Link-Based Privacy-Preserving Data Aggregation Scheme in Wireless Sensor Networks

Kai Zhang[1,2], Haiping Huang[1,2,3(✉)], Yunqi Wang[1,2], and Ruchuan Wang[1,2]

[1] College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
hhp@njupt.edu.cn
[2] Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing 210003, China
[3] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016 ,China

Abstract. Data privacy-protection is of great importance during data aggregation in Wireless Sensor Networks. A distinctive data aggregation scheme based on data link is proposed in this paper. To be specifically, the data link is formed according to energy consumption and distance. For each round of the data aggregation, nodes within a certain cluster perform data aggregation together by subtracting the base value(given by cluster head) from its real value, and then add the random number (generated by itself) for privacy protection. The cluster head will form the information matrix according to the data from the link, and then perform homomorphic transformation. Finally, the data reach the base station which will feed back the aggregation results effectively. Compared with previous work, our scheme can effectively protect data privacy and cause low computation overhead and energy consumption. Meanwhile, the base station can acquire the correlation between nodes in certain clusters.

Keywords: Wireless sensor networks · Data aggregation · Privacy · Data link · Homomorphic transformation

1 Introduction

Being composed by various tiny sensing nodes, Wireless Sensor Network (WSN) is used to monitor the environment nearby. It has greatly changed how people focus on and interact with the environment. However, energy of these sensors are strictly limited, data generated by neighbor nodes are relatively overlapped and redundant. Therefore, it is more meaningful to transfer those processed and essential data than raw ones during the data aggregation in WSN. The goal of data aggregation is to reduce computation overhead and energy consumption by all nodes processing the data together, and it can be used to do sum, average and min/max operation in WSN.

The data privacy owns the highest priority in data aggregation. For example, medical data like blood sugar and blood oxygen which concern the patients' privacy should be protected in Smart Medical System based on WSN. However, these sensitive data are

F. Chen and Y. Luo (Eds.): Industrial IoT 2017, LNICST 202, pp. 119–129, 2017.
DOI: 10.1007/978-3-319-60753-5_13

being transferred through wireless channels which will be easily attacked and thus exist the high risk that the private data may be exposed.

Some traditional solutions, like complicated encryption algorithm and data mining are not suitable for the special environment of WSN. In addition, sensor nodes in WSN are resource-restricted, so certain schemes that can protect data privacy and meanwhile reduce energy consumption are needed.

In this paper, we propose a link-based privacy-preserving data aggregation scheme (LPDA). Compared with previous work, our scheme presents the following advantages: (1) By introducing the base value, the amount of data is greatly reduced. (2) The data link can be used effectively and repeatedly. (3) The base station can acquire the correlation among nodes in certain cluster.

The paper is organized as follows: Sect. 2 summarizes the related work; Sect. 3 briefly introduces the models in this paper; Sect. 4 describes the procedures of our proposed scheme LPDA in detail; Sect. 5 evaluates LPDA and the paper is concluded in Sect. 6.

2 Related Work

Aiming at the data aggregation in WSN, many effective schemes have been proposed by researchers. Madden put forward the classic TAG scheme [1], Intanagonwiwat [2] and Bista [3] also proposed relevant data aggregation schemes. However, these schemes are all based on trusted environments. In reality, the WSN is probably being deployed in open environment, the attackers will capture and manipulate the nodes. In addition, sufficient privacy protections are not involved in these schemes.

Feasible schemes for privacy protection in WSN can be divided into data perturbation, secure multi-party computation, homomorphic encryption and polynomial regression. Among these four categories, secure multi-party computation has the advantage of low energy consumption and high level of privacy-preserving. W. He put forward two effective privacy-preserving schemes: CPDA and SMART [4]. CPDA, characterized by complicated inter-cluster computation, is in fact secure multi-party computation. It can effectively protect data privacy, but nodes have to interact with each other frequently and thus causing high computation overhead and energy consumption. SMART is based on data-slicing. Each node will slice its data into several pieces, encrypt these pieces and send them to its neighbors. And finally each node will pass data pieces to its parent along the tree structure. It has to be noticed that this scheme is also of high consumption. Sheikh R. put forward a k-sum secure protocol [5] to protect data privacy through secure multi-party computation. Shi J. proposed a privacy-preserving scheme PriSense [6] which uses city sensing as its background, defends conspiracy attacks through data-slicing and mixing techniques and thus provides sound privacy-preserving abilities. Shi E. put forward a privacy-preserving data aggregation scheme by using time series [7]. This scheme is carried forward through the encrypted data uploaded by nodes, but a trusted key manager is needed. Jung T. suggested several data aggregation algorithms to perform sum operation [8] based on the hypothesis that all the channels and nodes cannot be trusted. But the polynomials are partly public in this scheme, causing

information leakage during computation and communication. Wang proposed an efficient data aggregation scheme [9] with secure channel and data-slicing as its main techniques. Zhang [10] is the first to put forward the data aggregation scheme through data link. Compared with previous work, this scheme has the advantage of low computation overhead. For each round of the data aggregation, the cluster head will generate a random number and carries this number along this link to perform sum operation, and finally the cluster head subtract the random number from the results to get actual aggregation results. But this scheme fails to make the best use of the natural advantage of data link.

3 Models

3.1 Network Model

In this paper, the network model is a connected graph. Sensor nodes can be divided into three categories: base station that at the top of the entire network, cluster head and leaf node which gather data and upload data. Being different from the traditional data structure—tree, the plane structure—data link is used in our proposal. And this paper mainly focuses on sum operation, but other operations like average and variance can be done through some modifications.

The network model in this paper has the following features: (1) N sensor nodes are randomly distributed in the entire area. (2) All the nodes have the same communication range R and sensing abilities. (3) All the nodes have the sufficient initial energy to support the proposed scheme. (4) The base station is aware of the location of each node.

3.2 Security Model

This paper is based on the semi-honest model that each node executes the protocol strictly and correctly but it will try to capture or reveal others' private data. And meanwhile, the attackers will attempt to eavesdrop the raw data, capture nodes and tamper with the message.

We use the key distribution scheme in [11], which has three phases: key pre-distribution, shared-key discovery and key-pairs formation. For the key pre-distribution phase, a large key pool with K keys will be generated. Each node in the WSN will select k keys from the key pool randomly to form a key ring. For the shared-key discovery phase, nodes will discover neighbors with the same keys through the share of information. If two neighbors have the same key, a secure channel will be formed between them. And for the key-pairs formation phase, if there is no same keys between neighbors but they can reach each other through several hops, a channel key will be generate between them.

The data aggregation in WSN must satisfy the following requirements: (1) Privacy. (2) Accuracy, that no packages loss occur during aggregation and communication. (3) Efficiency, to reduce computation overhead and energy consumption.

4 Link-Based Privacy-Preserving Data Aggregation Scheme

4.1 Clustering

The clustering procedure can be described as follows: (1) Information is stored as set for each node. (2) Node will become the cluster head according to the probability *Pc*. (3) The base station will divide the entire network into layers according to the distance between each node and the base station, which is a significant factor that affects *Pc*.

The information set of *Nodei* can be described as <*NL, C, PPi, PDi, Er, Ec, P*>, *NL* is the neighbor set of *Nodei*, which is {*'Nd':distance, 'Nc':cluster*} in detail; *Nd* is the distance between the neighbor node and *Nodei*; *Nc* is the ID of a cluster that the neighbor belongs to; *PPi* stands for the random number for privacy-preserving for *Nodei*; *PDi* stands for the real value of *Nodei*; *Er* and *Ec* are the remaining energy and communication cost of *Nodei*, respectively; and *P* is the parent of *Nodei*.

Before clustering, the base station will divide the entire area into several layers, shown in Fig. 1. Since the base station is aware of the location of each node, so layers can be defined as:

$$L = \frac{d_{max}}{R} \tag{1}$$

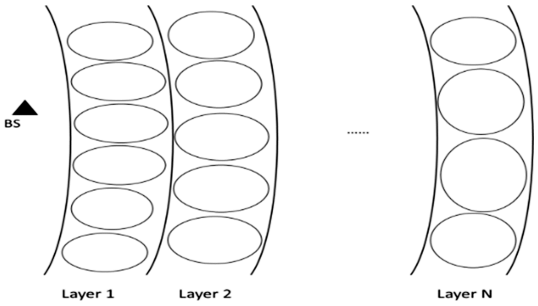

Fig. 1. Layers of the network

Where d_{max} stands for the furthest distance against the base station and R is the communication radius. The probability *Pc* is defined as:

$$P_c = \frac{1 - \frac{E_c}{E_r}}{L} \tag{2}$$

To be specifically, for the clustering, the base station will trigger a CLUSTER message to perform one query Q = (class = CLUSTER) ∩ (attribute = NONE). The message is described as <*NL0, C0, P0*> and for the base station, *NL0* = *C0* = *P0*. When receiving the CLUSTER message, node becomes the cluster head according to the probability *Pc* and when it becomes the cluster head, it will forward the CLUSTER message, or it will wait CLUSTER message from other nodes and select one cluster to

join in. For each node, upon receiving the CLUSTER message, it will update its neighbor set and if it joins a certain cluster, it will update its *C* and *P*. For example, node *a* sends the CLUSTER message to node *b* and node *b* decides to join in. It will update the neighbor set *NL*, update *C* to be the cluster that *b* belongs to and *P* to be node *a*. The process above will be done repeatedly and finally several clusters will be formed.

After the clustering, we may focus on the following issues:

(1) Nodes within each cluster should be no less than three. If there is only one node, apparently, the data aggregation operation cannot be performed, so it will join the cluster of its parent node. And if there is only two nodes, such cluster is too small in size and has low level of privacy and under such circumstance, we have one cluster head and one none-head node, so both of them will join the cluster of the parent node of the cluster head.
(2) Layer *L* should be inversely proportional to the probability *Pc*. This means that those nodes that are closer to the base station will have higher probability to become the cluster head for the closer the distance is, the more the amount of data, communication overhead and energy consumption. So more cluster heads around the base station shall bear more stress in the entire network.
(3) The less the energy ratio *Ec*/*Er*, the more 1 – *Ec*/*Er*. This means those nodes who become cluster nodes have the feature of low energy consumption.

The pseudo code of clustering are as follows:

```
L = dmax/R;
for each node Nodei do
    Pc = (1-Ec/Er)/L;
    if receives CLUSTER message form Nodej
         update NL;
         if Nodei elect itself as cluster head with Pc
               construct    CLUSTER    message    and
         broadcast;
         else if join the cluster
               Ci = Cj;
               Pi = Pj;
         end if
   else
         wait for next CLUSTER message;
   end if
   if node within a cluster < 3
         join the cluster of head's parent;
   end if
end for
```

4.2 Formation of Data Link

The meaning of constructing data link lies: forming a link to perform data aggregation which can be used effectively and repeatedly, and accomplishing initialization and information storage before data aggregation. All the nodes within a certain cluster have

received the CLUSTER message from the cluster head, so in the neighbor set of the cluster head, the exact locations of each node are stored. The energy ratio *Ec*/*Er* computed during clustering will be sent to the cluster head before the formation of data link.

It is obvious that the one significant feature of data link is that the last few hops in the link shall take responsible for more data packages and thus consuming more energy. So based on that, the cluster head will search for those nodes that are far away from it (greater than a threshold that has been previously set). And for the area nearby, the cluster head will choose the routes according to the energy ratio in the descending order. Here the less the energy ratio *Ec*/*Er*, the less communication cost and the more remaining energy a node has, and these are the exact nodes that are suitable to be the last few hops within the data link. The cluster head keeps above procedures repeatedly until it reaches the last node within its cluster.

Subsequently, a link that starts from the cluster head, goes through all the modes and finally goes back to the cluster head will be formed. All the none-head nodes within in certain clusters will generate a random number *PPi*, and then use the key between itself and the base station to encrypt the number and send it to the cluster head. Upon receiving these encrypted data, the cluster head will sort these data in the sequence as the data link, and send the data to the base station along with the ID of these nodes. The base station will decrypt these data and store them locally. And then the base station is able to query the random number of none-head nodes within a certain cluster according to the mark of the cluster.

The pseudo code of formation of data link are as follows:

```
generate a list called List1
generate a list called List2
for each Nodei do
   if distance between head and Nodei > threshold
         join List1;
   else
         join List2;
   end if
end for
for each Nodei in List2 do
   sort the list according to energy ratio;
   join List1;
end for
generate PPi and send it to base station;
```

4.3 Aggregation Within Clusters

Different from the traditional data aggregation technique that each node within the cluster uses their complete sensing data, in this paper, we require the data that each node uploaded is the one that subtract the base value from its real value. The benefit of doing this is that the amount of data is greatly reduced and thus both communication overhead and energy consumption are decreased. So the base value for each round of aggregation

obtains the priority because extreme results will occur if the base value given is too small, too large or not reasonable, and the data in the link will show signs of left-skewed or right-skewed.

We will take the example of a common sensing data item in WSN—temperature to show how the base value is given. The key to choose base value is to modeling the history data. Temperature is the kind of data that changes frequently, but take a long-term view, it is relatively stable. That is to say, the temperature fluctuate against a certain mean value, which obviously has the horizontal pattern statistically. So we model the temperature data by using the exponential smoothing.

The equation for exponential smoothing can be described as:

$$F_{t+1} = \frac{1}{n-1} Y_t + \left(1 - \frac{1}{n-1}\right) F_t \tag{3}$$

where F_{t+1} is the base value for time series $t + 1$, Y_t is the mean value (the mean of real values for all nodes) and F_t is the base value for time series t, n stands for the number of nodes within a cluster.

Before the data aggregation, the nodes within the cluster will sense and gather data in advance and send the data to the cluster head. The cluster head will do a mean operation on these data so as to give the first base value for the first round of data aggregation. After the aggregation, the cluster head will calculate the mean value, and according to Eq. (3), it gives the base value for second round of the aggregation and so on.

The packages in the data link is an array in the shape of $1 * (n - 1)$. Each node will subtract the base value from its real sensing value *PDi* and then add its private random number *PPi*, denoted as *X*, send it to the next hop along the link until it reaches the cluster head.

4.4 Aggregation Among Clusters

Take a cluster of five nodes as an example, after the cluster head receives the data package, it will do further processing on the data. It computes the subtraction between each *X*, and fill them into the following matrix:

$$\begin{matrix} 0 & X_1 - X_2 & X_1 - X_3 & X_1 - X_4 \\ X_2 - X_1 & 0 & X_2 - X_3 & X_2 - X_4 \\ X_3 - X_1 & X_3 - X_2 & 0 & X_3 - X_4 \\ X_4 - X_1 & X_4 - X_2 & X_4 - X_3 & 0 \end{matrix}$$

It can be seen that this matrix consists two triangular matrixes, so the cluster head only computes one triangular matrix, and then fill the negatives in corresponding positions.

Furthermore, we may get the adjacency matrix of the topological graph for the correlation of the nodes within the cluster by subtracting random numbers of each node, and this will be done by the base station. To take the security to a higher level, we can conduct the homomorphic transformation to raw topological graph, and transfer the processed graph in the network. The simplest homomorphic transformation can be finite

exchange of rows and columns in the matrix. Each cluster head shares the information of its own homomorphic transformation with the base station.

After the transformation, the cluster head finds the parent node according to its information set and sends the processed matrix. Finally, the base station restores the matrix. To do that, the base station will query and subtract the private data *PPi* of each node through the mark of the cluster and solve some simple equations to get the value *x*(the real value being subtracted by the base value).

If it is just to perform aggregation operation, the base station has no need of restoring original sensing value (real value) for each node. Through (4) the sum of a certain cluster can be acquired and through (5) the mean value *rb* can be computed. And the base station will feed these data back to each cluster head.

$$sum = \sum_{i=1}^{n-1} x_i + b * (n-1) \tag{4}$$

$$rb = \frac{\sum_{i=1}^{n-1} x_i}{n-1} + b \tag{5}$$

where b stands for the base value in a certain round of aggregation.

5 Analysis and Evaluation

In this section, we will compare our scheme LPDA with the classic TAG in [1], CPDA in [4] and RPDA in [10], so as to analyze the advantages of our scheme from different perspectives.

5.1 Privacy

For the key-distribution phase, we selected k keys from the key pool of K keys, for any pair nodes, the probability that they have common keys, namely the connectivity $P_{connect}$:

$$P_{connect} = 1 - \frac{((K-k)!)^2}{(K-2k)!K!} \tag{6}$$

In addition, we define the probability that a certain node can eavesdrop encrypted message as $P_{overhear}$, which means there is a third party that obtains the same key:

$$P_{overhear} = \frac{k}{K} \tag{7}$$

We suppose that there are 1000 keys in the key pool, and we have selected 200 keys from the key pool in the key-distribution phase. So $P_{connect} = 98.3\%$ and $P_{overhear} = 0.2\%$, which means that the probability that encrypted message being eavesdropped is relatively low. In the following, we will analyze the ability for resisting such attacks of our proposed scheme:

(1) LPDA can effectively resist eavesdrop attack: the data being transferred in the link have been encrypted. Even the attacker somehow obtains the plaintext, the private data in the plaintext are mixed with random numbers, so the attacker cannot acquire the private data.
(2) LPDA can effectively resist conspiracy attack: suppose there are two nodes in one cluster attempt to strike conspiracy attack over a third node. If these two nodes are none-head nodes, as illustrated above, the private data cannot be acquired; and if one of these two nodes is the cluster head, though it has the key to the third node, it still cannot obtain the private data for the data is protected by the random number of the third node which is only known to itself and the base station.

Furthermore, the cluster head has performed homomorphic transformation on the information matrix which makes it harder for attackers to acquire the raw data.

5.2 Communication Overhead

Figures 2 and 3 describe the changes of communication overhead against the network size and epoch duration. TAG has the lowest communication cost for it has no security protection. Nodes in TAG generally send two messages: one for constructing aggregation tree and the other for data aggregation. CPDA has the highest communication overhead which increases dramatically as the network size expands. This is because the cluster head has to send at least four messages and nodes within the cluster have to send at least three massages for the complicated computation. RPDA has relatively low communication overhead for the cluster head sends two messages and nodes within the cluster send one message. As for LPDA, none-head nodes only perform light-weight computation like subtraction and cluster head only performs computing base value and homomorphic transformation, thus greatly reducing the communication overhead. Overall, the communication overhead of LPDA increases as the network size expands, it is relatively low and slightly over TAG, and the epoch duration has low effects on LPDA.

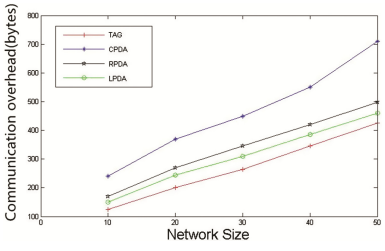

Fig. 2. Communication overhead against network size

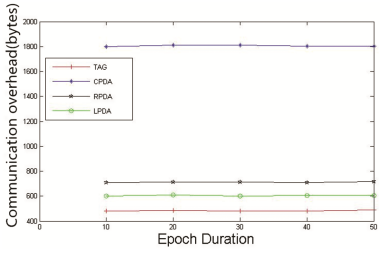

Fig. 3. Communication overhead against epoch duration

5.3 Energy Consumption

Figure 4 reveals the trends of average remaining energy with the increase of query times. TAG has the lowest energy consumption for it has no security protection. The average remaining energy drops significantly because it has to perform complicated computation and nodes have to frequently exchange messages. The performance of RPDA is better than CPDA. And the average remaining energy of LPDA is more than RPDA and relatively less than TAG, because the cluster head is chosen wisely, and the burden of data packages are bear by those nodes with better performance according to our strategy, and furthermore, the size of the data being transferred as relatively small.

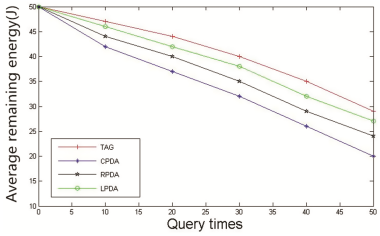

Fig. 4. Remaining energy against query times

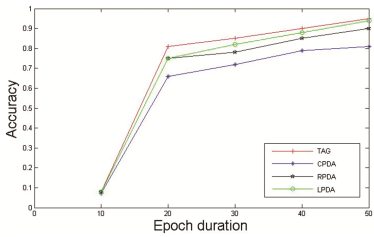

Fig. 5. Accuracy against epoch duration

5.4 Accuracy

Figure 5 shows the changes of accuracy against epoch duration. Under the ideal circumstances, without packages loss, the base station should have the 100% accurate results. Here we define accuracy as the ratio of the aggregation data against actual results from all nodes. We may see that the accuracy increases together with the epoch duration, this is because the larger the epoch duration, the less probability of data collision through data aggregation. Overall, the accuracy of LPDA is slightly less than TAG, for the base station can acquire the correlation of nodes and calculates the aggregation results through light-weight computation.

6 Conclusion

We propose the LPDA scheme to address the privacy-preserving in data aggregation and meanwhile focus on energy consumption. LPDA is based on data link and the data being transferred through the link is relatively light. The cluster head provide further security protection by performing homomorphic transformation on the data. And finally the base station can acquire abundant and accurate data. The simulation results show that our scheme is feasible, secure and effective. Our future works include: (1) Provide data integrity protection. (2) Dynamically form the data link according to the remaining energy in real time.

Acknowledgement. The subject was sponsored by the National Natural Science Foundation of P.R. China (No. 61373138, 61672297), the Key Research and Development Program of Jiangsu Province (Social Development Program, No. BE2015702), Postdoctoral Foundation (No. 2015M570468, 2016T90485), the Sixth Talent Peaks Project of Jiangsu Province (No. DZXX-017), the Fund of Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks (WSNLBZY201516), Science and Technology Innovation Fund for Postgraduate Education of Jiangsu Province (No. KYLX15_0853).

References

1. Madden, S., Franklin, M.J., Hellerstein, J.M., et al.: TAG: a tiny aggregation service for ad-hoc sensor networks. ACM SIGOPS Oper. Syst. Rev. **36**(SI), 131–146 (2002)
2. Intanagonwiwat, C., Estrin, D., Govindan, R., et al.: Impact of network density on data aggregation in wireless sensor networks. In: Proceedings of the 22nd International Conference on Distributed Computing Systems, pp. 457–458. IEEE (2002)
3. Bista, R., Kim, Y.K., Chang, J.W.: A new approach for energy-balanced data aggregation in wireless sensor networks. In: Ninth IEEE International Conference on Computer and Information Technology, 2009 (CIT 2009), vol. 2, pp. 9–15. IEEE (2009)
4. He, W., Liu, X., Nguyen, H., et al.: PDA: privacy-preserving data aggregation in wireless sensor networks. In: 26th IEEE International Conference on Computer Communications (INFOCOM 2007), pp. 2045–2053. IEEE (2007)
5. Sheikh, R., Kumar, B., Mishra, D.K.: Privacy preserving k secure sum protocol. arXiv preprint arXiv:0912.0956 (2009)
6. Shi, J., Zhang, R., Liu, Y., et al.: Prisense: privacy-preserving data aggregation in people-centric urban sensing system. In: Proceedings of IEEE INFOCOM 2010, pp. 1–9. IEEE (2010)
7. Shi, E., Chan, T.H.H., Rieffel, E., et al.: Privacy-preserving aggregation of time-series data. In: Proceedings of NDSS, vol. 2, pp. 1–17 (2011)
8. Jung, T., Mao, X.F., Li, X.Y., et al.: Privacy-preserving data aggregation without secure channel: multivariate polynomial evaluation. In: Proceedings of IEEE INFOCOM 2013, pp. 2634–2642. IEEE (2013)
9. Wang, T., Qin, X., Liu, L.: An energy-efficient and scalable secure data aggregation for wireless sensor networks. Int. J. Distrib. Sens. Netw. **9**, 843485 (2013)
10. Zhang, X., Chen, H., Wang, K., et al.: Rotation-based privacy-preserving data aggregation in wireless sensor networks. In: 2014 IEEE International Conference on Communications (ICC), pp. 4184–4189. IEEE (2014)
11. Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In: Proceedings of the 9th ACM Conference on Computer and Communications Security, pp. 41–47. ACM (2002)