Hidden Data Procession for Multiple Applications in Wireless Sensor Networks

Janarish Saju C¹, Sharon Nisha M²
1 PG Scholar, Department of CSE, Francis Xavier Engineering College, Tirunelveli, India
2 Associate Professor, Department of CSE, Francis Xavier Engineering College, Tirunelveli, India

Abstract—
In wireless sensor networks hidden data procession is the concept of collecting, summarizing and combining sensor node’s data in order to reduce the amount of data transmission in the networks. In previous studies we have found that homomorphic encryption algorithm have been applied to hide data during aggregation from sensor nodes. However, the principle involved in this algorithm does not satisfy multiple application in sensor environment, and second compromising node attack cannot be prevented and then finally the number of messages aggregated could be detected and whether it may be a duplicate copy, therefore a new scheme “Hidden Data Procession” has been introduced which is an extended form of Boneh et al’s homomorphic CRT algorithm such that the security schemes are applied using “Key Distribution” technique, since it has three methodology to satisfy the above mentioned problem. Initially it was designed mainly for multi-application environment and second it prevents compromising node attack and finally a special method of secure counting is applied here, to prevent unauthorized data sensed.

Keywords—Hidden data procession, Chinese redundancy theorem, key distribution techniques and wireless sensor networks.

I. INTRODUCTION

A wireless sensor network is a network consisting of various separated devices called sensors to detect environmental conditions. A wireless sensor network system provides the way of connecting many sensor nodes. It incorporates wireless connectivity of combining sensor nodes for a separated environment (see Figure 1). The wireless technology depends on your application requirements. The required applications include radio transmission, Wi-Fi connectivity, long term Bluetooth devices or satellite transmission etc. The WSN is built of “nodes” – from single to thousands of sensor nodes. Each and every sensor node has several method of construction: a radio transceiver with an internal antenna or connection to an outside antenna. Also there are many typical construction interfacing with the sensors networks. A sensor node might act like a planet surrounding the sun, which means the sun which acts like a group head. In case of sensor environment the each and every sensor node may considered to be cluster head according to the application requirements. The costs of sensor nodes are unpredictable depending on the utilization of sensor nodes.

II. RELATED WORKS

In [1] Perrig, A, et.al,(2011) proposes “SIA: Secure Information Aggregation in Sensor Networks” This is the paper which constructs framework for securing the data in sensor environment. Here the sensor nodes which act like the aggregator according to the query evaluated which strictly reduces the communication overhead, the information responded according to the query is in the form of average or median of the corresponding values. This can be achieved by constructing random sampling or interactive proofs. So even if the corresponding sensor nodes are corrupted, the sampling model of collected information provides the result to the user.
And also it enables the sub linear communication between aggregators and the users, and this was the first technology evolved for secure aggregating protocol that can handle malicious attack on sensor nodes.

In [2] Stankovic, et al. (2010) deals with “Security in Wireless Sensor Networks: Issues and Challenges” which describes a great purposes for various future applications, the inclusion of wireless communication technology in this paper includes several methods of security applications. The intent of this paper is to monitor the related security problems and occurrences in wireless sensor networks. This mechanism detects the security problems and proposes security solving applications for wireless sensor networks. This study also discusses the major view of security for implementing future, fast, and accuracy of security in wireless sensor networks. According to the use of layered codes, group heads of sensor environment does not know how the sensor data to perform data procession, which enables sensor nodes to communicate end-to-end secure communication with base station to the related sensor nodes.

In [3] David Evans, et al. (2007) focuses on “Secure Aggregation for Wireless Networks”, Here also the same criteria used as mentioned above such that the sensor nodes in proposed environment collects the data or information and distributed to the requested base station. To balance energy, intermediate sensor nodes should collects information from separate sensors nodes. However, this evaluates the risk of compromising any of the sensor nodes and provides false reading. In this paper the mechanism of new protocol was designed to avoid the compromising node attack. Here this protocol is designed with minimum energy or power, minimum cost and inexpensive sensor nodes.

In [4] Hasan Cam, et al. (2007) proposes a secure and energy considerable data aggregation protocol called ESPDA (Energy-Efficient Secure Pattern based Data Aggregation). ESPDA avoids the repeated data transmission from sensor nodes to group head called cluster head. If sensor nodes aggregate the same data repeatedly from sensor nodes, this approach gathers data and collected in the form of pattern code representation to determine the characteristics of data sensed. Cluster-heads collects the data aggregated and securely transmitting to the base station in the form of cipher texts. And this mechanism provides the way of communication by end-end process of data aggregation mechanism.

In [5] Cam. H, et al. (2008) proposes a secure data aggregation protocol, called SRDA. SRDA requires sensor nodes to send only the difference obtained data instead of all data aggregated by the sensor nodes. Effectiveness of the SRDA is managed by the key distribution technique in case of security purposes. SRDA establishes secure connectivity among sensor nodes. The incremental security requirement for data aggregation evaluates significant result’s that show SRDA technique yields preserve data security by minimizing the energy consumption.

III. SYSTEM DESIGN

The problem of aggregating data from sensor nodes directly to the base station requires more energy consumption, which is satisfied by forming the cluster node to collect data from a group of nodes; this principle is already proposed by traditional approaches. Another need for the problem is that it haven’t secured while collecting data for multi-application environment, since previous studies show that the security is made over only for single application environment. And finally third problem shows that the base station does not know how many times the sensor has been sensed, since there may be a chance for wrong update from unauthorized access. Therefore these problems need a quick remedy and solutions, which can be resolved by our “Hidden Data Procession” approach.

A. Existing System:

In wireless sensor networks data procession is the scheme of collecting data from several nodes and securely passing them to base station, which reduces the large amount of transmission, here the traditional approach of “Homomorphic Encryption” algorithm have been applied, this method supports for an effective single application environment, but there is a risk for multi-application environment, since the enciphering of data for several application cannot be aggregated together, because the decrypted aggregated result will be incorrect. And the existing methodology does not counts the number of aggregated messages, which results unauthorized update or editing on the cluster head and here there is no proved security for the compromised node attack of the same.

B. Proposed System:

The proposed scheme is the “Hidden Data Procession” method, which introduces the new extended form of “Boneh et al’s Homomorphic Encryption Algorithm”. This is a key distribution technique and is formally related with (CRT-Chinese Reminder Theorem) algorithm, which encrypts the data from multi-application environment, such that the cipher texts from different application can be encapsulated into only one cipher texts, whereas the corresponding base station can extracts the application related plaintext via the corresponding secret keys allotted. This scheme is specially designed to prove three contributions that do not satisfy previous studies of data procession technique.
in which the first contribution is that it was designed especially for multi-application environment, and second it mitigates compromising attack in cluster head, and finally it degrades the damage from unauthorized updating or editing sensor readings.

III. BGN Scheme

In 2006, Boneh et al. [3] proposed a public-key PH scheme, which integrates the Paillier [9] with the Okamoto-Uchiyama encryption schemes [2]. We call it BGN for simplicity. BGN provides the way of ensuring addition, multiplication and modulo homomorphism. Here the property of multiplication is based on pairing with bilinear operations. [3], is more expensive and complex [1]. Here we propose additive, multiplication, and modulo operation. In this paper, it shows the BGN scheme of operation for data collection. Here we modify BGN scheme to satisfy multiple applications in sensor nodes. The explanation about BGN is showed in Fig 2.BGN is evaluated in the form of elliptic curvepoints by representing cyclic points. However the point representation form an algebraic group, and similarly I denotes the identity points. [2]. Here (P) denotes the notation of order of P. Showing ord(P) ¼ q, it shows that it is the minimum integer of q represents q _ P ¼1. In the key generation function the order E is equal to the total number of points in E. The explained concept of E is mentioned above. The function of ENC is based on points G and H which is the scalar multiplications over that. The cipher text is designed in form of many parts. Where the scalar representation of points is considered to be point over G and Scalar of randomness is considered to be point over H. The homomorphic properties which proves the additive property; the scalar value of point p is added at the end gives the sum of total aggregated messages. Consequently, the final output gives the representation M _ G [ _ R _ H, Here M and R is the number of messages and effectiveness of randomness consequently. The decryption function DECRYPT gives the output M which is the number of messages aggregated. Here both the points G and H are different. By multiplying the results with private key the randomness of the point H is removed represents ord(H). Here the cipher texts contains only the product of H representing the function as ord(H) _ M _ G. In order to apply the value of M we should use discrete sample of algorithms. Here its efficiency is obtained by Pollard’s method. Now, we make a glance at the algorithm, when the sensor nodes S1 have sensed the reading M1, S1 encrypts M1 and gives the solution as C1, which is the cipher text. Then S1 sends the cipher text C1 to AGG which is an aggregator. When AGG receives all cipher text C1…Cn, which is given in the expression AGG( _ _ AGG(AGG(C1; C2) _ _ _Cn). Then it sends to the next generator.

KEYGEN(r): generate a public-private key pair
1. Based on security parameter r, it computes a triple elements, (q1, q2, E), where E is a set of elliptic curve points which form a cyclic group.
2. Randomly select two generators (i.e., base points) G, H, where ord(G) = ord(U) = n.
3. Compute point H = q1 * U such that ord(H) = q1.
4. Select parameter T as the total plaintext boundary and T < q2.
5. Output the public key: PK = (n, E, G, H, T).
6. Output the private key: SK = q1.

ENC(PK, M): Message encryption on M by public key PK.
1. Check if message M ∈ {0, · · · , T}.
2. Randomly select R ∈ {0, · · ·, n−1}.
3. Generate the ciphertext C as: C = M * G + R * H, where G, H ∈ PK.
4. Output C.

AGG(C1, C2): Aggregation on two ciphertexts C1, C2, where C1 = M1 * G + R1 * H and C2 = M2 * G + R2 * H.
1. Randomly select R’ ∈ {0, · · ·, n−1}.
2. Compute the aggregated ciphertext of (m1 + m2), C’ as: C’ = C1 + C2 + R’ * H = (M1 + M2) * G + (R1 + R2 + R’) * H.
3. Output C’.

DEC(SK, C): Message decryption on C by private key SK.
1. Compute logG(q1 * C) = logG(q1 * (M * G + R * H)) = logG(M * G) = M where G = q1 * G.
2. Output M.

Figure 2BGN scheme.

IV. HIDDEN DATA PROCESSION

BGN is proposed by the order of implementing two point constructions G and H which was described as earlier. By the representation of two points of order one of the orders can be removed by multiplying with response to aggregated solutions, and finally the scalar representation of another point is detected. By using the same logic hidden data
procession is applied for multiple representations. We can access one scalar operation of certain point by removing the effects of remaining points, this is done by multiplying the collection of encrypted cipher text along with the remaining points product. The security of hidden data procession concept and BGN are based on subgroup decision problem. Here we represent hidden data procession \((k \equiv 2)\) to derive how we use for multiple applications.

A. Hidden Data Procession \((k = 2)\) Construction

In this representation it shows that all the sensor nodes are grouped together separately. It contains four steps: encryption, decryption, aggregation and key distribution techniques. And similarly \((k=2)\) concept is applied here by using P,Q,H schemes whose order of representation is given by \(q_1,q_2,q_3\). The scalars of the first two points carry the aggregated messages in GA and GB, respectively, and the scalar of the third point carries randomness for security. As shown in the DEC functions, by multiplying the aggregated cipher text with \(q_2q_3\) (i.e., the SK in GA), the scalar of the point \(P\) carrying the aggregated message in GA can be obtained. Similarly, by multiplying the aggregated cipher text with \(q_1q_3\) (i.e., the SK in GB), the scalar of the point \(Q\) carrying the aggregated message in GB can be obtained. In this way, the encryptions of messages of two groups can be represented in the form of single cipher text, and the aggregated information of particular group or cluster can be encrypted by SK secret key it should be confidentially kept secret similarly the corresponding secret key should be known only by the base station. And the public key should be shared by the dispersed sensor node. And another major operation is the decryption procedure every encrypted result could be decrypted individually.

B. Generalization of Hidden Data Procession

Hidden data procession \((k \equiv 2)\) is generalised by expanding \(K<2\) representation. This generates different key pairs for each group of sensor nodes. To prove the efficiency of security the order of M could be expanded as large enough. Therefore when the value of \(k\) becomes more the efficiency of cipher texts also be more. For representing multiple application representation, the sensor nodes belonging to the applications are assigned with same public key. Under this concept the information from different sensor nodes are collected and encrypted in the form of single encrypted text and it can be transmitted to the base station. The base station decrypts the result individually from the single cipher text.

C. Key Distribution

There are two methods represented here one among them is key distribution. If we know the locations of deployed SNs, we can preload necessary keys and functions into SNs and AGs so that they can work correctly after being spread out over a geographical region. Key postdistribution. Before SNs are deployed to their geographical region, they are capable of nothing about hidden data procession keys. These SNs only load the key shared with the BS prior to their deployment, such as the individual key in LEAP [3] and the master secret key in SPINS [4]. Once these SNs are deployed, they can run the LEACH protocol [2] to elect the AGs and construct clusters. After that, the BS sends the corresponding hidden data procession keys, encrypted by the preshared key, to SNs and AGs.

V SECURITY ANALYSIS AND COMPARISON

In this section, we propose the security concepts applied in the hidden data procession. Here we compare hidden data procession security problem schemes.

A1. Ciphertext known attack. Which is the basic attack happened at the sensed node.

A2. Known plaintext attacks. Only CDA based on Domingo-Ferrer scheme [17] might suffer from this attack due to improper security parameters indicated by Wagner’s cryptanalysis [35]. However, the cost of proper parameters may render CDA infeasible to WSNs. For Casteluccia et al.’s scheme, although the previous encryption keys can be deduced by the pairs, no research shows that these keys help the deduction of the present or subsequent encryption key.

A3. Chosen plaintext attacks. If the scheme suffers from known plaintext attacks, then it also suffers from chosen plaintext attacks. Hence, CDA also suffer from this attack. Other schemes can defend against this attack because they are probabilistic encryption algorithms. It is hard to decrypta ciphertext by finding a match from known samples.

A4. Chosen ciphertext attacks. Unfortunately, all schemes suffer from this attack due to the homomorphic property. Assume that an adversary tries to decrypt the challenged ciphertext C \(\equiv E(M)\), where E (-) is a PH’s encryption function. The adversary can obtain the ciphertext C00 by adding C with a cipher text C0 \(\equiv E(M)\), where M0 is known. After that, she can decrypt C00 to its plaintext M00 by querying the decryption oracle. Consequently, she can obtain M by M \(\equiv M00\_M0\). Fortunately, it is difficult to launch this attack in WSNs because the adversary must have the ability to decrypt some chosen ciphertexts.
B1. Unauthorized aggregation. Since the aggregation of CDA requires only modular addition, an adversary may aggregate ciphertexts without additional information. Unlike CDA, encryption keys of SNs in Castelluccia et al. ’s scheme are generated dynamically for one-time use. Unauthorized aggregation probably results in an unexpected plaintext because the keys involved in these ciphertexts mismatch with those currently held by the BS with high probability. Since unexpected plaintexts can be observed by the BS, the impact of unauthorized aggregation is mitigated. For asymmetric schemes, EC-OU, Tiny-PEDS, BGN, and CDA are based on ECC. To aggregate ciphertexts, one has to know curve information. If the public key is preinstalled or delivered in a secure way, aggregation cannot be executed by an adversary without compromising SNs or AGs.

B2. Malleability. Castelluccia et al. ’s scheme suffers from this attack because of modular addition-based construction. For example, adding the value of plaintext is trivial by adding desired numeric value to the corresponding ciphertext directly. Other schemes based on modular multiplication (e.g., CDA) or those based on ECC can defend against this attack.

C1. B1/B2 under compromised AG. For CDA and Castelluccia et al. ’s scheme, compromising an AG will disclose the modulus; for ECC-based schemes, this will disclose the curve information. Except hidden data process scheme, revealing curve information makes unauthorized aggregation in other scheme easier. On the other hand, no malleability is still supported by all ECC-based schemes because point information stored in SN are not revealed.

C2. Unauthorized decryption under compromised SN. In CDA, when compromising an SN, an adversary can decrypt the aggregated ciphertexts because CDA is asymmetric scheme. Although Castelluccia et al. ’s scheme is also symmetric, it suffers from minor impact because each node is assigned a distinct key. On the contrary, EC-OU, Tiny-PEDS, BGN, and hidden data process do not suffer from this attack because they are asymmetric schemes.

C3. Unauthorized encryption under compromised SN. This is the strongest attack against which no schemes can defend. An adversary encrypts arbitrary values with the compromised secrets and alters the aggregated cipher text by the forged values. After aggregation, the polluted messages aggregated into the result would be difficult to remove or detect. Castelluccia et al. ’s scheme can mitigate the impact because the adversary cannot forge ciphertexts of uncompromised SNs. Similarly, hidden data process (k > 1) prohibits adversaries from forging cipher texts of SNs in uncompromised groups. Supporting more groups (i.e., bigger k) makes hidden data process more secure even if it brings additional cost; the size of ciphertexts increases linearly.

VI PERFORMANCE ANALYSIS

To perform the complex and computation cost of Tiny-PEDS, EC-OU, GN, and hidden data process are represented based on elliptical curve cryptographic representation. Here the process of encryption, decryption, and aggregation are based on two methods they are, addition by a point and scalar multiplication representation of point. In elliptic curve cryptographic arithmetic algorithm there are two basic operations are performed they are point doubling and adding. The representation of point adding shows the representation P + Q, Here Pand Q are curve points. And 2P is computed by point doubling concept. And similarly the representation r _ Q is computed by scalar multiplication representation. Since r is scalar. These two operations reproduce the half-and-add algorithm. More specifically, computing r _ Q requires around jrj doubling and jrj2 additions, amounting to about 3jrj2 point additions [11]. We have shown the cost relation between point addition and scalar multiplication.

Next, we show how to estimate the cost of scalar multiplication on different finite fields. In general, the cost depends on the size of the scalar and the size of underlying finite field. If the size of scalar doubles, the cost doubles too (i.e., linearly inclining). Moreover, if the size of the finite field doubles, the computation cost is almost four times the original (i.e., increasing by a power of 2). Based on these two rules, the cost of scalar multiplication on a 1,024-bit field is estimated to be 247.84 (i.e., 1024163 p3) times greater than that on a 163-bit field, where the scalar is chosen from the underlying field. Following the same analysis model in [11], we can estimate computation costs among these schemes. Let the base unit be the point addition on 163-bit field. For encryptions, Tiny-PEDS is the most efficient one because their curves are chosen from smaller fields. Tiny-PEDS can be built on smaller fields because its security is based on the hardness of elliptic curve discrete logarithm problem (ECDLP). In contrast to Tiny-PEDS, the security of EC-OU, BGN, and hidden data process are based on the hardness of integer factorization problem (IFP). Their curves have to be chosen from larger fields, resulting in higher encryption costs.
Performance gain of all the techniques are compared along with the graphical representation, for which it can be compared with three representations of categories they are end to end technology, database as a service and showing cloud as a platform. For each application there could be advantages and drawbacks. See (figure 3) in the above diagram.

A. Performance Gain of Hidden Data Procession

In the above process the computation cost is considerably high. And the data procession is high. This shows that sensors nodes to perform encryption and decryption are complex. To prove the efficiency evaluates the performance gain. Initially we should classify the sensor nodes based on their tasks, and be separated to specific cluster head according to the requirements. Leaf nodes from the tree gather information from the sensor deployed and finally sends the information back to the base station. Aggregated nodes are the sensor nodes which have the collection of information these information can be forwarded to the base station. During this process the energy consumption is measured such that the aggregated results could be minimised by reducing it to single information which is already described above. The data forwarding scheme (DFS) is enhanced to every aggregated node to forward the data to the neighbourhood node. For every data transmission a hop by hop process is enhanced and is secured by applying security mechanism which may be AES and the parent node which stores the encrypted data and finally the base station collects the information in the form of single cipher text. This reduces the amount of data transmission and complexity. Thus proves effectiveness of performance measures.

VII CONCLUSION

Multi-application environment,” Hidden Data Procession” is the first Data aggregation scheme. Through “Hidden Data Procession”, the encrypted cipher texts from different applications can be collected and combined in the form of single encrypted text. For a single-application environment, it is still more secure than other data aggregation schemes. Whereas it was the first scheme proposed to satisfy multiple information gathered. Whenever compromising attacks occur in WSNs, it provides the way of securing by counting the expected aggregated results. All of this function satisfies through the BGN scheme proposed. Thus it provides database as service in the form of aggregating queries.

In future we may propose PH scheme, which may use cloud as a platform for separating queries rather than database as a service and can be evaluated as much as possible for efficiency.

REFERENCES


