ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 3, Issue 4, April 2018

Privacy Protection for Preventing Data Over-Collection in Smart City

[1] Rashmi Chetry, [2] C.Rajabhushan
[1] PG Student ,Dept. of CSE, BIHER, Chennai, Tamil Nadu.
[2] Assistant Professor, Dept.of CSE, BIHER, Chennai, Tamil Nadu.

Abstract: ***In smart city, all kinds of users' data are stored in electronic devices to make everything intelligent. A smartphone is the most widely used electronic device and it is the pivot of all smart systems. However, current smartphones are not competent to manage users' sensitive data, and they are facing the privacy leakage caused by data over-collection. Data over-collection, which means smartphones apps collect users' data more than its original function while within the permission scope, is rapidly becoming one of the most serious potential security hazards in smart city. In this paper, we study the current state of data over-collection and study some most frequent data over-collected cases. We present a mobile-cloud framework, which is an active approach to eradicate the data over-collection. By putting all users' data into a cloud, the security of users' data can be greatly improved. We have done extensive experiments and the experimental results have demonstrated the effectiveness of our approach.***

.

I. INTRODUCTION

SMART city is meant to be and will be the next generation of urbanization. However, it brings some new challenges to be solved, such as security and privacy. The most arduous challenge about the cyber security and privacy of smart city is to ensure sensitive data secure. People living in a smart city use all kinds of electronic devices instead of traditional manual or mechanical equipments. To make the whole smart city efficiently, almost all these electronic devices need to be smart enough to recognize different users. Consequently, they must have the ability of storing and sharing data. To use various kinds of smart systems in a smart city, residents must offer their personal information to these smart systems. Residents must offer the information of their bank accounting numbers and passwords to shop online. They must offer the information of their addresses to receive packages. Consequently, data are the core of a smart city, because they consist of all users' information, which is invaluable in the Big Data age. Nevertheless, users are suffering the potential privacy leakage when they are enjoying the convenience brought by the smart city. However, there may be lot kinds of data leakage in smart city.

SMART CITY

At present, the most serious potential security hazard is that apps collect data more than enough on its original function while in permission scope, which we call it data over-collection. We survey current solutions to solve data over-collection problem and find that almost all these approaches are passive defense measures, which are a remedy approach after being hurt. To study the data over-collection, we firstly analyze how these behaviors happen and the risks they bring to users. We take location, photos, International Mobile Equipment Identity (IMEI) and Unique Device Identifier (UDID) as cases and find that large parts of apps are over-collecting users' data without noticing them. Then we discuss the two main factors of data over-collection in smartphones, which are operating system and permissions.

ISSN (Online): 2456-0448
International Journal Of Innovative Research In Management, Engineering And Technology
Vol. 3, Issue 4, April 2018

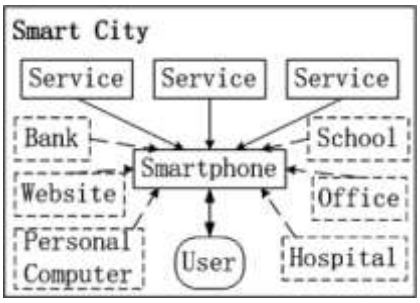

Fig. 1. The smartphone in a smart city.

RELATED WORKS

In this section, we discuss current solutions of solving data over-collection problem.

MONITORING AND DETECTING METHODS IN MOBILE

To detect privacy leaks in iOS applications, we use static analysis to detect sensitive data flow to achieve the aim of detecting privacy leaks in applications in iOS. PiOS checks an iOS application by 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 reachability 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.

USER-AWARE APPROACHES

We present an approach 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 con-sent to install the app if these invariants are violated. We also propose a user-ware privacy control approach to reveal how private information is used inside applications [1]. 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 two approaches reduce the operation burden for users who just have to choose whether to allow permissions to those apps that meet a certain degree of security requirements.

OFFLOAD IN MOBILE CLOUD COMPUTING

There are some other researches related to mobile cloud computing [2], [3], [4], [5] analyzed an environment in which computational offloading is adopted amongst mobile devices. We studied a framework to provide runtime support for the dynamic computation partitioning and execution of the application. This frame-work not only allows single user partitioning but also sup-ports sharing computation instances among multiple users. [6].The main aim of these researches is to achieve efficient utilization of cloud resources and to save mobile resources.

II. PROPOSED SYSTEM

We present a mobile-cloud framework, in which user's data is stored in a cloud. Cloud services are in charge of managing users' data and provide fine-grained access control and encryption/decryption operations which previously were implemented

in smartphones. Furthermore, we evaluate the feasibility and advantages of our approach by experiments. The two major contributions of this paper are:

We present a mobile-cloud framework to solve the data over-collection problem, which immensely improves the security and saves storage space of smartphones.

We solve the problem of quantifying security risk and design a benchmark to score apps mainly focusing on data over-collection behaviors. Using this benchmarks, we prove that our framework improves the security of smartphones significantly.

III. SYSTEM MODELS

In a smart city, data are the most important for users to keep their privacy not exposed. Furthermore, smartphones, as the pivot of a smart city, not only offer convenience but also undertake the responsibility of protect users' private data. However current smartphones are not competent for the job of protecting users' privacy. To analyze and solve this problem, we define some terms used in our research: Data Over-Collection - Collecting data more than enough on original function while within the permission scope. Security Risk - The security risk is not the immediate harm to the security, but the potential one. However, to some extent, the potential harm is much severer than the immediate one.

SECURITY AND CONSUMPTION MODEL

In smart city, users may store various kinds of data into their smartphones. Some of these data are sensitive, while some of them are not. To model security concern for various data, we divide users' data into several kinds of security levels (SLs). The most privacy data of users has the highest security level, normal data has medium security level, and public or shared data has the lowest security level. The most privacy data mainly includes personal information such as location, contacts, mails, messages, and some original photos. The public or shared data, which has the lowest security level, represents those data that users download from public servers or publish to the public. The rest data is classified into medium security level, which contains the app data, temporary computational data, and others.

After being set into different security levels, users' data can be stored using different storage service in cloud. The lowest security level data is stored in the simplest storage service, which barely provides fine-grained security permissions and complex encryption, but consumes lowest computational resource. As the security level grows, more precise permission and complex encryption service can be provided.

PERMISSION MODEL

Current smartphone operating systems just provide coarse-grained permission authorization, which is simply all or none. This is the chief culprit of data over-collection in smartphones. To quantify the effect of coarse-grained per-mission authorization towards data with various security levels, we add different permission authorizations into the security and consumption model.

According to cloud storage service, we set some different kinds of permission authorizations, which are to all, to specific and to none. Furthermore, the permission to specific can be defined by the proportion of accessing data. If an app has the permission of accessing N users' data, who has M data in total, the permission of this app is N=M. To describe the influence of permission for different security level data detailed, we take SL into consideration. We multiply N=M by SL, and then get the final permission (Perm) as

Perm ¼ SL N=M:

SECURITY RISK MODEL FOR DATA OVER-COLLECTION SECURITY

To formulate the security problem may caused by data over-collection, it is necessary to introduce security risk model. The data over-collection is a kind of potential risk, which is the product of security violation probability and the damage of security problem.

THE MOBILE-CLOUD FRAMEWORK DESIGN

It is impossible to enforce app developers not to share users' data with advertisement networks and other third party organizations, and it is unreasonable to expect that all smartphone users can understand permissions clearly and protect their privacy carefully.

In fact the security problem is created by ourselves, and to solve it we have to change our patterns of thought, not to deal with aftermath but to eradicate it. We present a mobile-cloud framework, shown in Fig. 2. In this framework, all users' data is stored in the cloud, and smartphones only deal with some basic operations of apps, such as managing the apps and showing the result of them.

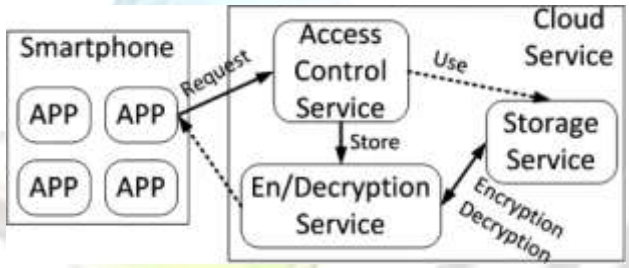

Fig. 7. The mobile-cloud framework.

CONCLUSION

The most direct improvement of our approach is saving storage space in users' smartphones. As shown in Table 1, users' native data are 26.5, 12.3, 48.5 and 40.7 percent of used storage space in their own smartphones. These native data includes photos, music, movies, videos and other data except app data and system used data. It is obvious that all or most of these storage spaces can be vacated to allow users to install more apps. We evaluate the security scores and risks of chosen apps in original and Mobile-Cloud framework environment. Then we evaluate the security risks of four smart-phones in original and mobile-cloud framework environment. Then we implement more detailed function in mobile-cloud framework and this function is fine-grained permission authorization.

TABLE 1
Usage of Experimental Smartphones

	Total	Used	App	Photo	Music	Movie & Video
Device A	64G	16G	10G	242M	4G	0
Device B	15G	13.8G	11.5G	1.4G	0	311.5M
Device C	10G	7.8G	529M	2G	270M	1.52G
Device D	38G	18.14G	1.72G	7.04G	254M	102M

REFERENCE

[1] 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.

[2] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "Clonecloud: Elastic execution between mobile device and cloud," in Proc. 6th Conf. Comput. Syst., 2011, pp. 301–314.

[3] D. Papamartzivanos, D. Damopoulos, and G. Kambourakis, "A cloud-based architecture to crowdsource mobile app privacy leaks," in Proc. 18th Panhellenic Conf. Informat., 2014, pp. 59:1–59:6.

[4] S. Subashini and V. Kavitha, "Review: A survey on security issues in service delivery models of cloud computing," J. Netw. Comput. Appl., vol. 34, no. 1, pp. 1–11, 2011.

[5] A. Fahim, A. Mtibaa, and K. A. Harras, "Making the case for computational offloading in mobile device clouds," in Proc. 19th Annu. Int. Conf. Mobile Comput. Netw. Miami, FL, USA, 2013,pp. 203–205.

[6] L. Yang, J. Cao, Y. Yuan, T. Li, A. Han, and A. Chan, "A frame-work for partitioning and execution of data stream applications in mobile cloud computing," SIGMETRICS Perform. Eval. Rev., vol. 40, no. 4, pp. 23–32, 2013.