Journal of Green Engineering (JGE)

Volume-10, Issue-12, December 2020

A Secure and Light Weight Privacy Preserving Data Aggregation Algorithm for Wireless Sensor Networks

[1]M.M. Naresh Babu, [2]A.S.N. Chakravarthy, [3]Cherukuri Ravindranath

[1,2]Dept. of Computer Science & Engineering, JNTU Kakinada, Kakinada, A.P., India
[3]Dept. of Computer Science & Engineering, Christ University,Bengaluru, Karnataka, India.E-mail: itsnaresh4u@gmail.com, asnchakravarthy@yahoo.com, ravindranathc@gmail.com

Abstract

WSN is a collection of sensors, which senses critical information related to military, opponent tracking, patient health details etc. These sensed critical and private data will be collected and aggregated by aggregators and forward it to the base station. Due to the involvement of sensitive data, there is a demand for secure transmission and privacy preserving data aggregation. In this paper, we propose a light weight, secure, multi party, privacy preserving data aggregation scheme, in which one or more sensors share their private data with aggregator securely without revealing the original content. The aggregators also perform the aggregation operation without knowing the original content.

Keywords: Privacy preserving, Data Aggregation, WSN, Wireless communication

1 Introduction

In today's computer world, data collection is a rapidly evolving and challenging process in various areas, such as unfriendly and low maintenance, wherein traditional methods stated to be extremely costly. For

Journal of Green Engineering, Vol. 10_12, 13329-13336.

these applications, sensors provide a low-priced and simple solution. These physical devices are small and can collect information about an item 's environment, such as heat, light or movement. Sensors are used in a simple model tomonitor events and collect data on the environment in the area of interest. Networking of such sensors will have an important effect on the performance of various military and civil applications, including battle field monitoring, defense and adversity management [1][5].

Sensor motes appear to have been throwing away for these processes and should last to drain their energy. The power of a sensor network and the lifespan of a particular task is thus an extremely inadequate resource. The life of the motes of the sensor must be extended wisely. The sensor networks adopt the models of a base station in which the sensors regularly or actionally send data streams to the base station. The controlling node / base station might well be positioned permanently in the area around the sensor, or it may be mobile to travel across the field and gather network data. While in any of the cases, all sensor motes in the network are not able to reach the base station strongly. Notes /nodes far from the base station need more power to relay data than other nodes, so they die easily [4][6][7].

Wireless Sensor Networks (WSNs) have become popular thanks to their ease of deployment and their energy autonomy. These networks are more and more considered for deployment in many aspects of our daily life. They are applied in several domains with various applications, among them we can cite: weather or volcanic earthquake monitoring, automation monitoring, health care monitoring and industrial plan monitoring. A WSN is a network composed of a large number of sensor nodes that sense the environment and communicate this information possibly via multiple hops to one or more collection points called sink. Data transmission is done through wireless links.

2 Proposed System

The sensor senses and stores the private or sensitive data. The aim is to transfer these private data to aggregator in converted form, such that, the aggregator cannot comprehend the original content of the data received, but able to perform aggregation operation. In order to model the behavior of the communicating entities, we adopt the semi-honest model [3]. A semi honest communicating entity will adhere to rules laid in the protocol, but it can use the intercepted information during the execution of the protocol to compromise the data privacy of other communicating entities.

3 Key Establishment

In our scheme, for secure transmission of data between two sensors

or between sensor and aggregator, pair wise symmetric keys will be established using matrices. Our scheme uses matrices for key generation which is light weight technique and consumes negligible resources from resource drained sensors. (server or aggregator or cluster head can be used interchangeably)

3.1 Matrix Generation Phase

Step 1: The base station B.S generates a pool of keys with more than 2^{20} distinct keys. The keys are selected from a Galois field GF(p), where p is a large prime number.

Step 2: B.S computes 'n' matricesMCS_ii = 1,2,3,4for each cluster head CH_j, where 'n' is the number of sensors under a cluster head CH_j. In our case, we are also assuming four sensors for each cluster head[2]. B.S selects a random 4*100 keys from the key pool and prepares 4, 10*10 square matrices i.e MCS_1, MCS_2 MCS_3, $MCS_4$for each sensor S1,S2,S3,S4 and pre loads these matrices into '4' sensors under CH_j.

Step 3: B.S computes a single 10*10 square matrixi.eMS and pre loads the matrix into all the sensors under a cluster head CH_j.

Therefore each sensor under a cluster head CH_j stores two 10*10 matrices. One is a unique matrix shared with its cluster head CH_ji.eMCS_j and another 10*10 matrix is a common matrix for all the sensors under CH_ji.e MS. The cluster head stores '4' different matrices for '4' sensors in its clusteri.eMCS_1, MCS_2, MCS_3, MCS_4 for S1,S2,S3 and S4 respectively. Due to space limitations for explaining our algorithm we use below 4*4 matrices.

$$M_{CS1} = \begin{bmatrix} 1 & 2 & 8 & 4 \\ 5 & 6 & 7 & 8 \\ 3 & 10 & 2 & 10 \\ 13 & 14 & 5 & 16 \end{bmatrix} M_{CS2} = \begin{bmatrix} 1 & 14 & 2 & 3 \\ 5 & 6 & 7 & 8 \\ 3 & 10 & 2 & 4 \\ 13 & 14 & 10 & 6 \end{bmatrix} MS = \begin{bmatrix} 1 & 10 & 2 & 3 \\ 14 & 6 & 7 & 8 \\ 3 & 10 & 2 & 4 \\ 7 & 8 & 10 & 6 \end{bmatrix}$$

3.2 Computing Pairwise Key between Aggregator and a Sensor

S1 will perform the following steps to compute a symmetric key with the cluster head CH_j. The steps are discussed below.

Step1: S1 selects a row no eg: 2 and a column no eg: 3 i.e<2,3> from MCS_1 and forwards the selected row and column combination <2,3> to CH_j.

Step2: On receiving <2,3>, CH_j also selects a row eg: 1, column eg:4, i.e<1,4> and performs following steps to compute the key.

Step 2.a) Write down the matrix elements belongs to the <row,col> combination sent by S1 i.e2ndrow and 3rd columnfrom the matrix MCS_1 = 5 6 7 8 8 7 2 5.

Step 2.b) Eliminate duplicates if any from 5 6 7 8 8 7 2 5, which results in 5 6 7 8 2.

Step 2.c) Write down the matrix elements belongs to the <row,col>combination chosen byCH_j himself i,erow 1 and column 4 from MCS_1 = 1 2 8 4 4 8 10 16.

Step 2.d) Eliminate duplicates if any from 1 2 8 4 4 8 10 16 which results in 1 2 8 4 10 16.

Step 2.e) Find out the common values from both the <row,col> combinations i.ecommon from 5 6 7 8 2 and 1 2 8 4 10 16 = 2, 8.CH_j, computes the pairwise key by adding the entire common values i.ekey = (2+8) = 10.

Step 3: After computing the key, CH_j, forwards his <row,col> combination. i.e<1,4> and h(key) i.eh(10) to S1

Step 4: On receiving {<1,4> and h(10)}, S1 also performs same steps by selecting the row 1, column 4 elements from MCS_1 and computes the hash of the sum of the common values. If the hash value received from CH_j and hash value computed by himare equal, then S1 authenticates CH_j and stores the key.

The base station B.S computes the matrices such that the any two <row, col> combination must have minimum one value common to compute the symmetric key.

3.3 Computing Pairwise Key between Two Sensors

As the sensors cannot exchange the data directly with each other, it must be routed through cluster head.If S1, S2need to exchange data, S1 and S2 first execute the pair wise key sharing algorithm with cluster head as discussed above. On computing the key with CH_j, S1 and S2 share pairwise key as follows:

Assume that S1 and CH_j shared secret key K1 = 20 and S2 and CH_j shared K2 = 30.Assume that the base station preloaded all the sensors under CH_j with the matrix MS as described above.

S1, CH_j, S2 will perform following steps to share a pair wise key between S1 and S2.

Step 1: S1 selects the <row,col>i.e<2,4> and submits {M1 = <2,4>⊕h(20)}. (20 is the key K1 exchanged between CH_j and S1 using the algorithm discussed in 2.1).

Step 2: On receiving the message M1 = <2,4>⊕h(20), CH_jgets the <row, col> combination sent by S1 i.e<2,4> = M1⊕h(20).

Step 3: CH_j computes M2 = <2,4>⊕h(30) and forwards it to S2. (30 is the key exchanged between CH_j and S2 using the algorithm mentioned above.).

Step 4: On receiving M2, S2 computes <2,4> = M2⊕h(30).

Step 5: S2 selects his own <row,col> combination, eg: <3,2> and computes the common values:

Step 5.1) Get row 2, col 4 values from MS = 14 6 7 8 3 8 4 6.
eliminate duplicates: 14 6 7 8 3 4

Step 5.2) Get row 3, col 2 values from MS = 3 10 2 4 10 6 10 8.
eliminates duplicates: 3 10 2 4 6 8 .

Step 5.3) Find out common values:3 4 6 8 .

Step 5.4) Compute the sum of common values: 3+4+6+8 = 21.

Step 6: S2 forwards {M3 = <3,2>⊕h(30), h(21)}, in which, the <row, col> combination chosen by S2 is XORed with hash of key shared with CH_j, and hash of key computed for S1.

Step 7: On receiving M3, CH_j computes M4 = M3⊕h(30)= <3,2>. On getting the <row,col> combination sent by S2, CH_j XOR's the <row,col> combination of S2, with the hash of key shared between himself and S1i.eh(20).

Step 8: CH_j forwards {M5 = <3,2>⊕h(20), h(21)} to S1.

Step 9: On receiving M5, S1 computes <3,2> = M5⊕h(20). S1 proceeds to compute the shared key as similar to S2 to get 21.

Step 10: S1 verifies h(21) equal to the hash of received value from S2. If both are equal, S1 believes S2.

Even though the <row, col>messages from S1 and S2 are exchanged through CH_j, as CH_j doesn't have access to MS matrix, it is impossible for CH_j to compute the session key which is computed between S1 and S2 and vice versa.

4 Secure and Light Weight Privacy Preserving Data Aggregation Algorithm

As discussed, in our scheme, a cluster consists of 4 sensor nodes. If four nodes are having secret datato transfer to aggregator, to satisfy SHM model[3], the others sensors nodes and aggregator should not intercept the original content of the sensor private data. The aggregator must able to compute the aggregation of data received from sensors, in this case SUM, without knowing the original sensor private data. The algorithm is as follows:

Step 1: S1, S2,S3 and S4executes the key sharing algorithm with aggregator (2.2), letthe keys shared be KA1, KA2, KA3, KA4.

Step2: S1 executes key sharing algorithm with sensor S2, as discussed in section 2.3, let the key shared be K12.

Step3: S1 computes {M1 = PriData1+KA1+K12} where PriData1 is the private data of sensor S1 and forwards M1 toaggregator.

Step4: CH_j forwards M1 to sensor S2. On receiving the message, S2 executes key sharing algorithm with S3 (2.3)to get K23. Now the message becomes M2 = {(PriData1+KA1+K12)+(PriData2+KA2+K23)}. Similarly S3 also follows the same steps.

Step 5: In case of sensor 4, which is the last one, no need to execute the key sharing algorithm with next sensor. S4 executes only key sharing algorithm with aggregatori.e(PriData4+KA4).

Step 6: The combined data results in M4 = {(PriData1+KA1+K12)+(PriData2+KA2+K23)+(PriData3+KA3+K34)+(Pri

Data4+KA4}.

Step 7: Sensor S4 forwards M4 to aggregator.

Step 8: All the sensors forwards their pair wise symmetric keys shared with other sensors i.e K12, K23, K34 to aggregator. On receiving the keys, aggregator computes (PriData1+PriData2+PriData3+PriData4) = M4- (K12+ K23+ K34)- (KA1+ KA2 +KA3+ KA4) as server knows the pair wise keys it shared between S1, S2, S3, S4 i.e KA1, KA2,KA3, KA4. (for next round of data aggregation, different set of keys are computed by sensors. Also B.S refresh the matrices periodically)

Hence, the aggregator, able to compute the SUM of the original data in privacy preserved way.

In our algorithm, we carefully restricted the data leakage between sensors and CH_j by using two kinds of keys i.e key shared between sensor and aggregator, key shared between two sensors based on two different matrices. As shown in fig1, the simulation results reveals that compared to recently proposed schemes i.e CPDA proposed by He et al and Jaydip et al, our scheme requires less computation cost. In our scheme, we have used matrices of size 10*10, light weight XOR and hash operations, which make our scheme light weight, robust and practically adopted into practical environments.

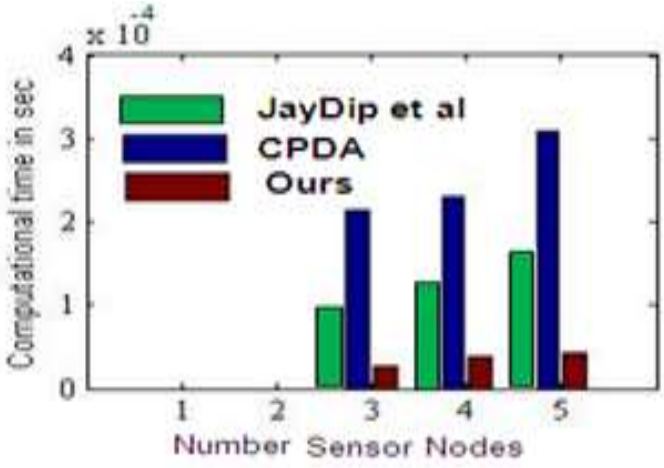

Figure 1 Comparision of Computation Costs

5 Conclusion

In this manuscript, we have made two contributions. We first proposed a light weight, secure, key sharing algorithm among sensors and aggregator based on matrices. Based on key sharing algorithm, we devised a light weight, secure, privacy preserving data aggregation algorithm for WSN. We have shown that our scheme is secure, prevents data leakage and requires less computation time compared to recently proposed schemes.

References

[1].He,W., Xue .L., Hoang .V.N., Klara.N., Tarek, A., "PDA: Privacy-Preserving Data Aggregation for Information Collection", ACM Transactions on Sensor Networks (TOSN), Vol.8, no.1, 2011.

[2]. Arijit, U., Jaydip,S., "A Secure Privacy Preserved Data Aggregation Scheme in Non Hierarchical Networks.Computational Science and its Application ICCSA, pp 436-451, 2011.

[3]. O. Goldreich,, "Secure Multiparty Computation,", ver 1.4, 2002.

[4]. Ali, Amiri., Reza, Barkhi., "The combinatorial bandwidth packing problem", European Journal of Operational Research ,Vol.208, no.1,pp.37-45, 2011.

[5]. Yogesh, Suresh, Gunjal., Yogesh, Kumar, Sharma., Satish, Ramchandra, Todmal., "Load Balancing And Bandwidth Optimization In Wireless Sensor Networks Using Energy Efficient Clustering Protocol (EECP)", International Journal of Advanced Science and Technology, Vol.29, no.12s, pp. 1019 – 1028, 2020.

[6].J. Wang., R. K. Ghosh., S. Das., "A Survey on Sensor Localization", Journal of Control Theory and Applications, Vol.8, no.1, pp.2– 11, 2010.

[7].H. Liu., H. Darabi., P. Banerjee., J. Liu., "Survey of Wireless Indoor Positioning Techniques and Systems", IEEE Transactions on Systems, Man, and Cybernetics, Applications, Vol.37, no.6, pp.1067–1080, 2007.

Biographies

M.M. Naresh Babu, Dept. of Computer Science & Engineering, JNTU Kakinada, Kakinada, A.P., India.

A.S.N. Chakravarthy, Dept. of Computer Science & Engineering, JNTU Kakinada, Kakinada, A.P., India.

Cherukuri Ravindranath, Dept. of Computer Science & Engineering, Christ University,Bengaluru, Karnataka, India.