PRIVACY ON DATA OVER-COLLECTION IN SMART CITY

Minu Mariyam Panicker
M.Tech Student
Dept. of Computer Science and Engineering
Lourdes Matha College of Science and Technology
Trivandrum, Kerala
minumpanicker292@gmail.com

Smitha J.C
Associate Professor
Dept. of Computer Science and Engineering
Lourdes Matha College of Science and Technology
Trivandrum, Kerala

Abstract—Smart city can be considered as a big information system which consists of smaller subsystems connected to other systems. The challenge of making smart city is becoming very complex. The function of smart city is development of security structures. Smart phones are used as one of the commonly used electronic device to store data of the users. But smartphones are more vulnerable to privacy leakages for data over-collection. Data over-collection does the purpose of storing large amount of data to users’ app. To prevent privacy, this paper proposes a mobile cloud framework, which uses cloud to store different sizes of data. It also shows some experimental results by showing in real scenario.

Keywords—Smart city, Data over-collection, Privacy, Mobile cloud framework.

I. INTRODUCTION

Smart city has become important for today’s life. The metropolitan can be considered as smart. The primary goals of smart city consists of offering digital means for supporting social needs in all daily transactions, to adapt the citizens to the notion of the information society and to collect information from the public departments and citizens in order to support sustainable growth of the city. The smart city is designed and constructed by highly advanced technologies which electronics sensors, and networks. Smart city can be considered as a big information system which consists of smaller subsystems connected to other systems. The challenge of making smart city is becoming very complex. Usually the smart city consists of different smart systems as shown in figure 1. The smart systems can easily recognize by giving a person’s personal information such as bank account number or any other personal ids to smart systems, for example, shopping online, online transactions etc.

To make the city smart electronic devices such as smart phones are used. Smartphones have become increasingly popular because it provides all-in-one convenience. Using smartphone people can access to internet everywhere. They can pay online bill, can do transactions etc. But the major challenges related to these smart systems are security, privacy and protection. One most security threat related to smartphone is viruses. Viruses damages the systems of smartphone which may lead erasing of important documents needed for smart systems. Even though there is a defense against the smartphone viruses, which is smartphone antivirus software, it has some limitations. First, a smartphone has limited capacity of storage and battery capacity, which can hamper the effectiveness of on-device antivirus software. Second, as smart phones may not be always connected to internet, it is challenging to distribute virus signature files to smartphone. So a virus detection system is proposed known as smart siren [1].
To store sensitive data such as bank account number, password, smartphones need security to potential users for storing data. The most potential hazard is storing enormous amount of data more than its capacity. They are known as data over collection [3]. To analyze data over-collection we must analyze the risks that bring to users. The risks are also related to main factors of data over-collection in smartphones which are operating system, permissions [6].

Though smart cities are facing many problems like interoperability and cost efficient technology along with technical issues, the concern for security and privacy is very important. Information security plays major role in security and privacy [4] to protect information from viruses, attacks, frauds etc.

This paper proposes different ways to protect from data over-collection with the use of cloud framework and also to provide security and privacy.

II. RELATED WORK

In a smart city, security and privacy issue consists of several aspects, such as, sensitive information, communication protocol, key management and authorization. Most widely used techniques are inherited from distributed systems. Smart grid is an example for these techniques [2]. There has been extensive research on the security and privacy issues that arise from the Advanced Metering Infrastructure (AMI) in which smart meters record fine-grained energy consumption measurements and send these to other external entities. For the privacy preserving smart metering, several privacy preserving protocols have been proposed [5]. AMI is where the only part that arise privacy concerns. In addition to AMI protocols, there is another architecture known as Demand Response protocols for managing energy consumption. There are specific types of DR such as demand bidding (DR-DB) protocols that has a to and fro communication with consumers and entities for reducing their energy consumption. As it is bidirectional, it addresses various security and privacy issues. The Open Automated Demand Response (Open ADR) specification is an example that can be used in DR communication [7]. There are two adversary models used for DR systems. Dolev-Yao (D-Y) model has control over communication network. This model can be used for eavesdropping, or block false messages, synthesize the message etc. This model is based on cryptographic systems, so it is assumed that it can neither read encrypted messages without the correct decryption key, nor reverse cryptographic hash functions. Second model is Honest-But-Curious model. It is based on the terms of sending and receiving messages. It is also limited by constrains of cryptographic systems. It usually involves linking messages from the received messages. It is also referred to as semi-honest model.

Smartphones play major role for smart city. But the main challenge for smartphones is viruses. Viruses damages the systems of smartphone which may lead to erasing of important documents needed for smart systems. Even though there is a defense against the smartphone viruses, which is smartphone antivirus software, it has some limitations. First, a smartphone has limited capacity of storage and battery capacity, which can hamper the effectiveness of on-device antivirus software. Second, as smart phones may not be always connected to internet, it is challenging to distribute virus signature files to smartphone. So a virus detection system is proposed known as smart siren [1]. The goal of smart siren is to halt viruses which out brakes by minimizing the number of smartphones that will be affected by a new release virus. Smart siren does the protection of user privacy. It addresses the activities of their phone to proxy by an anonymous and ticketed report submission scheme. It also prevents virus or attacker from attacking privacy mechanisms of phone. Smart siren architecture consists of a large set of smartphones and a proxy that interacts with the smartphones through cellular networks.
smartphone runs a log which gives repeated signals to the proxy informing about behavioral status of the phone. When any viral activity is been shown the proxy alerts the phone with suspicious activities. It also alerts other smartphones if it is infected with virus.

For solving issues related to data over-collection, the existing techniques use distributed systems. They prevent privacy leakage in iOS applications [3]. To prevent sensitive data flow they used static analysis. It is done in a way that, first it constructs a control flow to find the paths related, and then it performs a standard reachability analysis which connects to the network to give sensitive information. Then finally it performs data flow analysis along the entire path. This helps in tracking information flows of smartphones. But running these applications, leads to energy consumption.

People need to know whether the data stored can be shared or not. PEOPLEFINDER is such an application [8]. The main objective of this application is to better understand the privacy related to sharing the data. Location tracking is usually done for the privacy keeping. But it creates certain problems related to time, request of the user etc. Certain studies have been done which shows that functionality that increases user awareness can contribute to the definition of more accurate policies. PEOPLEFINDER shows how these studies have refined their preferences based upon their requests. The PEOPLEFINDER applications rely on Policy Enforcing Agents (PEA) to handle queries about their locations. The user’s PEA operates based on some conditions to check their location with PEOPLEFINDER using either a mobile phone client or the PEOPLEFINDER web site. PEOPLEFINDER is available for Windows Mobile cell phones and for both PC and Apple laptops.

III. BACKGROUND

A. The Rise of Smart Cities

Over last two decades, the urban has grown to big cities which lead to smart cities. It has become centers of big companies, social and educational facilities etc. over half of the population lives in cities. The use of smart systems leads to more smart cities. The key infrastructure based on smart city includes, network of sensors attached to real world object, networks of digital communication, high capacity cloud based infrastructure etc. There are certain activities that can be identified for define a smart city, such as, smart governance, smart infrastructure, smart building, smart connectivity, smart healthcare etc [9].

B. Smart City Operations Related to Data Over-Collection

During installing an application, there is certain information given for users. But users miss that information. Current mobile phone operating systems provide only coarse grained permissions. They will be some hidden privacy issues and risks. The following shows some apps based on data over-collection and its risks.

a) Location Tracking

In today’s life, most frequently used application is location tracking app. It can be used in Social Network Service (SNS). While using these apps, users are warned whenever an app intends to capture user’s location, they usually choose to allow. The risk associated to this application is physical security concerns. Users can be easily tracked by knowing real location. The second risk is about corporate, government, and martial espionage concerns. Fig. 2 [3] shows a tracking location in an ios.

![Fig 2. Detailed location data tracked by the frequent location service in an ios.](image)

b) Accessing Photos

Users widely access photos not only for storing but also for sharing in social networks. There are some other kinds of apps also access users’ album, such as cloud storage app, wallpaper app, and customized album app. Users use these apps for all parts of the mobile. They provide coarse grained permission for the apps which is always one-time operation. If the users use these apps and allow permission once, for lifetime this permission
will be stored, they cannot be changed. The risk associated with these applications is, with the data over-collection the third part organization can access the photos and can misuse them.

c) Accessing Address Book

Address books are that can store a person’s email address, phone number etc. Users can also modify the address book. The address book also includes user names, physical address, phone numbers, email addresses, and other notes. Similar to photos, address book provides only coarse-grained permissions. The risks associated with this application is, these data can be used by the app developers to expand their customer base, and can be used by third party organizations to market additional mobile apps. Furthermore, the data over-collection of contacts can bring potential corporate espionage.

d) Accessing Calendar

The calendar apps are already stored in mobile. So it’s impossible to uninstall the app. These apps get easily can easily get permissions to users’ calendar. There are some other apps such as lifestyle apps and travel apps which get the permission to users’ calendar. The main risk associated with calendar based on data over-collection is corporate espionage. The calendars have important dates for certain meetings or dates which a third party does not wish to know. They can easily get to know.

IV. PROPOSED APPROACH

To analyze the privacy certain system models are proposed. They are security model and permission model.

A) Security Model

Smartphones store different kinds of data. Some of them are sensitive and some are not. To model security concerns for various data, different kinds of security levels (SL) are considered. The data which is having more privacy is given with high security level and the data with low privacy is given lowest security level. Most privacy data include personal information such as location, addresses, number etc. The public or shred data is given lowest privacy level. The rest of the data is given medium security level. After being set into different security level, users’ data can be stored using different storage service in cloud. The data which is having lowest security level is given simplest storage service. As the security level increases, the storage level increases. Along with security, resource consumption is also computed. There is a linear relation between consumption and encryption complexity. A parameter $\alpha$ is set to express the linear relation between security level and execution time (T), and b to express the linear relation between energy consumption (EC) and complexity (CX) as follows [3],

$$ SL = \alpha \ast T, \quad EC = \beta \ast CX, \quad T = CX $$  \hspace{1cm} (1)

B) Permission Model

Till now, for all applications coarse grained permissions were given. This is the main culprit of data over-collection. So to quantify these permissions, different permission authorizations are employed. They are, to all, to specific and to none [3]. 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, take SL into consideration. The final permission will be:

$$ Perm = SL \ast N / M $$  \hspace{1cm} (2)

The following figures shows relation between security and amount of data over-collected with different values of SL.
c) Mobile Cloud Framework

Cloud storage offers big amount of storage space resources. Cloud provides services like Infrastructure-as-service (IaaS), Platform-as-Service (PaaS) and Software-as-service (SaaS) to its clients [10]. Secure cloud is a reliable source of information. The cloud can be protected by making sure data is available for customers, delivering high performance for customers etc. For safety, the providers must provide a supporting system for the client. The problems faced by cloud computing are data protection, authentication, data verification, infected application and availability.

In model cloud framework, users’ data are stored in a cloud. Cloud services provide control over fine-grained access control and encryption/decryption operations. The following figure shows a mobile cloud framework.

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. The encryption and decryption of data can be finished in cloud and apps work as Data Requester requesting data from Cloud and it can be implemented by CP-ABE framework [11].
Two algorithms are used to illustrate the workflow. For example, take Facebook. It can share photos, videos, data, etc. First algorithm is that users (UR) use this app to access his/her camera for taking an instant photo with location information and to store this photo in the cloud storage.

**Algorithm 1: An App Stores Data into Cloud**

Input: ID1, ID2, data.

1: Judge the type of this access request by ID1, get the result T;
2: if T = = hardware then
3: UR authorize the permission to this app;
4: else
5: Send request to Access Control Service including ID1, ID2 and data;
6: Access Control Service judges whether this app has permission by ID1, ID2, data and accessControlList, get the result P;
7: if P = = true then
8: Encryption Service encrypts data;
9: Store data into Cloud Storage with label ID1;
10: else
11: return;
12: end if
13: end if

Here user takes a photo and do sharing. First decide whether this app can be accessed by users’ camera. So two set of permissions are taken: access for data and access for hardware. If the request is for accessing, set the permission as authorized; otherwise send the request to cloud for judging. In second algorithm, user wants to share the data. To authorize various apps with different fine-grained permissions, the Access Control Service has lists about every operations of every app. When receiving a request from some app for permissions, it check these lists to find what specific permissions are authorized to this app, and return the result of yes or no.

**Algorithm 2: An App uses Data in Cloud**

Input: ID1, ID2, preview information of requesting data PD.
Output: concrete content of requesting data D

1: Send request to Access Control Service including ID1, ID2 and PD;
2: Access Control Service judges whether this app has permission by ID1, ID2, PD and accessControlList, get the result P;
3: if P = true then
4: Access Control Service sends request with ID1, userId and PD to Storage Service;
5: Storage Service finds the encrypted data by ID2 and PD;
6: Storage Service sends data with ID1 and ID2 to Decryption Service;
7: Decryption Service checks the permission authorization again by matching ID1 with data and return P1;
8: if P1 = true then
9: Decryption Service decrypts data to D;
10: return D;
11: else
12: return none;
13: end if.

**V. RESULTS**

The results obtained can be compared by a real scenario. Use four real smartphone and one simulative cloud to build a mobile cloud environment. Two scenarios are set: in original environment and in mobile cloud framework. Then choose some apps from Google Play to score their security about location, photos, and contacts in two scenarios. The security scores and risks of chosen apps can be evaluated by setting four different score degrees. Do not collect data: 110; do not transmit data: 80; transmit data to app developers: 50; transmit data to third parties: 15. The results show that all apps score under 60, while in Mobile-Cloud framework apps can score more than 64. The same is for security risks. The following graph shows the experimental result of security risks [3]. It clearly shows that the security risks of smartphones in mobile-cloud framework are lower than in original environment.
VI. CONCLUSION

Data over-collection is one of the most dangerous hazards in a smart city. As there is need for electronic devices for making a city smart, it becomes more hazardous. Electronic devices such as smartphones are not always reliable. They can cause damage, or can access permissions through the internet. To maximize the security for data over-collection, an active approach is used, where all the data are sent to the cloud. Security levels are provided according to the security of the data. They can allow or deny the permission. So that there will be less chance for third parties to access the data. The approach used here gives promised services.

VI. REFERENCES