Implementing Peer-To-Peer Data Sharing in Mobile Crowdsensing along with Data Protection and Data Recovery Algorithm

Elslin Bernisha J M [1], Muthulekshmi L [2], Leshmi M S [3], Muthuselvi M [4]
[1, 2, 3] Department of Computer Science and Engineering, University College of Engineering, Nagercoil, Tamil Nadu, India.
[4] Assistant Professor, Department of Computer Science and Engineering, University College of Engineering, Nagercoil, Tamil Nadu, India.

Abstract – Mobile computing is a technology that allows distribution of data, voice and video through a computer or any other wireless devices without having to be connected to a fixed network. Mobile Crowd Sensing (MCS) is an application that runs on consumer mobile devices such as GPS, smart phones, and car sensors. MCS is to collect and share data about the user or the material world, either attractively or autonomously, towards a common goal. In server-client Mobile Crowd sensing it lacks from the more operational cost on the server for storing and processing more amounts of data. Peer-to-peer data sharing can reduce the server's cost by operating the user devices. This system provides amount for users due to better data quality. For the purpose of data protection and recovery, the two algorithms used are Remote Backup Algorithm and AES Algorithm.

Index Terms – Mobile Crowd Sensing, Peer-to-Peer Data Sharing, Incentives, Backup Cloud, Data protection.

1. INTRODUCTION

Mobile Crowd sensing (MCS) is one of the quickest developing sensing applications of cell phones, for example, Smart phones, workstations and sensor-prepared vehicles and their inbuilt portable sensors. In MCS, all detecting information is gathered by a numerous number of portable clients utilizing their cell phones. Because of the base cost and vast scope, Mobile Crowd Sensing has actualized in numerous applications, for example, activity observing and condition monitoring. In MCS applications basically takes a shot at the concentrated design, where the client sense and report the information to focal server, at that point forms and disperses the information to those clients sent demand to get to the information. Server-customer engineering isn't reasonable for the application with a large amount of clients and a lot of information demands, because of the high handling cost on the server for information trading, preparing, and capacity. Clone2Clone and P2P Cloud, contain distributed information sharing by concentrating the detecting and handling capacities of cell phones. The calculation and capacity process on the server to the appropriated cell phones, offering ascend to portable Crowd detecting with P2P information sharing. In a P2P-based MCS framework, the detecting information may not be accounted for to server and spared in the server; on the grounds that the detecting can be spared and handled rapidly in portable clients' gadget and offer information between clients straightforwardly with no brought together server. The server, in customary P2P systems, is mostly used to monitor every client's information gathering data which information they contain and arrange association data, for example, IP address of every client. With this material, the server can help clients to interface and offer their data with each other. Information sharing among clients can be handled in view of the nearby associations through Wi-Fi or Bluetooth.

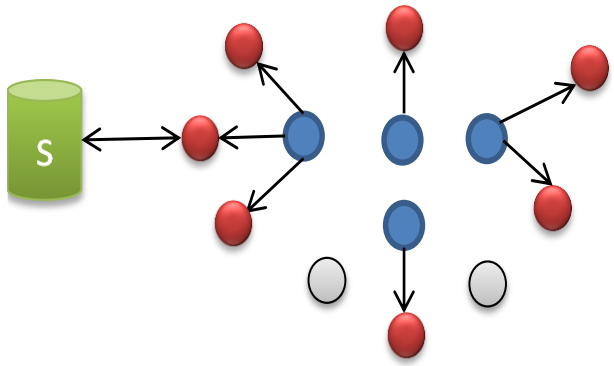

Fig.1 denotes a P2P-based MCS model, where the blue users collect data and share the data with red users through local Wi-Fi or the Internet, and the server is only responsible for the necessary control information exchange with users which establishing a network connection between users. There are many P2P- based MCS systems are use nowadays , including MPS Data Store, SmartP2P, and LL-Net. These studies concentrate only on some issues in P2P-based Mobile Crowd Sensing such as how to store data in distributed manner and to search the distributed data efficiently, economic issues and none of them considered the data security and backup . Due to the cost involved in data

sensing and sharing, users have incentives to communicate with each other for data quality. Sometimes data can be loss during transmission there is no backup system to retrieve data. This improves our study of the data security and backup issue in the P2P-based MCS system. Quality denotes the device state and the user preference. Location of data can be obtained through using GPS, Wi-Fi, and cellular networks, with decreasing levels of accuracy. Compared to Wi-Fi and cellular networks, continuous GPS location provides the most accurate location information, while draining the battery faster. The consideration of data quality, security and backup mechanism improve overall system performance.

2. RELATED WORK

Data sensing cost and the data transfer cost incentive becomes extremely important [1]. P2P reduce the server's operational cost sensing data is saved and processed in mobile users devices in a distributed fashion. Participatory sensing removes the cost of installing and maintaining sensors and their inter-connection, while achieving much broader geographical coverage [2]. LL-Net (Location-based Logical Network) achieves efficient propagation of location dependent queries issued by context-aware services. LL-Net is a P2P network based on the locations of peers [7]. Remora: a Smartphone-based Body Sensor Network activity recognition system which shares sensing resources among neighboring BSNs. Compared to other resource sharing approaches, Remora provides both increased accuracy and significant energy savings [9]. From these we note that no one has concentrated on backup, so we introduce backup algorithm such as remote backup algorithm is used to prevent from loss of data.

3. NETWORK MODEL

We consider an information quality P2P-based MCS display with an arrangement of versatile clients, who can detect a few information in a specific territory and offer the detecting information with each other in a disseminated P2P way. Every datum alludes to some of particular data at a specific area and time. Each client has the ability to detect a particular territory comprising of one or multiple data, contingent upon components, for example, her versatility, gadget write, and vitality spending plan. We consider an arrangement of various information, which can be utilized by one or numerous detecting applications. Each client can acquire her intrigued information in two ways initial one acting as a information sensor and detecting information specifically from physical world, and second path going about as an information requester and asking for information from an information. The latter case may happen when the user is not able to sense the data by herself, or when the user's sensing cost is very large. The data sharing among a data sensor and requester can be based on local Wi-Fi or Bluetooth connections or the Internet connection. To facilitate such data sharing, the server needs to keep track of each user's network connection information such as IP address and data occupancy information similar as in the traditional P2P system. In P2P system the server does not need to store and process the sensing data. All the data processing and information will be conducted by the associated apps on the mobile devices.

4. USER BEHAVIOR

To obtain the data, a user can choose to be a *data sensor* that senses the data directly from physical world or a *data requester* that requests the data from a sensor. The user can also choose to be a *stranger* that neither senses nor requests data.

- *Sender*: Data sensor, the user senses the data directly with a specified quality with some sensing cost. Meanwhile, the user can share the sensing data with others to obtain some incentive.
- *Receiver*: Data requester, the user requests data with a desirable quality from a sensor that has sensed the data already. Such sharing introduces data transmission cost. The requester needs to bear all of the transmission cost, and provide some additional incentive to the sensor.
- Stranger: The user neither senses the data nor requests data from others. This may occur when the user is not interested in the data or the cost of obtaining the data is too high.

It is necessary to that a user can choose different roles for different data, e.g., be a data sensor for one data while a data requester or stranger for another data.

5. DATA QUALITY

A data can be collected by different *qualities* denotes a photo can be captured by different locations.

In common, a user needs to consume more resources for sensing a data with a better quality. A user can prefer a data with a better quality than with a lower quality. Similarly, different users may have different personal preferences for the same data with the same quality.

- Consider N set of mobile users and they have I set of data.
- When a mobile user needs data it can send request to the server S.
- The server processes the request and forwards it to all active mobile users.
- After receiving the request the sender can establish connection between all receivers in a peer to peer manner.
- Finally the transmission ends the server allocates a incentive based on the quality of data.

6. DATA PROTECTION

We focus on security to avoid unauthorized user to access sensing data. In this project we have interface for user registration. User can register their details such as email id, name, and location through user registration. After registration user can login and send resource request to the server. The Server forwards the request to the data owner. The data owner can directly send the requested data to the receiver. After the data transmission, the sender can collect incentive from server. Hence the sensing data is transferred to the authorized user securely.

6.1 AES ALGORITHM

AES is based on a design principle known as a substitution–permutation network, a combination of both substitution and permutation, and is fast in both software and hardware. AES is a variant of Rijndael which has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. By contrast, the Rijndael specification per se is specified with block and key sizes that may be any multiple of 32 bits, with a minimum of 128 and a maximum of 256 bits.

AES operates on a 4 × 4 column-major order matrix of bytes, termed the state, although some versions of Rijndael have a larger block size and have additional columns in the state. Most AES calculations are done in a particular finite field.

For instance, if there are 16 bytes, b_0, b_1... b_{15}, these bytes are represented as matrix. The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the cipher text. The number of cycles of repetition is follows:

- 10 cycles of repetition for 128-bit keys.
- 12 cycles of repetition for 192-bit keys.
- 14 cycles of repetition for 256-bit keys.

Each round consists of several processing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform cipher text back into the original plaintext using the same encryption key.

Step 1: Derive the set of round keys from the cipher key.

Step 2: Initialize the state array with the block data (plaintext).

Step 3: Add the initial round key to the starting state array.

Step 4: Perform nine rounds of state manipulation.

Step 5: Perform the tenth and final round of state manipulation.

Step 6: Copy the final state array out as the encrypted data (cipher text).

6.2 PSEUDOCODE

```
Cipher (byte in [16], byte out [16], key_arrayround_key [Nr+1])
begin
byte state [16];
state = in;
AddRoundKey (state, round_key[0]);
For i = 1 to Nr-1 stepsize 1 do SubBytes (state);
ShiftRows (state);
MixColumns (state);
AddRoundKey (state, round_key[i]);
end for
SubBytes (state);
ShiftRows (state);
AddRoundKey (state, round_key [Nr]);
End
```

7. DATA RECOVERY

Data recovery is a process of retrieving inaccessible, lost, corrupted, damaged or formatted data from secondary storage, removable media or files, when the data stored in them cannot be accessed in a normal way. In this project we introduce Backup cloud to retrieve data. Data may be loss during transmission so before sending data, sender take backup of the original data so when sender knows his data fails during transmission then sender retrieve data from Backup cloud and send data to the receiver again. Before requesting data, receiver prefer amount for that data. After getting data from sender, balance of receiver is deducted and added it to the sender bank balance.

7.1. REMOTE BACKUP ALGORITHM

For taking backup store the data on the cloud server and the remote server simultaneously. On loss of the main file, the data is recovered from the remote server. For the backup and recovery process to be secure, user authentication based on attributes, secret keys is done so that the transaction is safe and authenticated.

- In this work during the transaction, the server S collects the check sum of transmitted data twice.
- Then take a copy of data to remote cloud RC.
- Server S compares the check sum that generated before the transmission and after the transmission.
- If both are equal the transmission is success.

- Else it will roll back the data to receiver.

8. CONCLUSION

In this work, we study that the architecture, which can quality-aware P2P-based Mobile Crowd Sensing effectively reduce the management and operational cost on the server and also user behavior dynamics. In this project there is an interface for user registration. User who wants information or data can register their details through user registration form. After the user registration user can login using their email id and password then the user send resource request to the server. Server forwards the resource request to the data owner. The data owner can directly send the requested data to its clients. If any failure occurs during the data transmission, we can retrieve it from our backup cloud. By using these Backup we prevent from loss of data and save cost. After the data transmission, the sender can collect incentive from server.

REFERENCES

[1] C. Jiang, L. Gao, L. Duan, and J. Huang, "Economics of peer-to-peer mobile crowd sensing," in Proc. IEEE GLOBECOM, December 2015.
[2] T. Luo and C.-K. Tham, "Fairness and social welfare in incentivizing participatory sensing," in Proc. IEEE SECON, June 2012.
[3] X. Zhang, et al., "Free market of crowd sourcing: incentive mechanism design for mobile sensing," IEEE Trans. Parallel and Distrib.Syst., vol. 25, no. 12, pp. 3190-3200, December 2014.
[4] J. Niwa, et al., "MPSDataStore: a sensor data repository system for mobile participatory sensing," in Proc. ACM MCC Workshop, 2013.
[5] S. Kosta, et al., "Clone2Clone (C2C): peer-to-peer networking of smart phones on the cloud," in Proc. USENIX HotCloud'13, 2013.
[6] O. Babaoglu, M. Marzolla, and M. Tamburini, "Design and implementation of a P2P Cloud system," in Proc. ACM SAC12, 2012.
[7] Y. Kaneko, et al., "A location-based peer-to-peer network for context-aware services in a ubiquitous environment," in Proc. IEEE Symp. Appl. and Internet Workshops, January 2005.
[8] R. Kravets, et al., "Crowd Watch: enabling in-network crowd sourcing," in Proc. ACM MCC Workshop, August 2013.
[9] M. Keally, G. Zhou, G. Xing, and J.Wu, "Remora: sensing resource sharing among smart phone-based body sensor networks," in Proc.IEEE/ACM IWQoS, June 2013.
[10] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," IEEE Commun. Mag., vol. 49, no. 11, pp. 32- 39, November 2011.