not but the authorization of coarse grained permissions may weaken the functionalities of these procedures. Since they are passive approaches they cannot protect users' sensitive data.

IV. PROPOSED SYSTEM

The methodology we are implementing is mobile cloud framework. Though mobile applications are widely used now-a-days due to shortage of resources for battery, storage and computation they are not able to perform well. So, using mobile-cloud framework to solve data over-collection problem not only improves the security for the sensitive data of citizens of smart city but also saves storage space of smart phones which is most essential feature of now-a-days mobile phones. In this model, users' data is stored in cloud and apps work as data requesters to cloud. The user performs the basic operations and apps work as data requester requesting data from cloud.

Volume 5 Issue IV, April 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

For cloud storage service we used Firebase technology. It is a mobile and web application platform with tools and infrastructure which helps to build high quality apps. Data is stored in JSON format and synchronized in real time to a every connected client. It allows secure access to data directly from client-side code. Cloud storage stores files such as images, audio, video and other user generated content. It has read and write rules which determine who has permission to your data, how your data is structured and what indexes exit. These read and write rules called security rules and authentication features protect users' stored in cloud storage. Further it performs the encryption operation of multimedia data what we try to upload to cloud and stores it as an uri in cloud storage. Further when we try to retrieve the image we need to get the uri where again the user authentication is checked to know they can perform the function or not. If so, then they can get the uri and download the uri and can retrieve the image. Firebase Real time Database offers full set of tools for managing security of your app. These tools make it easy to authenticate users, enforce user permissions and validate inputs. It also uses SSL (Secure Socket Layer) to provide security.

V. EXPERIMENTAL RESULTS

In experimental results we have 3 modules. We implemented these modules in our project.

Fig.1 Home Activity

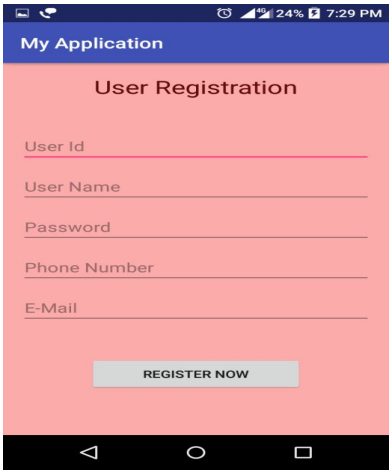


Fig.2 User registration

The first picture is home activity, where user first need to register to access app home or accessibility for apps in app home. Without registering if we try to access app home it displays "register first" toast message. Later after registering and clicking "register now" it displays "success" message.

Fig. 4 Apps Screen

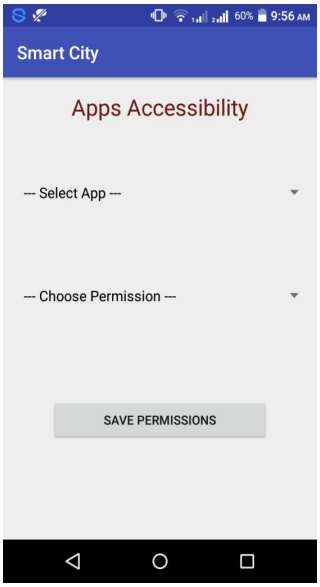


Fig 5.App Accessibility Screen

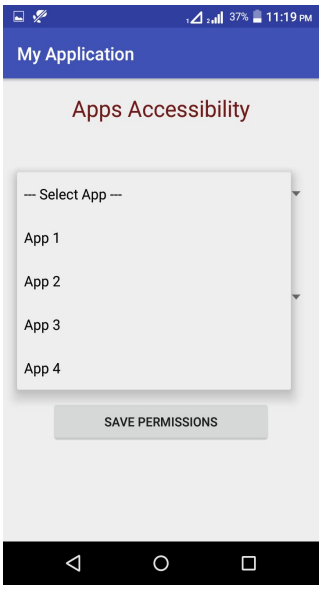


Fig 6.Selecting App

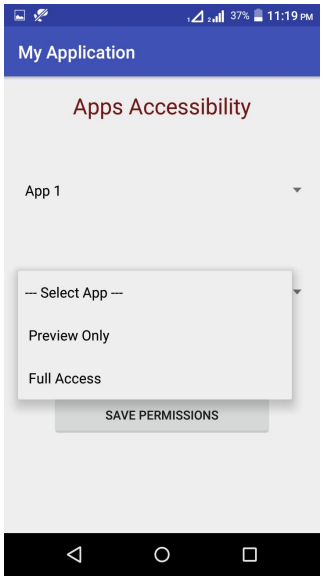


Fig 7.Setting Permissions

Volume 5 Issue IV, April 2017
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

These are the images of apps home and app accessibility in "Home Activity". Using Apps Accessibility screen we select an app of 4 different apps in app home and also set permission these apps either as “preview only” or “full access” or save the permissions.

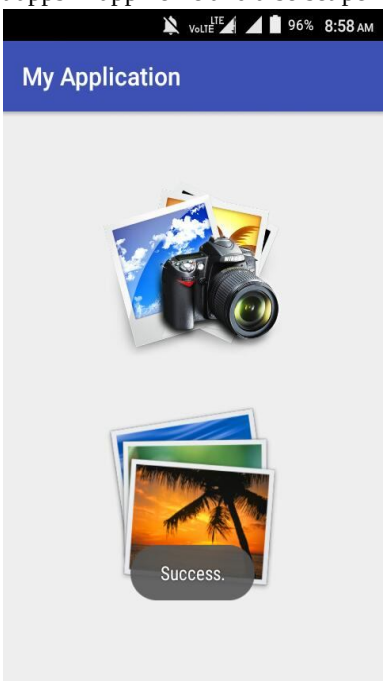


Fig 8. Image Uploaded To cloud

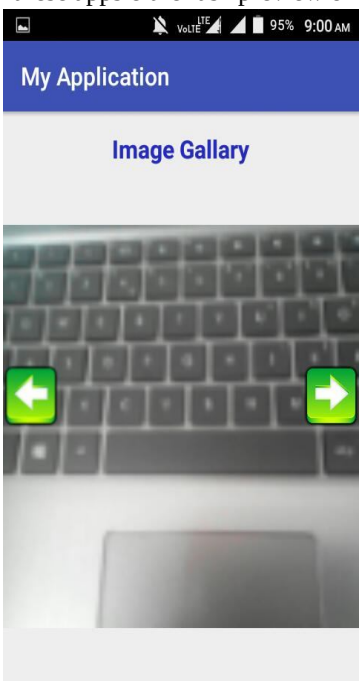


Fig 9.Image retrieved to from cloud

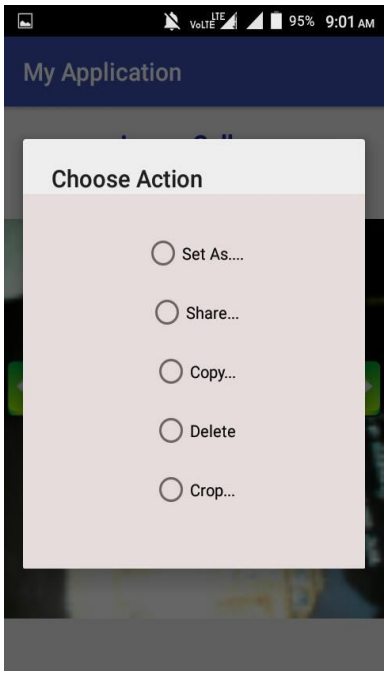


Fig 10. Can use edit options on img

In Fig 8. shows an app having " full access" upload image to cloud after capturing an image from users' camera and after successful uploading it displays "success" toast message. Fig .9 shows the retrieved images in gallery. Further Fig 10. shows that the images can be edited by using either of functions provided.

VI. CONCLUSION

Data over-collection in smart phones becomes the most severe potential privacy hazard in smart city. Unlike malwares, data over-collection is difficult to be solved, because this kind of behaviours are within permissions authorized by users. To maximize releasing users’ operation pressure and eradicating the data over-collection problem, we presented an active approach. Every app that wanted to use users’ data sent its request for accessing to the cloud, and the cloud access control service could provide detailed permissions for every app to every block of users’ data. Meanwhile the operations of encryption and decryption were achieved by cloud encryption/decryption service that saves computation resource of smart phone for dealing with these complex calculations. Finally, experimental result verifies the feasibility and advantages of our framework.

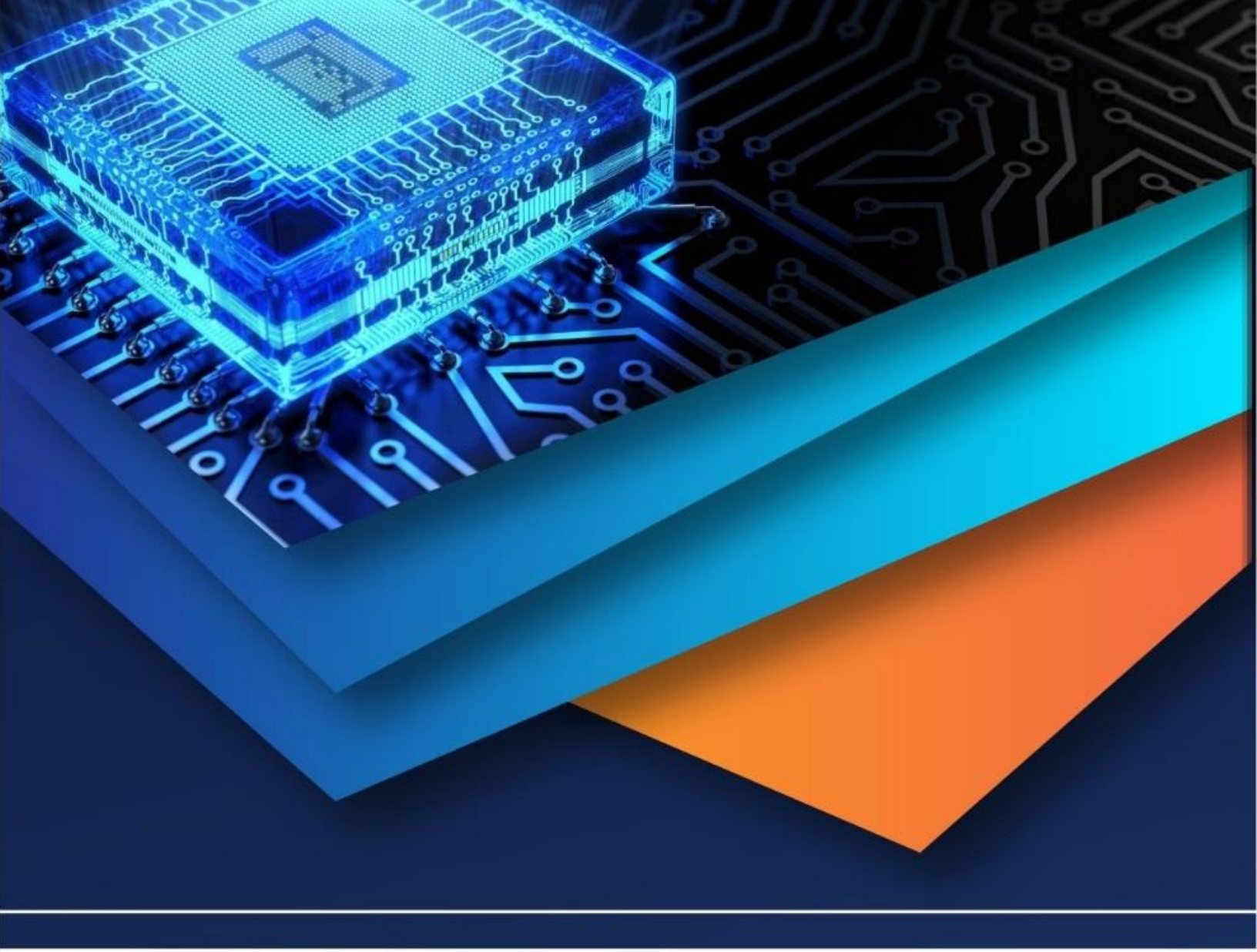
VII. FUTURE WORK

In future cloud computing is the best option where computing becomes more portable. But there are some issues such as security, reliability, integration and monitoring of clouds. Cloud localization is also an interesting area because public and private clouds are integrated to form a hybrid cloud and security is the main issue in this cloud. So in future we have to contribute to some more cloud security related algorithms. In future we try to extend our work in such a way that it can be used to provide security for other types of user data other than multimedia like his Contacts and text files etc.

REFERENCES

[1] Li, Yibin, et al. "Privacy protection for preventing data over-collection in smart city." IEEE Transactions on Computers 65.5 (2016): 1339-1350
[2] M. Egele, C. Kruegel, E. Kirda, and G. Vigna, “PiOS: Detecting privacy leaks in iOS applications,” in Proc. 18th Annu. Netw. Distrib. Syst. Security Symp., 2011, pp. 1–15
[3] Enck, William, et al. "TaintDroid: an information-flow tracking system for real time privacy monitoring on smart phones." ACM Transactions on Computer Systems (TOCS) 32.2 (2014): 5
[4] Bose, Abhijit, et al. "Behavioral detection of malware on mobile handsets." Proceedings of the 6th international conference on Mobile systems, applications, and services. ACM, 2008.

[5] Bettini, Anthony John, et al. "Quantifying the risks of applications for mobile devices." U.S. Patent No. 8,713,684. 29 Apr. 2014.
[6] X. Xiao, N. Tillmann, M. Fahndrich, J. De Halleux, and M. Moskal, "User-aware privacy control via extended static-information-flow analysis," in Proc. IEEE/ACM 27th Int. Conf. Automated Softw. Eng., 2012, pp. 80–89
[7] Zhou, Zhibin, and Dijiang Huang. "Efficient and secure data storage operations for mobile cloud computing." Network and service management (cnsm), 2012 8th international conference and 2012 workshop on systems virtualization management (svm).
[8] Breaux, Travis D., Daniel Smullen, and Hanan Hibshi. "Detecting repurposing and over collection in multi-party privacy requirements specifications." Requirements Engineering Conference (RE), 2015 IEEE 23rd International. IEEE, 2015.
[9] Lai, Junzuo, et al. "Attribute-based encryption with verifiable outsourced decryption." IEEE Transactions on Information Forensics and Security 8.8 (2013): 1343-1354.
[10] W. Enck, M. Ongtang, and P. McDaniel, "Understanding Android security," IEEE Security Privacy, vol. 7, no. 1, pp. 50– 57, Jan. 2009
[11] M. Qiu, W. Gao, M. Chen, J.-W. Niu, and L. Zhang, "Energy efficient security algorithm for power grid wide area monitoring system," IEEE Trans. Smart Grid, vol. 2, no. 4, pp. 715–723, Dec. 2011



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY