Privacy-Preserving Data Aggregation in Wireless Sensor Networks

Kushal Gupta 1, Ashutosh Kumar 2, Saurabh Gupta 3
Department of Computer Science and Engineering
1,2MNNIT Allahabad, Allahabad, India
3Al-Falah School of Engg. & Tech., New Delhi, India
1kushal.loveiit@gmail.com, 2ashutosh.k.sonkar@gmail.com, 3en.saurabhgupta@gmail.com

Abstract— It is very challenging problem in Wireless Sensor Networks to provide privacy preservation during Data aggregation. In this we are describing two schemes that provide different approaches for privacy preservation. In the first one there are two approaches for additive aggregation functions, which can be extended to approximate MAX/MIN aggregation functions. The first scheme: Cluster-based Private Data Aggregation (CPDA) uses clustering protocols and algebraic properties of polynomials. It has the advantage of less communication overhead. The second scheme: Slice-Mix-AggRegaTe (SMART) uses slicing techniques and the associative property of addition. It has the advantage of less computation overhead. The motive of this paper is to minimize the complexity between collaborative data collection by wireless sensor network and data privacy. In the second one the goal is to find a new effective scheme that minimizes the computational overhead in CPDA. In this paper we proposed a modified scheme for privacy preserving data aggregation which was inspired by CPDA and consume less resource than CPDA and made a comparison with CPDA in computational overhead and communication overhead. Here we present simulation results of the schemes and compare their performance to a typical data aggregation scheme - TAG, where no data privacy protection is provided.

Keywords – Wireless sensor Networks, Data Aggregation, Clustering, TAG.

I. INTRODUCTION

As we know a wireless sensor network is a network consisting of small sensors that senses the different attributes in their surroundings. Work of sensor is to collect some information from the surroundings and pass it to its neighbor and so to the destination. As sensors are using wireless medium to send the information the major problem is of power consumption. If the frequency of sending data is kept high then the power consumption will be very high due to continuous sending of data. A sensor sending a bit of information continuously over sensor sending a bit of information after some manipulations over some period of time is quite tedious and power consuming. It suggests a network processing that is used to minimize the raw data sent by the sensor nodes. This approach is named as Data Aggregation. In many applications we may only concerned with some aggregate results of the readings over a period of time, that may be sum, average or max/min of data.

Now-a-days sensor network applications are increasing day by day so privacy of data sent by the sensors is one of the major problems imposed by wireless sensor network. For example let’s take an example of an application that measure the household details such as how much water and electricity is used in a particular home. This information may be used for govt. agencies for resource planning of supply. But there is a counter-side of the application that it reveals the daily activities of the particular home as when all the family members are out of house. So we must have to use such a scheme that collects the aggregated sensor readings while at the same time preserve data privacy.

Here in both of these a scheme named as Homomorphic encryption scheme [3] is used to aggregate the data. This scheme is used for additive aggregation functions. Using this scheme we can aggregate encrypted data without any need of decryption in the intermediate nodes. In the approach, there are two privacy-preserving data aggregation schemes called Cluster-based Private Data Aggregation (CPDA) and Slice-Mix-AggRegaTe (SMART) respectively. These both schemes are for additive aggregation functions in wireless sensor networks. The motive of this paper is to minimize the complexity between collaborative data collection by wireless sensor network and data privacy. When there is no packet loss during data transmission, in both schemes CPDA and SMART we get the accurate results and it guarantees that no individual sensor reading is disclosed to other sensors.
The paper evaluates the schemes on the basis of computation and communication overheads and proposed a new scheme which was inspired by CPDA. The modified scheme has lower computation overhead than CPDA and it is more secure. Furthermore there is a comparison with CPDA in computation overhead and communication overhead.

II. DESCRIPTION OF FIRST SCHEME

In this our focus is to build some approach that can help in securing the data sent within a wireless sensor network. For this homomorphic Scheme is used to design two privacy preserving data aggregation schemes for additive aggregation functions: CPDA and SMART. In both the schemes the information from a sensor node is known only to that sensor node only while others will get the aggregated values of the readings.

A. Model and Background

1) Sensor Networks and the Data Aggregation Model: In this paper, a sensor network is modeled as a connected graph \( G(V, E) \), where sensor nodes are represented as the set of vertices \( V \) and wireless links as the set of edges \( E \). Then the number of sensor nodes is defined as \( |V| = N \). A data aggregation function is defined as:

\[
y(t) = f(d_1(t), d_2(t), ..., d_N(t))
\]

Where \( d_i(t) \) is the individual sensor reading at time \( t \) for node \( i \). Generally functions of \( f \) include sum, average, min, max and count.

In this article, we focus on additive aggregation functions, that is,

\[
f(t) = \sum_{i=1}^{N} d_i(t).
\]

As we know that additive aggregation is not too restrictive and moreover many other aggregation functions such as average and count can also be reduced to the additive aggregation function sum. Furthermore, some functions such as min and max, can also be approximated through additive functions. According to

\[
\max(x_1, \ldots, x_N) = \lim_{n \to \infty} (x_1^k + \cdots + x_N^k)^{1/k}
\]

and

\[
\min(x_1, \ldots, x_N) = \lim_{n \to \infty} (x_1^k + \cdots + x_N^k)^{1/k}
\]

Now we can find out the max and min values by assigning \( k \) a large value.

2) Key Set-up for Encryption: Here we briefly review a random key distribution mechanism proposed in [5]. Here key distribution consists of three phases:

1. Key Pre-distribution
2. Shared-key Discovery and
3. Path-key Establishment

In the beginning, a key pool of \( K \) (Sufficiently large) keys and their corresponding identities are generated. Each sensor within the sensor network is given \( k \) keys that are randomly taken from the key pool. These \( k \) keys are called as the keying for the particular sensor node. During the key discovery phase, every sensor node searches for a common key that it is holding with its neighbors by exchanging discovery messages. If two neighbors have the same key then there is a secure link between two nodes. In the path-key establishment phase, a key (path key) is given to the pairs of neighboring sensor nodes that are not having common key but can be attached by two or more multi-hop secure links.

B. Private Data Aggregation Protocols

Here we will study about two private data aggregation protocols that emphasize on additive data aggregation methods. First uses the concept of clustering and so named as cluster-based private data aggregation (CPDA). In CPDA there are 3 phases named as cluster formation, calculation of aggregate results and data aggregation. The next scheme is based on the simple feature of breaking the whole data into pieces and so named as slice-mix-aggregate (SMART). In this method each node slice (breaks into pieces) its data and send encrypted slices to different
aggregators to hide its private data. Finally the aggregators collect all data slices and forward it to the query server. After receiving aggregated data, query server calculates the final aggregation result.

1) **Cluster-Based Private Data Aggregation (CPDA):**

a) **Formation of Clusters [3]:** As the name suggest firstly it construct clusters to perform intermediate aggregation. The cluster formation procedure is illustrated in figure 1. In the very beginning a query server (let’s say Q) broadcasts a message say “hello” message. When a sensor receives the hello message it elects itself a cluster leader on the basis of a predefined probability pc. If a node becomes cluster head in a cluster then it again forward the hello message to its neighbors, the node waits for certain time duration for receiving the hello message otherwise and joins the cluster by sending JOIN message to the cluster head. This procedure repeats again and again until all the nodes joins some cluster.

![Fig. 1: Formation of Clusters](image)

b) **Calculation within Clusters:** The next step in CPDA is to aggregate the intermediate values among the cluster members. For simplification let a cluster contains three members: A, B and C and a, b, c are the private data held by A, B and C. Let

A-Cluster Leader
B, C-Cluster Members

Figure 2 illustrates the message exchange among the three nodes to obtain the desired sum without releasing individual private data.
All cluster members share a common knowledge of nonzero numbers, referred to as seeds, \(x, y\) and \(z\), which are distinct from each other (as shown in Figure 2(1)). Now A will calculate:

\[
\begin{align*}
    v_A^A &= a + r_1^A x + r_2^A x^2 \\
    v_B^B &= a + r_1^B y + r_2^B y^2 \\
    v_C^C &= a + r_1^C z + r_2^C z^2 
\end{align*}
\]

Where \(r_1^A\) & \(r_2^A\) are random numbers generated by A and known only to node A.

Similarly node B and C calculate their values

\[
\begin{align*}
    v_A^B &= b + r_1^B x + r_2^B x^2 \\
    v_B^B &= b + r_1^B y + r_2^B y^2 \\
    v_C^C &= b + r_1^B z + r_2^B z^2 
\end{align*}
\]

and

\[
\begin{align*}
    v_A^C &= c + r_1^C x + r_2^C x^2 \\
    v_B^C &= c + r_1^C y + r_2^C y^2 \\
    v_C^C &= c + r_1^C z + r_2^C z^2 
\end{align*}
\]

Then node A encrypts \(v_A^A\) and sends to B using the shared key between A and B. It also encrypts \(v_A^B\) and sends to C using the sharing key between A and C (Figure 2(2)). Similarly node B encrypts and sends \(v_B^B\) to A and \(v_B^C\) to C; node C encrypts and sends \(v_C^C\) to A and \(v_C^B\) to B.

Node A receives \(v_A^B\) and \(v_A^C\), it has the knowledge of \(v_A^A = a + r_1^A x + r_2^A x^2\), \(v_B^B = b + r_1^B y + r_2^B y^2\) and \(v_C^C = c + r_1^C x + r_2^C x^2\).

Now node A calculates assembled value-

\[
F_A = v_A^A + v_B^A + v_C^A \\
F_A = (a + b + c) + r_3 x + r_2 x^2
\]

Where

\[
r_1 = r_1^A + r_1^B + r_1^C
\]
\[ r_2 = r_1^A + r_1^B + r_1^C \]

Now the cluster head A can calculate the final aggregated value \((a + b + c)\) as \(x, y, z, F_A, F_B, F_C\) are known to A.

c) **Cluster Data Aggregation:** There is a simple and well known technique for data aggregation is to build a routing tree. Each cluster head sends the final sum within the cluster back to the query server through a routing tree rooted at the server or we can say converge all the intermediate sums to the server.

2) **Slice-Mix-AggRegaTe (SMART):** As we have seen that there is a very typical computation in CPDA, its major drawback is computational overhead. To overcome this, another scheme came into picture is SMART, which will reduce computational overhead at the cost of slightly increased communicational bandwidth consumption. As the name suggests Slice-Mix-Aggregate is a three step scheme for preserving data privacy. Steps are:

- Each node hides its private data by slicing it into pieces
- Sends encrypted data slices to different intermediate aggregation nodes
- After the pieces are received, intermediate nodes calculate intermediate aggregate values and further aggregate to the Sink

a. **Step 1(Slicing):** All nodes \(i (i = 1, \ldots, N)\) within the network, randomly selects a set of nodes \(S_i (J = |S_i|)\) within \(h\) hops. Generally for a dense wireless sensor network, we can take \(h = 1\). Then node \(i\) slices (breaks into pieces) its private data \(d_i\) randomly into \(J\) pieces. Figure 3 shows the Slicing procedure.

\[ d_i = \sum_{j=1}^{I} d_{ij} \]

\[ d_{ij} = \text{set of slices} \]

For the node to which node \(i\) does not sending any slice-

\[ d_{ij} = 0 \]

![Fig. 3: Slicing](image)

b. **Step 2(Mixing):** On receiving encrypted slices node \(J\) decrypts the data using its shared key with the sender, then the sum of all the received slices is calculated.[Figure 4]

\[ r_j = \sum_{i=1}^{N} d_{ij} \]
Privacy-Preserving Data Aggregation in Wireless Sensor Networks

Where

\[ d_{ij} = 0, j \notin i \]

\[ r_1 = d_{11} \quad 1 \]
\[ r_2 = d_{12} + d_{22} + d_{32} + d_{42} + d_{52} + d_{62} \quad 2 \]
\[ r_3 = d_{33} + d_{43} + d_{73} \quad 3 \]
\[ r_4 = d_{24} + d_{44} \quad 4 \]
\[ r_5 = d_{15} + d_{55} + d_{65} \quad 5 \]
\[ r_6 = d_{26} + d_{56} + d_{66} + d_{76} \quad 6 \]
\[ r_7 = d_{37} + d_{47} + d_{67} + d_{77} \quad 7 \]

Fig. 4: Mixing

**c. Step 3 (Aggregation):** All nodes aggregate the data and send the result to the query server. [Figure 5]

III. DESCRIPTION OF SECOND SCHEME

As we have discussed the two approaches to preserve data privacy, this approach proposed a modified scheme for privacy preserving data aggregation which is inspired by CPDA. Using this scheme we can preserve the data privacy with consuming fewer resources than in CPDA

A. System Model
Consider a connected graph $G(V,E)$, where $V$ is the sensor nodes in the network and $E$ is the edge between vertices. $E_{i,j}$ denotes the edge between node $i$ and node $j$. The total number of nodes in the wireless sensor network is $N$. Here data aggregation function is

$$f(t) = f(m_1(t), m_2(t), ..., m_N(t))$$

Where $m(t)$ - sensor reading at time $t$.

Here also we will focus upon additive properties of polynomials i.e. additive aggregation. We have already seen that other aggregation functions like average, max, min etc can be derived from additive function.

For securing the data communication among sensor nodes we need a key distribution mechanism, in this technique we follow the random key pre-distribution. There are three phases in this scheme:

1) There is a key pool which has $M$ keys. Every sensor node can store $N$ keys in itself. For each sensor node $N$ keys are randomly selected from the key pool. $P_e$ is the probability that the two sensors have at least one same key.

2) Each sensor node sends out discovery messages to find out which neighbors share a common key with itself. If two neighbors share a common key, then a secure link is set up.

3) A path-key is assigned to pairs of sensors who do not have keys but can be connected by multi-hop secure links.

B. Data Aggregation Protocol

Let a seed $k$ in each sensor node, which is different from each other. It must be given to each node before using it. We are using range of $k$ is 1 to $M$, where $M$ is the number of sensors in the network.

As in CPDA the very first step here is cluster construction. For this cluster formation we use the previous first approach. Where a query server broadcast hello message and all others on receiving message selects them self as the cluster head with the predefined probability or wait for the message to come over it. This process of cluster creation resists until all the nodes are present in some cluster.

Suppose a cluster contains three sensor nodes A, B and C. Then-

- All The nodes within the same cluster broadcast their seeds $k_1$, $k_2$ and $k_3$. So A, B and C share the common knowledge of $k_1$, $k_2$ and $k_3$. Then all nodes in the cluster randomly generate a positive number within 100.

![Fig. 6: Message Exchange in Modified CPDA](image)

- Now node A calculates:

$$V_{A2} = k_2x + a_1$$

$$V_{A3} = k_3x + a_1$$

Where $x$-private data kept in A, $a_1$-random number generated by A.
Similarly node B calculates:

\[ V_{A2} = k_2 y + a_2 \]
\[ V_{B3} = k_2 y + a_2 \]

Where \( y \)-private data kept in B, \( a_2 \)-random number generated by B.

Similarly node C calculates:

\[ V_{C2} = k_2 z + a_3 \]
\[ V_{C3} = k_3 z + a_3 \]

Where \( z \)-private data kept in C, \( a_3 \)-random number generated by C.

- Now node A encrypts \( V_{A2} \) and \( V_{A3} \), then send them to node B and C respectively. Node B encrypts \( V_{B3} \) and send it to node C. Similarly node C encrypts \( V_{C2} \) and send it to node B.

- Now node B can calculate:

\[ F_B = V_{A2} + V_{B2} + V_{C2} = k_2 (x + y + z) + a_1 + a_2 + a_3; \]

Node C can calculate:

\[ F_C = V_{A3} + V_{B3} + V_{C3} = k_3 (x + y + z) + a_1 + a_2 + a_3; \]

- Now, node B send \( F_B \) to node C. Since C has the knowledge of \( F_B, F_C, k_2 \) and \( k_3 \), so it can find the sum of \( x, y \) and \( z \) without the knowledge of \( x \) and \( y \).

Figure 6 shows the process of message exchange during the scheme.

IV. INTEGRATION OF BOTH SCHEMES AT A GLANCE

In both the schemes we got different approaches to preserve the privacy while aggregating the data and sending it to the query server. From both the schemes we find three different ideas to preserve the privacy:

1) CPDA
2) SMART
3) Modified CPDA

Now we are going to compare all the approaches as discussed in the schemes on the basis of computation overhead and communication overhead. We compare all of them with a commonly used data aggregation scheme TAG(Tiny-Aggregation), where no data privacy is provided.

A. Communication Overhead

CPDA and SMART use data-hiding techniques and encrypted communication to protect data privacy. This introduces some communication overhead. Figure 7 shows the communication overhead of TAG, CPDA with \( p_c = 0.3 \), and SMART with \( J = 3 \) under different epoch durations. From figure 7 we can see that the message overhead in CPDA is less than twice that in TAG and the overhead of SMART with \( J = 3 \) is double that of TAG.

As in paper if we simulate the process of data aggregation and set the same parameters for Modified CPDA and CPDA. Then It calculates the bits transmitted among the nodes and find out the average bits sent out by a sensor node. For example, if a sensor node wants to send out \( k \) which it calculates. Then the bits it sends out is \( b = 56 + \log_2(k) \), where 56 bits is the header. Simulation result is shown in Figure 8.
From the Figure 8 we can see that Modified CPDA use less data in communication than CPDA. In CPDA the data increase linearly with the addition of the number of the nodes in a cluster. In Modified Scheme it increases in a comparatively slow manner.

**B. Accuracy**

If we think about the ideal situations when there is no data loss in the network, both CPDA and SMART are good enough to get 100% accurate aggregation results. However, in wireless sensor networks, due to collisions over wireless channels and processing delays, messages may get lost or delayed. So the aggregation accuracy is directly affected. It define the accuracy metric as the ratio between the collected sum by the data aggregation scheme used and the real sum of all individual sensor nodes. A higher accuracy value means the collected sum using the specific aggregation scheme is more accurate. An accuracy value of 1.0 represents the ideal situation.
Figure 9 shows the accuracy of TAG, CPDA (with \( pc = 0.3 \)), and SMART (with \( J=3 \)). Here we have one observation:

The accuracy increases as the epoch duration increases.

Reasons are:

- In the larger surveillance time, packets to be sent within the duration are high thus the chances to collide will be less.
- In the larger surveillance time, the packets will have a better chance of being delivered within the deadline.

TAG has better accuracy than CPDA and SMART because without the communication overhead of using privacy preservation, there will be fewer data collisions.

V. CONCLUSION

As we have seen the importance of data privacy is the major problem in various applications as many civilian applications require privacy, without which individual parties are reluctant to participate in data collection. These approaches proposed three privacy-preserving data aggregation schemes—"CPDA, SMART and Modified CPDA" focusing on additive data aggregation functions. We compared the performance of our presented schemes to a typical data aggregation scheme—TAG.

In Future we can work upon designing schemes for data privacy for general aggregation functions and we can further find out some robust privacy-preserving data aggregation schemes under outside attacks.

REFERENCES