Volume 6, Issue 1, January 2016 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Privacy-Aware Data Aggregation in Mobile Sensing with Fault Tolerance

Swapnil Mohan Ranjankar, Ashwini Jadhav

ME Computer Networks, Savitribai Phule Pune University, Nutan Maharashtra Institute of Engineering and Technology College, Pune, Maharashtra, India

Abstract— Over the years capability of the mobile sensing devices like smart phones are increased in terms of capturing and sharing the information. This information can be useful if analysed as an aggregated value or values. Classic examples are like traffic trend in particular area, medical information in particular locality etc. When such information is shared to an aggregator, user's identity should be compromised and only information needs to be shared. To achieve this user's identity Li, Cao and Porta [1] has suggested efficient protocol to obtain sum aggregate value using homomorphic encryption and novel key management. Unfortunately, the use of cryptography in the projected protocol scheme introduced the all-or-nothing decryption model. Therefore, the aggregator learns nothing if a single user fails. In some of the critical applications, data has to be collected from all users to get right results. Any leakage in data will lead to incorrect aggression and sub-sequent calculations. Fault tolerance has to be handled to get right results. T.-H.H. Chan, E. Shi, and D. Song [4] have proposed a technique to handle fault tolerance. In proposed technique, a binary interval tree over n users, and allow the aggregator to estimate the sum of contiguous intervals of users as represented by nodes in the interval tree. The binary-tree technique allows handling user failures joins and leaves, with a small logarithmic (or polylog) cost in terms of communication and estimation error.

Keywords— Secure computing, Mobile Sensing, Privacy, Data Aggregation, Fault Tolerance

I. INTRODUCTION

1.1 Motivation

A WSN is a large network made of numerous number of sensor nodes with sensing, computation, and wireless communications capabilities. The wireless sensor network consists of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc. and to cooperatively pass their data through the network to a main location. Wireless Sensor Networks are used in variety of fields which includes military, healthcare, environmental, biological, home and other commercial applications. With the huge advancement in the field of embedded computer and sensor technology, Wireless Sensor Networks, which is composed of several thousands of sensor nodes which are capable of sensing, actuating, and relaying the collected information, have made remarkable impact? A mobile sensing is a system in which a mechanism whose computing, sensing, and communication capabilities enable the realization of different applications and services. Mobile sensing is considered to be one of the most popular techniques in wireless sensor networks. Mobile sensing in wireless sensor network is increasingly becoming a part of everyday's life due to rapid evolution of the mobile phone into a powerful sensing platform. Some of the popular consumer smartphones are now equipped with the necessary sensors to monitor a diverse range of human activities and commonly encountered contexts. Mobile sensing can simply be defined as a wireless sensor network in which the sensor nodes are mobile. However, many of their applications are similar, such as environment monitoring or surveillance. Commonly the nodes consist of a radio transceiver and a microcontroller powered by a battery as well as some kind of sensor for detecting light, heat, humidity, temperature, etc [9], [10] and [11].

A mobile device such as smart phones is gaining an ever-increasing popularity. Most of the smart phones are equipped with a rich set of embedded sensors such as camera, microphone, GPS, accelerometer, ambient light sensor, gyroscope, and so on. The environmental monitoring, healthcare are monitored and sensed using sensors.

Considering this popularity and variety of usefulness of mobile sensing, various applications are built based on the aggregating the collected data. User shares data and collected by a central application called aggregator who performs aggregation and provides the final result [7]. This activity happens on certain time interval. In such continuous data collection process, user's identity is crucial thing. User's identity should not be compromised and should be intact. Work done so far in this field is considering the "Trusted Aggregator". This assumption will not be always true. And there is possibility that user's identity can be compromised. Li, Cao and Porta [1] had proposed solution for this thing considering aggregator will be untrusted. In their solution -- an efficient technique is proposed for user's privacy which is query processing and can also check the aggregate statistics. The query is processed for the user inputs as the data from user's are privacy sensitive. The aggregation accumulates all the data and sends the user only the final computed value. The aggregate values of users are obtained by computing the inputs, the minimum and sum values are obtained. These values get updated for each input given by the user. All these values are viewed in statistical form through smart phones. This technique is based on homomorphic encryption and a novel key management technique.

In this proposed technique, user's identity and privacy is maintained but fault tolerance is not rightly supported. The use of cryptography in the projected protocol scheme introduced the all-or-nothing decryption model. Therefore, the aggregator learns nothing if a single user fails. In some of the critical applications, data has to be collected from all users to get right results. Any leakage in data will lead to incorrect aggression and sub-sequent calculations.

T.-H.H. Chan, E. Shi, and D. Song [4] have proposed a technique to handle fault tolerance. In proposed technique, a binary interval tree over n users, and allow the aggregator to estimate the sum of contiguous intervals of users as represented by nodes in the interval tree.

1.2 Goal

The aim is to protect privacy for each user in mobile sensing by implementing encryption and decryption methods when the aggregator is untrusted with the feature of fault tolerance to get right results in case of business critical applications.

II. LITERATURE SURVEY

Literature survey is the way of finding previous work done on the topic selected for project. This survey helps in finding and acknowledging the progress previously done on the paper. This also enables researchers to define scope of project and modifications needed to existing system so that proposed system can give advantages to the end user.

2.1 Related Work

Initial study for privacy aware data aggregation is done based on the assumption of "Trusted Aggregator". Considering the development done so far in this field, this assumption is not always true as the aggregator is not trustworthy [2], [3], [4], and [6]. Aggregator may be interested in the identity of the communicating entity. This fact motivates various researches in this field.

A very first algorithm was proposed by Rastogi and Nath [2]. They proposed the first differentially private aggregation algorithm for distributed time-series data that offered good practical utility without any trusted server. To ensure differential privacy for time-series data despite the presence of temporal correlation, they proposed the Fourier Perturbation Algorithm. Standard differential privacy techniques were not performing well with time series data. These techniques add noise in answer. And when inputs were more automatically noise is more and answer were practically useless. With Fourier Perturbation algorithm, noise was getting reduced even for the large data.

To deal with untrusted aggregator problem, Rastogi and Nath [2] proposed another algorithm -- Distributed Laplace Perturbation Algorithm. In this algorithm they add noise in distributed way in order to guarantee differential privacy. DLPA was the first distributed differentially private algorithm that can scale with a large number of users: DLPA outperformed the only other distributed solution for differential privacy proposed so far, by reducing the computational load per user.

Below diagram provides basic model how data aggregation is preformed –

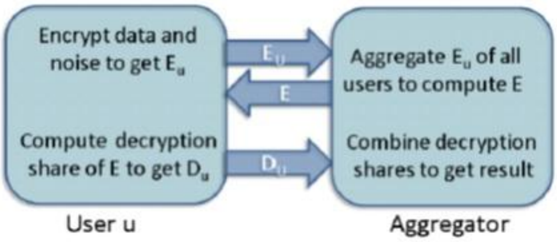

Encrypt data and noise to get E_u
E_u
E
Compute decryption share of E to get D_u
D_u
Aggregate E_u of all users to compute E
Combine decryption shares to get result
User u
Aggregator

Figure 1: Basic model of Rastogi and Nath [2] proposed algorithm

This proposed solution is based on threshold Paillier cryptosystem [5]. In a key generation phase, the private key is getting generated and distributed among the users Thus the users all together could perform distributed decryption using their keys. The protocol gets executed in two phases. In the first phase, the aggregator computes the required Q in encrypted form. Then in the second phase, a distributed decryption protocol is run to recover Q from the encrypted form.

Rastogi and Nath [2] algorithm is based on threshold Paillier cryptosystem[5] in which to decrypt the sum, their scheme needs an extra round of interaction between the aggregator and all users in every aggregation period, which means high communication cost and long delay. Moreover, it requires all users to be online until decryption is completed, which may not be practical in many mobile sensing scenarios due to user mobility and the heterogeneity of user connectivity.

To overcome extra round of interaction, E.Shi, T-H.H.Chan, E.Rieffle, R.Chow and D.Song [3] has proposed another protocol based on unidirectional communication which considers untrusted aggregator. They achieved strong privacy guarantees using two main techniques. First, to utilize applied cryptographic techniques to allow the aggregator to decrypt the sum from multiple cipher texts encrypted under different user keys. Second, using a distributed data randomization procedure, that guarantees the differential privacy of the outcome statistic, even when a subset of participants might be compromised.

E.Shi, T-H.H.Chan, E.Rieffle, R.Chow and D.Song [3] proposed algorithm based on novel Private Stream Aggregation (PSA) which allow users to upload a stream of encrypted data to an untrusted aggregator, and allow the aggregator to decrypt (approximate) aggregate statistics for each time interval with an appropriate capability. Privacy is achieved through two ways - first, aggregation scheme is aggregator oblivious, meaning that the aggregator is unable to learn any unintended information other than what it can deduct from its auxiliary knowledge and the desired statistics. Second, distributed differential privacy for each individual participant, in the sense that the statistic revealed to the aggregator will not be swayed too much by whether or not a specific individual participates.

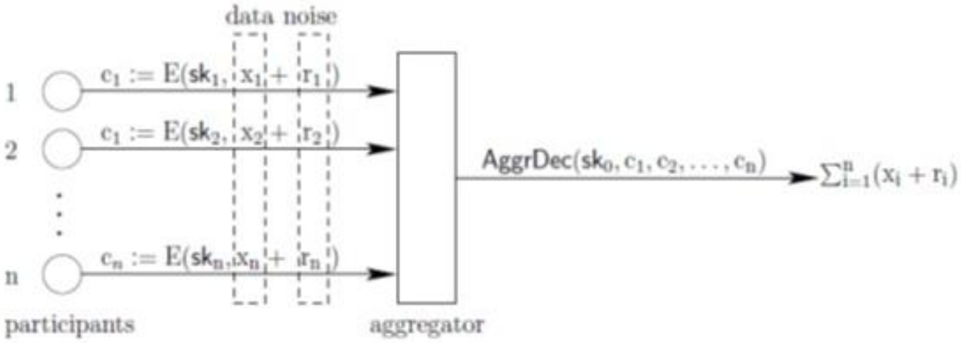

data noise
1
2
n
$c_1 := E(sk_1, x_1 + r_1)$
$c_1 := E(sk_2, x_2 + r_2)$
$c_n := E(sk_n, x_n + r_n)$
$AggrDec(sk_0, c_1, c_2, \ldots, c_n)$
$\sum_{i=1}^{n}(x_i + r_i)$
participants
aggregator

Figure 2: Overview of the proposed solution by E.Shi, T-H.H.Chan, E.Rieffle, R.Chow and D.Song [3]

Figure gives overview of the proposed solution by E.Shi, T-H.H.Chan, E.Rieffle, R.Chow and D.Song [3]. In every time period, each participant adds noise r_i to her value x_i before encrypting it. The aggregator uses the capability sk_0 to decrypt a noisy sum, but learns nothing more. The noisy sum output by this distributed mechanism ensures each participant's differential privacy.

Their research further studied to deal with fault tolerance by T.-H.H. Chan, E. Shi, and D. Song [4]. Concepts related to privacy control are same with added feature of fault tolerance. They achieved failure tolerance by building a binary interval tree over n users, and allow the aggregator to estimate the sum of contiguous intervals of users as represented by nodes in the interval tree. This is helped to handle user failures, joins and leaves, with a small logarithmic (or polylog) cost in terms of communication and estimation error.

When we studied both these protocols in detail, we found that, it has high computation and storage cost to deal with collusions in a large system. One limitation of their cryptographic construction is that it supports only polynomial-sized plaintext spaces for computing sums.

To overcome extra round of interaction and to support large text, Q.Li, G. Cao, T.Porta [1] proposed efficient protocol. Their protocol is designed to obtain sum aggregation of time series data in case of untrusted aggregator which can be extended to min and other aggregation. Protocol employs an additive homomorphic encryption and novel key management scheme based on efficient HMAC. Each user needs to calculate very small number of HMAC [Hash Message Authentication Code] to encrypt data.

Key benefits of this protocol are

- Low cost for computation
- Scale to large system
- Support for plain large text
- Single round of user to aggregator communication
- Extended to count, avg, max, min aggregation with some extra derivation

Below diagram provides how protocol model works –

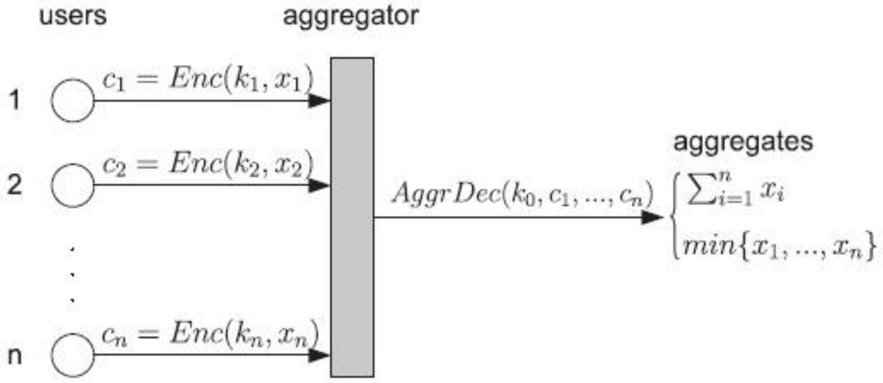


Figure 3: Protocol proposed by Q.Li, G. Cao, T.Porta [1]

This proposed protocol also rightly handles the dynamic joins and leaves without affecting the system. But proposed protocol won't handle fault tolerance rightly. There is no full proof mechanism to handle fault tolerance as it is important for some business critical applications. We can implement binary protocol for handling fault tolerance proposed by T.-H.H. Chan, E. Shi, and D. Song [4]. This will make protocol proposed by Q.Li, G. Cao, T.Porta [1] full proof to handle fault tolerance.

III. PROBLEM STATEMENT

There is a need to collect data with various advanced sensors some of them are fitted into latest smart phones. Data collected by individual sensor won't be always helpful. Data has to be collected and aggregated to get right results. To perform this task, basic assumption is "Trusted Aggregator" which is not true. There is a need of protocol to provide data aggregation without compromising privacy with "Untrusted aggregator" and with an enhancement to handle fault tolerance.

IV. PROPOSED SOLUTION

We propose a new protocol for mobile sensing to obtain the sum aggregate of time-series data in the presence of an untreated aggregator with fault tolerance.

We propose an efficient protocol to obtain the Sum aggregate, which employs an additive homomorphism encryption and a novel key management technique to support large plaintext space and fault tolerance by creating binary tree.

We propose a scheme that utilizes the redundancy in security to reduce the communication cost for each join and leave.

We also propose a scheme that employs the redundancy in security to reduce the communication cost of dealing with dynamic joins and leaves.

One building block of our solution is the additive homomorphism encryption scheme proposed by Castelluccia [8].

Advantages:

- It reduces the Communication cost of dealing with dynamic joins and leaves
- Users may frequently join and leave in mobile sensing
- In each time period, a mobile user sends her encrypted data to the aggregator via Wi-fi, 3G or other available access networks
- No peer-to-peer communication is required among mobile users, since such communication is nontrivial in mobile sensing scenarios due to the high mobility of users and users may not be aware of each other for privacy reasons

4.1 High Level Architecture

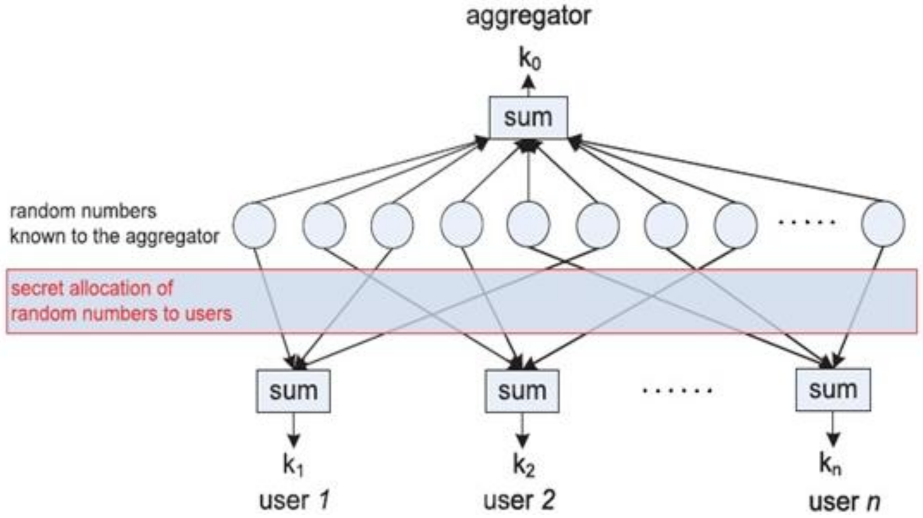

aggregator
k0
sum
random numbers known to the aggregator
secret allocation of random numbers to users
sum
sum
sum
k1
user 1
k2
user 2
kn
user n

Figure 4: High level architecture for proposed system

The intuition behind the straw man construction. The aggregator computes the sum of a set of random numbers as the decryption key. These numbers are secretly allocated to the users, and each user computes the sum of its allocated numbers as the encryption key. The aggregator does not know which random numbers are allocated to each user, and thus does not know any user's key.

4.2 Encryption & Decryption Methodology

One building block of our solution is the additive homomorphic encryption scheme proposed by Castelluccia[8]. This scheme works as follows:

Encryption:

1. Represent message m as an integer within range [0, M-1], where M is a large integer.
2. Let k be a randomly generated key, k ϵ {0, 1} λ, where λ is a security parameter.
3. Output cipher c = (m + h(fk(r))) mod M, where fk is a pseudorandom function (PRF) that uses k as a parameter, h is a length-matching hash function and r is a nonce for this message.

Decryption:

Output plaintext m = (c - h(f_k(r)))mod M

4.3 Protocol Implementation

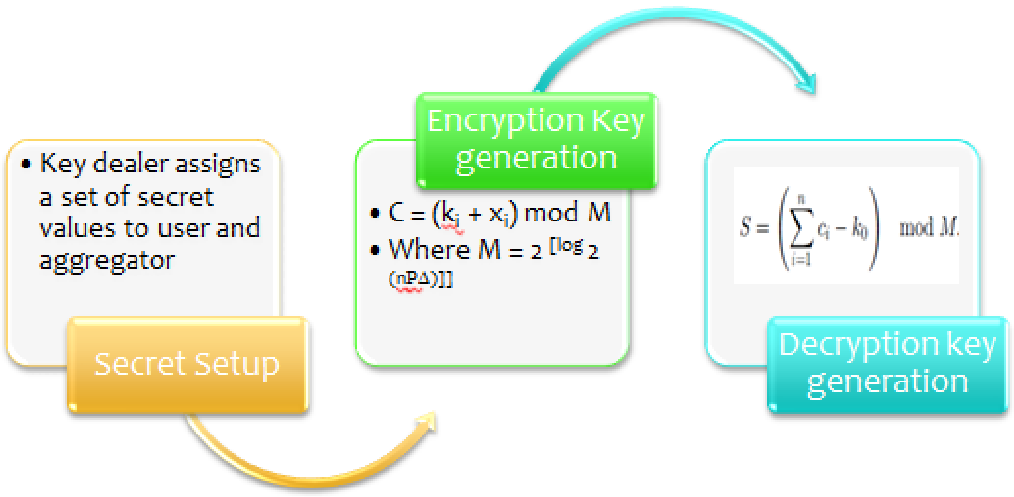

Key dealer assigns a set of secret values to user and aggregator
Secret Setup
Encryption Key generation
C = (ki + xi) mod M
Where M = 2 [log 2 (nPΔ)]]
S = (∑ci − k0) mod M.
Decryption key generation

Figure 5: Protocol Implementation

Above diagram depicts the protocol implementation for projected solution which consists of three main steps –

Secret setup -- The key dealer assigns a set of secret values (secrets for short) to each user and the aggregator. Any standard key distribution algorithm can be used for this. A unique key will be shared to all users which will be used to encrypt the date. This encrypted data will be shared to aggregator.

Encryption key generation -- In each time period, user i (i ϵ [1, n]) generates encryption key k_i using the secrets that it is assigned.

It encrypts its data x_i by computing

$c_i = (k_i + x_i) \bmod M$

where $M = 2^{[\log_2 (n\Delta)]}$. Then, it sends the ciphertext c_i to the aggregator.

Decryption key generation -- In each time period, the aggregator generates decryption key k_0 using the secrets that it is assigned, and decrypts the sum aggregate $S = \Sigma x_i$ by computing

$S = (\Sigma c_{i\,-}\ k_0)$

The keys are generated using a PRF family and a length matching hash function.

V. DEALING WITH DYNAMIC JOINS & LEAVES

In mobile sensing applications, users may join and leave. When a user joins, it should be assigned some secrets for encryption key generation. When a user leaves, its secrets should be reclaimed such that the aggregator can still get the aggregate statistics of the remaining users. Dynamic joins and leaves should be properly dealt with to protect each user's privacy and ensure the secrecy of the aggregate statistics.

When the number of users is not large and the churn rate is low, the key dealer can rerun the secret setup phase for all the users whenever a user joins or leaves. However, for the applications with a large number of users and/or a high churn rate, the communication overhead may be too high to redistribute secrets to all users. In this section, we propose efficient techniques to deal with dynamic joins and leaves for a large-scale system. Basically, we use redundancy in security to reduce the communication overhead of joins and leaves.

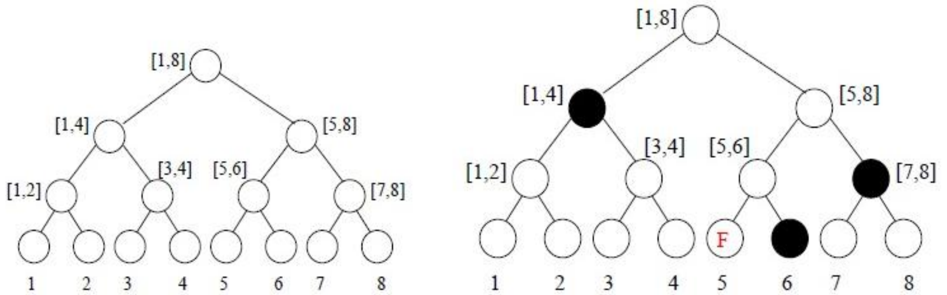
For simplicity, we evaluate the communication overhead of dealing with a user's join and leave by the number of users that the key dealer should redistribute secrets to (or the number of updated users for short). Since the number of secrets redistributed to each user is not large, if we assume that these secrets can be included in one message, the number of updated users is equivalent to the number of messages that should be transmitted from the key dealer to the users.

For simplicity, we only consider the Sum protocol when describing our scheme to deal with dynamic joins and leaves, but the scheme applies to the protocol for Min as well.

Here we need to consider a strong adversary who can monitor the communications between all entities including the key dealer and users. Through eavesdropping the message sending and receiving activities, the adversary can know which user joins or leaves and to which users secrets are redistributed to.

VI. FAULT TOLERANCE USING BINARY PROTOCOL

Efficient protocol for privay suggested by Li, Cao and Porta [1] is not having fault tolerance feature. Fault tolerance is projected implementation as a part of this paper. T.-H.H. Chan, E. Shi, and D. Song [4] have proposed a technique to handle fault tolerance. In proposed technique, a binary interval tree over n users, and allow the aggregator to estimate the sum of contiguous intervals of users as represented by nodes in the interval tree. The binary-tree technique allows handling user failures joins and leaves, with a small logarithmic (or polylog) cost in terms of communication and estimation error.



(a) The aggregator obtains block estimates corresponding to all nodes appearing in the binary interval tree.

(b) When user 5 fails, the aggregator sums up the block estimates corresponding to the black nodes.

Figure 6: Binary Protocol for Fault Tolerance

As depicted in above figure, construction is based on a binary interval tree, hence the name Binary Protocol. For ease of exposition, assume for now that n is a power of 2. Each leaf node is tagged with a number in [n]. Each internal node in the tree represents a contiguous interval covering all leaf nodes in its subtree. As a special case, we can think of the leaf nodes as representing intervals of size 1. For each node in the tree, we also use the term block to refer to the contiguous interval represented by the node.

Intuitively, the aggregator and users would simultaneously perform the BA Scheme for every interval (or block) appearing in the binary tree. Hence, the aggregator would obtain an estimated sum for each of these blocks. Normally, when n is a power of 2, the aggregator could simply output the block estimate for the entire range [1, n]. However, imagine if a user i fails to respond, the aggregator would then fail to obtain block estimates for any block containing i, including the block estimate for the entire range [1, n].

Fortunately, observe that any contiguous interval within [n] can be covered by O(log n) nodes in the binary interval tree. If users have failed, the numbers 1 through n would be divided into k + 1 contiguous intervals, each of which can be covered by O(log n) nodes. This means that the aggregator can estimate the sum of the remaining users by summing up O((k + 1) log n) block estimates.

VII. CONCLUSION

To facilitate the collection of useful aggregate statistics in mobile sensing without leaking mobile users' privacy, we proposed a new privacy-preserving protocol to obtain the Sum aggregate of time-series data. The protocol utilizes additive homomorphic encryption and a novel, HMAC based key management technique to perform extremely efficient aggregation. Implementation-based measurements show that operations at user and aggregator in our protocol are orders of magnitude faster than existing work. Thus, this protocol can be applied to a wide range of mobile sensing systems with various scales, plaintext spaces, aggregation loads, and resource constraints.

Based on the Sum aggregation protocol, we also proposed two schemes to derive the Min aggregate of time-series data. One scheme can obtain the accurate Min, while the other one can obtain an approximate Min with provable error guarantee at much lower cost.

To deal with dynamic joins and leaves, we proposed a scheme that utilizes the redundancy in security to reduce the communication cost for each join and leave.

Using binary tree various blocks are formed. One idea is to form user groups, and run the Block Aggregation Scheme for each block. The aggregator is then able to estimate the sum for each block. If a subset of the users fails, we must be able to find a set of disjoint blocks to cover the functioning users. In this way, the aggregator can estimate the sum of the functioning users.

REFERENCES

[1] Qinghua Li, Guohong Cao, Thomas F. La Porta, "Efficient and Privacy-Aware Data Aggregation in Mobile Sensing", IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 11, NO. 2, MARCH/APRIL 2014

[2] Rastogi and S. Nath, "Differentially Private Aggregation of Distributed Time-Series with Transformation and Encryption," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2010.

[3] E. Shi, T.-H.H. Chan, E. Rieffel, R. Chow, and D. Song, "Privacy- Preserving Aggregation of Time-Series Data," Proc. Network and Distributed System Security Symp. (NDSS '11), 2011

[4] T.-H.H. Chan, E. Shi, and D. Song, "Privacy-Preserving Stream Aggregation with Fault Tolerance," Proc. Sixth Int'l Conf. Financial Cryptography and Data Security (FC '12), 2012.

[5] P.A. Fouque, G. Poupard, and J. Stern, "Sharing Decryption in the Context of Voting or Lotteries," Proc. Fourth Int'l Conf. Financial Cryptography (FC '00), pp. 90-104, 2000.

[6] E.G. Rieffel, J. Biehl, W. van Melle, and A.J. Lee, “Secured Histories: Computing Group Statistics on Encrypted Data While Preserving Individual Privacy,” http://arxiv.org/abs/1012.2152, 2010.

[7] Q. Li and G. Cao, “Mitigating Routing Misbehavior in Disruption Tolerant Networks,” IEEE Trans. Information Forensics and Security, vol. 7, no. 2, pp. 664-675, Apr. 2012.

[8] G. Acs and C. Castelluccia, “I Have a Dream!: Differentially Private Smart Metering,” Proc. 13th Int’l Conf. Information Hiding (IH ’11), pp. 118-132, 2011.

[9] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M.Hansen, E. Howard, R. West, and P. Boda, “Peir, the Personal Environmental Impact Report, As a Platform for Participatory Sensing Systems Research,” Proc. ACM/USENIX Int’l Conf. Mobile Systems, Applications, and Services (MobiSys ’09), pp. 55-68, 2009.

[10] J. A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H.Balakrishnan, S. Toledo, and J. Eriksson, “VTrack: Accurate, Energy-Aware Road Traffic Delay Estimation Using Mobile Phones,” Proc. ACM Seventh Conf. Embedded Networked Sensor Systems (SenSys ’09), pp. 85-98, 2009.

[11] S. Consolvo, D.W. McDonald, T. Toscos, M.Y. Chen, J. Froehlich,B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby, I. Smith, and J.A. Landay, “Activity Sensing in the Wild: A Field Trial of Ubifit Garden,” Proc. SIGCHI Conf. Human Factors in Computing Systems (CHI ’08), pp. 1797-1806, 2008..

ABOUT AUTHOR

Mr. Swapnil Mohan Ranjankar completed B.E. Information Technology from Pune University, India in 2004 and currently pursuing M.E. Computer Networks from Savitribai Phule Pune University. His areas of interest are *Mobile Sensing, Privacy and Data Aggregation.*

Prof. Ashvini Jadhav completed M.E.Computer Networks from Pune University, India in 2013.
I am currently working in Nutan Maharashtra Institute of Engineering and Technology College as Assistant Professor.