ISSN : 2231-0711

R. Divya et al | International Journal of Computer Science Engineering and Technology(IJCSET) | November 2014 | Vol 4, Issue 11,319-325

Trust Computation Model for Secure Data Aggregation (TCMSDA) in Wireless Sensor Networks

R. Divya[#1], K. Geethalakshmi[#2]

[#1] *Research Scholar*

Department of Computer Science, PSGR Krishnammal College for Women
Coimbatore, India

divya.gac@gmail.com

[#2] *Assistant Professor*

Department of BCA, PSGR Krishnammal College for Women
Coimbatore, India

geethalakshmi@psgrkc.com

***Abstract*— A 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. In the wireless sensor networks, data aggregation scheme is used to reduce the large amount of transmissions. In the wireless sensor network, security is an important concern. So, in order to overcome this problem a new concealed data aggregation scheme is used. It has three contributions: First, it is designed for a multi-application environment. The base station extracts application-specific data from aggregated cipher texts. Next, it mitigates the impact of compromising attacks in single application environments. Finally, it degrades the damage from unauthorized aggregations. In this method, the aggregator is selected based on the transmission range. But the drawback is less security and less efficient. So, in this paper, an innovative technique called Trust Computation Model for Secure Data Aggregation (TCMSDA) in wireless sensor networks is introduced. A new trust management scheme is essential to differentiate illegal and normal nodes and filter out the malicious nodes in the network. In the trust computation model, each node identifies trustworthiness of sensor nodes. This model suggests a defensible approach against insider attacks incipiently beyond standard authentication mechanisms and conventional key management schemes. Experimental results show that when compared to the existing system, the proposed system achieves more efficiency and high security.**

***Keywords*— Wireless Sensor Network, Data Aggregation, Trust Value.**

I. INTRODUCTION

Wireless Sensor Networks(WSN) have been used in many rich applications such as environmental/ habitat monitoring, acoustic detection, seismic detection, military surveillance, inventory tracking, medical monitoring, smart spaces, process monitoring, etc. Depending on the application SN read different kinds of data (e.g., temperature, light, or smoke). Typically, SNs are limited by the resources due to limited computational power and low battery supply; thus, energy saving technologies must be considered when we design the protocols. For better energy utilization, cluster-based WSNs have been proposed. In CDAMA, cluster-based WSN is used. In cluster-based WSNs, SN resident in nearby area would form a cluster and select one among them to be their cluster head (CH). The CH organizes data pieces received from SN into an aggregated result, and then forwards the result to the base station based on regular routing paths. Generally, aggregative operations are algebraic, such as the addition or multiplication of received data, or statistical operation, such as a median, a minimum, or a maximum of a data set. In CDAMA, the collected data from SN will be encrypted and the encrypted cipher text data will be transferred to the CH. CDAMA utilizes the privacy homomorphism encryption (PH) to facilitate aggregation in encrypted data. CHs can execute algebraic aggregation operations on encrypted numeric data. The aggregated data will be sent to the base station (BS). The BS will decrypt the aggregated data.

II. RELATED WORKS

In the paper Recoverable Concealed data Aggregation for data integrity in Wireless Sensor Network [2], RCDA schemes are proposed for two types of WSN i.e. homogeneous and heterogeneous WSN. Special feature of this scheme is that the base station can securely recover all sensing data generated by the sensor nodes rather than aggregated results with less transmission overhead. In addition, to ensure the authenticity and integrity, the aggregate signature scheme is integrated. Although integration of signature brings additional cost but still it is affordable for WSN.

In the paper Public Key Based Crypto schemes for Data Concealment in Wireless Sensor Network [3], Mykletun proposed various Public key Encryption Schemes with the comparison of their costs as well as indication of how practically they can be implemented. He worked on a concealed data aggregation scheme based on elliptic curve Elgamal (EC-EG) cryptosystem. Symbols that are used in this scheme are + and x, where + denote addition and x

denote scalar multiplication on elliptic curve points. Four procedures are there in this scheme: a) Key Generation: Generate a key pair of private and public key. b) Encryption: Encrypt message with public key. c) Aggregation: A few aggregation functions are listed which can be computed over enciphered data and recommended which cryptosystem should be used in which application. d) Decryption: Decrypt cipher text with private key [4]. He showed that their indeed exists a viable public key cryptosystem candidate for WSNs. This scheme has two applications, Aggregation and Long term data storage. The former involve functions like sum, average, variance, checksum and movement detection. The latter application relies on the fact that data is stored in the nodes for later retrieval when needed. Due to limited storage capability, the amount of values is reduced. Therefore, we can use the concept of Data aggregation to reduce the amount of data stored at the nodes.

Symbols that are used in this scheme are + and x, where + denote addition and x denote scalar multiplication on elliptic curve points. Four procedures are there in this scheme: a) Key Generation: Generate a key pair of private and public key. b) Encryption: Encrypt message with public key. c) Aggregation: A few aggregation functions are listed which can be computed over enciphered data and recommended which cryptosystem should be used in which application. d) Decryption: Decrypt cipher text with private key.

In the paper CDA: Concealed Data Aggregation in Wireless Sensor Networks [4], tiny and cheap cost sensors consist of application-specific sensors, a wireless transceiver, simple processor and a battery. Problem of end-to-end encryption of data is introduced. An encryption transformation known as privacy homomorphism that allows encrypted data to be computed without decrypting it. Let P and C denote Plaintexts and Cipher texts respectively. Let K be the key space.

Encryption transformation: E: K x P→C

Decryption transformation: D: K x C→P

PH can be performed additively and multiplicatively.

Additive Homomorphism:

a+b = Dk(Ek(a)+ Ek(b))

Multiplicatively Homomorphism:

a x b = Dk(Ek(a) x Ek(b)). RSA is a multiplicative PH.

But disadvantages of these schemes are

1. They do not satisfy multi-application environments
2. They become insecure in case some sensor nodes are compromised
3. They do not provide secure counting; thus, they may suffer unauthorized aggregation attacks

III. CDAMA

A new Concealed Data Aggregation scheme for Multiple Applications (CDAMA) extended homomorphic public encryption system was proposed in the paper CDAMA: Concealed Data Aggregation Scheme for Multiple Applications in Wireless Sensor Networks [1]. The proposed scheme has three contributions.

The first scenario is designed for multi-application WSNs. In practice, sensor nodes having different purposes, e.g., smoke alarms and thermometer sensors may be deployed in the same environment. If we apply conventional concealed data aggregation schemes, the cipher texts of different applications cannot be aggregated together; otherwise the decrypted aggregated result will be incorrect. The only solution is to aggregate the cipher texts of different applications separately. As a result, the transmission cost grows as the number of the applications increases. By CDAMA, the cipher texts from different applications can be encapsulated into "only" one cipher text. Conversely, the base station can extract application-specific plaintexts via the corresponding secret keys.

The second scenario is designed for single application WSNs. Compared with conventional schemes; CDAMA mitigates the impact of compromising sensor nodes through the construction of multiple groups. An adversary can forge data only in the compromised groups, not the whole system.

The last scenario is designed for secure counting capability. In previous schemes, the base station does not know how many messages are aggregated from the decrypted aggregated result; leaking count knowledge will suffer maliciously selective aggregation and repeated aggregation. In CDAMA, the base station exactly knows the number of messages aggregated to avoid above attacks.

Advantages:

1. The Proposed system CDAMA satisfies the Multiple Application Environments
2. CDAMA mitigates the impact of compromising sensor nodes. An adversary can forge data only in the compromised groups, not the whole system.
3. The base station exactly knows the number of messages aggregated to avoid above attacks.

Disadvantages:

1. Less efficiency
2. Less security

IV. TCMSDA

In the proposed system, in order to diminish the computation complexity an innovative technique is introduced which is called a trust computation model for secure data aggregation (TCMSDA) in wireless sensor networks. To differentiate false data from legal ones is an essential process for a normal and effective function of sensor networks, because false reports can drain out the finite amount of energy resources in a battery-powered sensor networks, and even a small amount of compromised nodes can influence the whole sensor networks critically. This method helps the networks to operate normally with high probability, although some nodes or data would be compromised. General direction for resilience is to gather multiple and redundant sensing data and cross check them for consistency. For reasonable cross checking, they are compared with the expected sensing results within the possible and legitimate sensing range. Based on the result of that cross checking, each node estimates its neighbour nodes' trust

values. As the sensor nodes operate trustworthily, they will get higher trust values from its neighbour nodes. On the other hand, of course, as the sensor nodes operate maliciously or inconsistently, they will get lower trust values. In the proposed scheme, each sensor node has a trust value which is based on the trust evaluation factors, such as identification, sensing data and consistency. Based on these factors, one can identify malicious or compromised nodes, and filter their data from the networks. So, by using this method high efficiency and security is achieved.

Advantages:

- More efficient
- High security

A. Architecture Diagram

The architecture diagram of the TCMSDA is shown in the Fig. 1.

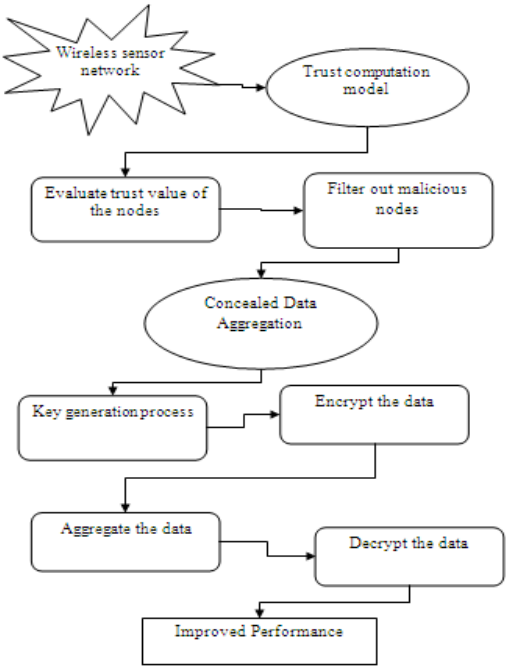

Fig. 1 Architecture diagram of TCMSDA

B. Modules

List of Modules

- Creation of Network
- Trust Computational Model
- BGN Scheme
- CDAMA (k = 2) Construction
- Generalization of CDAMA
- Key Distribution
- Performance Evaluation

1)Creation of Network: An undirected graph G (V, E) where the set of vertices V represent the mobile nodes in the network and E represents set of edges in the graph which represents the physical or logical links between the mobile nodes. Sensor nodes are placed at a same level. Two nodes that can communicate directly with each other are connected by an edge in the graph. Let N denote a network of m mobile nodes, $N_1, N_2, \dots N_m$ and let D denote a collection of n data items $d_1, d_2, \dots d_n$ distributed in the network. For each pair of mobile nodes N_i and N_j, let t_{ij} denote the delay of transmitting a data item of unit-size between these two nodes.

2) Trust Computational Model: In this step, a trust evaluation process is proposed. The trust defined in our model is the confidence of a node on another node. The trust value means the level of trustworthiness of a node, which is computed based upon several trust evaluation factors, such as battery lifetime, sensing communication ratio, sensing result, and consistency level. As a node communicates and revalues trust factor values for their neighbour nodes continuously, trust quantification process is imperative to impartial comparison among each node's trust values. Trust quantification processes for each trust evaluation factor are as follows:

Consistency value: $C_i = \frac{cs_i - ls_i}{cs_i + ls_i}$, where $-1 \leq C_i \leq 1$.

Sensing communication value: $S_i = \frac{ss_i - sf_i}{s + sf_i}$, where $-1 \leq S_i \leq 1$.

Battery value: B_i: $-1 \leq B_i \leq 1$

where each sensor node broadcasts quantification value of its own B_i.

Trust computation involves an assignment of weights to the trust factors that are evaluated and quantified in trust quantification step. W_i is defined as a weight which represents importance of a particular factor from 0-unimportant to +1- most important. The weight is dynamic and dependent on the application. Trust value for node *i* is computed by the following equation:

If $B_i \neq -1$, $T_i = \frac{W_1 C_i + W_2 S_i + W_3 B_i}{\sum_{k=1}^{3} W_i}$

where $0 \leq W_i \leq 1$. In case of $B_i \neq -1$, -1 to T_i is assigned and exclude the node from the networks because it totally cannot work in the networks. As the time elapses, trust values for neighbour nodes change dynamically and continuously. If a node makes some trivial and contemporary mistakes in communication or sensing events, such mistakes have little influence on the trust value which is evaluated by its neighbour nodes. It is because each sensor node uses histograms for the accumulative trust evaluation, which are implemented as several count factors in the trust evaluation matrix. Else if a node broadcasts inconsistent data steadily or seldom communicate with its neighbour nodes, trust value for that node is decreasing and convergent to -1. Therefore, some malicious or compromised nodes that broadcast inconsistent or deceitful data continuously can be detected and classified in this step.

3) BGN Scheme: BGN provides additive and multiplicative homomorphism. BGN is constructed on a cyclic group of elliptic curve points. Precisely, these points form an algebraic group, where the identity element of the group is the infinite point ∞. ord (P) denotes the order of a point P. Supposing ord (P) = q, it indicates that q is the minimum integer that satisfies q *P = ∞. In the KEYGEN function, the order of E is equivalent to the number of points in E. The ENC function is based on point addition

and scalar multiplication over points $\mathcal{G}$ and $\mathcal{H}$. Due to homomorphic properties, the AGG function aggregates ciphertexts via point addition; it is trivial to see that the scalar values of point G were added in the end, yielding the sum of the corresponding message. Consequently, the final result will be the form of M* $\mathcal{G}$ + R* $\mathcal{H}$ .where M is the sum of the messages and R is the sum of the randomness. The DEC function decrypts the aggregated result to obtain the plaintext value, M. Recall that the order of points $\mathcal{G}$ and $\mathcal{H}$ are different. Hence, the DEC function removes the randomness of point $\mathcal{H}$ by multiplying the result with the private key. Now, the cipher text contains only the product of G such that the discrete logarithm can be applied to retrieve the value M.

KEYGEN(τ): generate a public-private key pair
1. Based on security parameter τ, it computes a triple elements, (q_1, q_2, E)
   where E is a set of elliptic curve points which form a cyclic group.
   The order of E, ord(E), is n where n equals to the product of q_1 and q_2; q_1 and q_2 are large primes.
2. Randomly select two generators (i.e., base points) $\mathcal{G}$, $\mathcal{U}$, where ord($\mathcal{G}$) = ord($\mathcal{U}$) = n.
3. Compute point $\mathcal{H} = q_2 * \mathcal{U}$ such that ord($\mathcal{H}$) = q_1.
4. Select parameter T as the maximum plaintext boundary and $T < q_2$.
5. Output the public key: $PK = (n, E, \mathcal{G}, \mathcal{H}, T)$.
6. Output the private key: $SK = q_1$.

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

AGG(C_1, C_2): Aggregation on two ciphertexts C_1, C_2.
where $C_1 = M_1 * \mathcal{G} + R_1 * \mathcal{H}$ and $C_2 = M_2 * \mathcal{G} + R_2 * \mathcal{H}$.
1. Randomly select $R' \in \{0, \cdots, n-1\}$.
2. Compute the aggregated ciphertext of $(m_1 + m_2)$, C' as:
   $C' = C_1 + C_2 + R' * \mathcal{H} = (M_1 + M_2) * \mathcal{G} + (R_1 + R_2 + R') * \mathcal{H}$.
3. Output C'.

DEC(SK, C): Message decryption on C by private key SK
1. Compute $\log_{\tilde{\mathcal{G}}}(q_1 * C) = \log_{\tilde{\mathcal{G}}}(q_1 * (M * \mathcal{G} + R * \mathcal{H})) = \log_{\tilde{\mathcal{G}}}(M * q_1 * \mathcal{G}) = M$ where $\tilde{\mathcal{G}} = q_1 * \mathcal{G}$.
2. Output M.

Fig. 2 BGN Scheme

4) CDAMA (k = 2) Construction: Assume that all SNs are divided into two groups, G_A and G_B. CDAMA contains four procedures: Key generation, encryption, aggregation, and decryption. CDAMA (k = 2) is implemented by using three points $\mathcal{P}$, $\mathcal{Q}$ and $\mathcal{H}$ whose orders are q_1, q_2 and q_3 respectively. The scalars of the first two points carry the aggregated messages in G_A and G_B , 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 $q_2 q_3$, the scalar of the point P carrying the aggregated message in G_A can be obtained. Similarly, by multiplying the aggregated ciphertext with $q_1 q_3$, the scalar of the point Q carrying the aggregated message in G_B can be obtained. In this way, the encryptions of messages of two groups can be aggregated to a single cipher text, but the aggregated message of each group can be obtained by decrypting the cipher text with the corresponding Secret Key (SK).

KEYGEN(τ): generate public-private key pairs for group G_A and G_B
1. Based on security parameter τ, compute (q_1, q_2, q_3, E), where
   E is the set of elliptic curve points which form a cyclic group;
   ord(E) = n, and $n = q_1 q_2 q_3$; q_1, q_2, q_3 are large primes;
   the bit lengths of q_1, q_2, and q_3 are the same, i.e., $|q_1| = |q_2| = |q_3|$.
2. Randomly pick up three generators, $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ such that ord($\mathcal{G}_1$) = ord($\mathcal{G}_2$) = ord($\mathcal{G}_3$) = n.
3. Compute point $\mathcal{H} = q_1 q_2 * \mathcal{G}_3$; ord($\mathcal{H}$) = q_3.
4. Select parameter T as the maximum plaintext boundary where Pollard's λ method is feasible;
   then compute $T_A = T_B = \lfloor \frac{T}{x} \rfloor$ where x is the number of sensors in an application.
5. Compute $\mathcal{P} = q_2 q_3 * \mathcal{G}_1$, ord($\mathcal{P}$) = q_1;
   then output G_A's group public key PK_A: $PK_A = (n, E, \mathcal{P}, \mathcal{H}, T_A)$.
6. Compute $\mathcal{Q} = q_1 q_3 * \mathcal{G}_2$, ord($\mathcal{Q}$) = q_2;
   then output G_B's group public key PK_B: $PK_B = (n, E, \mathcal{Q}, \mathcal{H}, T_B)$.
7. Output G_A's group Private key SK_A as $(q_2 q_3)$, and G_B's group Private key SK_B as $(q_1 q_3)$.

ENC(PK_A, M): Message encryption in G_A
1. Check if message $M \in \{0, \cdots, T_A\}$.
2. Randomly select $R \in \{0, \cdots, n-1\}$.
3. Generate the resulting ciphertext C as: $C = M * \mathcal{P} + R * \mathcal{H}$.
4. Return C.

ENC(PK_B, M): Message encryption in G_B
1. Check if message $M \in \{0, \cdots, T_B\}$.
2. Randomly select $R \in \{0, \cdots, n-1\}$.
3. Generate the resulting ciphertext C as: $C = M * \mathcal{Q} + R * \mathcal{H}$.
4. Return C.

AGG(C_1, C_2): Message aggregation on two ciphertexts C_1 and C_2
1. Compute the aggregated ciphertext
   $C' = C_1 + C_2$; $C' = (\sum M_i) * \mathcal{P} + (\sum M_j) * \mathcal{Q} + (\sum R_i) * \mathcal{H}$, where
   $\sum M_i$ represents the aggregated result of G_A,
   $\sum M_j$ represents the aggregated result of G_B,
   and $\sum R_i$ represents the aggregated randomness of both groups.
2. Return C'.

DEC(SK_A, C): Message decryption on C for group G_A
1. Compute $M = \sum M_i = \log_{\tilde{\mathcal{P}}}(q_2 q_3 * C)$ where $\tilde{\mathcal{P}} = q_2 q_3 * \mathcal{P}$
2. Return M.

DEC(SK_B, C): Message decryption on C for group G_B
1. Compute $M = \sum M_j = \log_{\tilde{\mathcal{Q}}}(q_1 q_3 * C)$ where $\tilde{\mathcal{Q}} = q_1 q_3 * \mathcal{Q}$.
2. Return M.

Fig. 3 Procedures of CDAMA (k = 2)

5) Generalization of CDAMA: CDAMA (k = 2) can be generalized to CDAMA (k > 2). The paradigm of generalization uses different generators to construct different key pairs for groups. For security reasons, the order of E should be large enough. Therefore, when k becomes large, the length of cipher text will also expand. For multi-application WSNs, the SNs belonging to one specific application are assigned the same group public key. Under CDAMA, the cipher texts from different applications can be aggregated together, but they are not mixed. The cipher texts can be integrated into a cipher text and transmitted to the BS. The BS then individually decrypts the aggregated cipher text to extract the aggregated value of each application.

KEYGEN(τ): generate public-private key pairs for group G_i, $\forall i = 1 \sim k$
1. Based on security parameter τ, compute elements, $(q_1, q_2, \cdots, q_{k+1}, E)$,
   where E is the set of elliptic curve points which form a cyclic group;
   ord(E) = n; n is the product of $q_1, \cdots, q_{k+1}$ and $q_1, \cdots, q_{k+1}$ are large primes;
   the bit length of q is the same, i.e., $|q_1| = \cdots = |q_k| = |q_{k+1}|$.
2. Randomly pick up $k+1$ generators, $\mathcal{G}_1, \cdots, \mathcal{G}_{k+1} \in E$ where ord($\mathcal{G}_i$) = n, $\forall i$.
3. Compute point $\mathcal{H} = (\prod_{i=1}^{k} q_i) * \mathcal{G}_{k+1}$ such that ord($\mathcal{H}$) = q_{k+1}.
4. Let T be the maximum plaintext boundry where Pollard's λ method is feasible,
   and let $T_i = \lfloor \frac{T}{x} \rfloor$, $i = 1 \sim k$, where x is the average number of sensors in an application.
5. Compute point $\mathcal{P}_i = (\prod_{i=1, i \neq i}^{k+1} q_i) * \mathcal{G}_i$ such that ord($\mathcal{P}_i$) = q_i for $i = 1, \cdots, k$.
6. Output G_i's group public key (PK_i): $PK_i = (n, E, \mathcal{P}_i, \mathcal{H}, T_i)$.
7. Output the private key = $SK_i = (q_1, q_2, \cdots, q_{k+1})$.

ENC(PK_i, M): Message encryption in G_i
1. Check if message $M \in \{0, \cdots, T_i\}$.
2. Randomly select $R \in \{0, \cdots, n-1\}$.
3. Generate the ciphertext C as: $C = M * \mathcal{P}_i + R * \mathcal{H}$ where $\mathcal{P}_i \in PK_i$.
4. Return C.

AGG(C_1, C_2): Message aggregation on two ciphertexts C_1 and C_2
1. Aggregated ciphertext $C' = C_1 + C_2 = \sum_{i=1}^{k} (\sum M_i) * \mathcal{P}_i + (\sum R_i) * \mathcal{H}$,
   where $\sum M_i$ represents the aggregated result of group G_i
   and $\sum R_i$ presents the aggregated randomness of all groups.
2. Return C'.

DEC(SK_i, C): Message decryption on C for group G_i using private key SK_i
1. Compute $M = \sum M_i = \log_{\tilde{\mathcal{G}}_i}((\prod_{i=1, i \neq i}^{k+1} q_i) * C)$ where $\tilde{\mathcal{G}}_i = (\prod_{i=1, i \neq i}^{k+1} q_i) * \mathcal{P}_i$.
2. Return M.

Fig. 4 Procedures of Generalization of CDAMA

6) Key Distribution:

Key Pre-distribution

If the locations of deployed SNs are known, necessary keys and functions can be preloaded into SNs and AGs so that they can work correctly after being spread out over a geographical region.

Key Post-distribution

Before SNs are deployed to their geographical region, they are capable of nothing about CDAMA keys. These SNs only load the key shared with the BS prior to their deployment. Once these SNs are deployed, they can run the LEACH protocol to elect the AGs and construct clusters. After that, the BS sends the corresponding CDAMA keys, encrypted by the pre-shared key, to SNs and AGs.

7) Performance Evaluation: In this section, the performance of the existing and the proposed system is compared using NS2 Visual Trace Analyser. In the existing system, concealed data aggregation between multiple groups (CDAMA) is used. In the proposed system, Trust Computation Model for Secure Data Aggregation (TCMSDA) in wireless sensor networks is used to find the malicious nodes. A new trust management scheme is essential to differentiate illegal and normal nodes and filter out the malicious nodes in the network. In the trust computation model, each node identifies trustworthiness of sensor nodes. When compared to the existing method there is high security and high performance in the proposed system.

V. Experiments And Results

A. Working Environment

The experiment has been carried out by implementing CDAMA and TCMSDA data aggregation schemes. CDAMA and TCMSDA data aggregation schemes are implemented using NS-2 and its trace files are analyzed using NS2 Visual Trace Analyzer. The results of the experiment are compared using Packet Delivery Ratio and Packets Delay.

B. Network Setup

For the performance analysis, CDAMA and TCMSDA are compared. CDAMA scheme, which provides CDA between multiple groups and TCMSDA scheme are simulated in the NS-2 with the following network setup.

1) *Simulation Parameters:*

Simulator	:	NS-2
Topology size	:	1000m X 1000m
Number of nodes	:	31 nodes
Transmission range	:	250m
Bandwidth	:	2 Mbps

C. Performance Metrics

The performance of the CDAMA and TCMSDA schemes are evaluated and analysed based on the metrics Packet Loss, Packet Delivery Ratio, Throughput Transferred and Lifetime in NS-2 Visual Trace Analyzer.

1) Packet Loss: Packet Loss is defined as number of packets of dropped during the transmission.

2) *Packet Delivery Ratio:* Packet delivery ratio is defined as the ratio of data packets received by the destinations to those generated by the sources. Mathematically, it can be defined as:

PDR= S1÷ S2

Where, S1 is the sum of data packets received by the each destination and S2 is the sum of data packets generated by the each source.

3) Throughput Transferred: Throughput Transferred is defined as the number of data bytes transferred per second.

4) Lifetime: Lifetime of the network is defined as the difference between time at which the first packet generated and the time at which the last packet transferred.

D. Results

1) Experiment for CDAMA Data Aggregation Scheme:

The CDAMA scheme is implemented in NS-2. The performance of the CDAMA scheme is evaluated and analysed using NS-2 Visual Trace Analyser. Packet Loss, Packet Delivery Ratio, Throughput Transferred and Lifetime are used as performance metrics for the CDAMA scheme.

The simulation of CDAMA scheme in NS-2 is shown in Fig. 5. Performance of the CDAMA scheme is summarized in Table I.

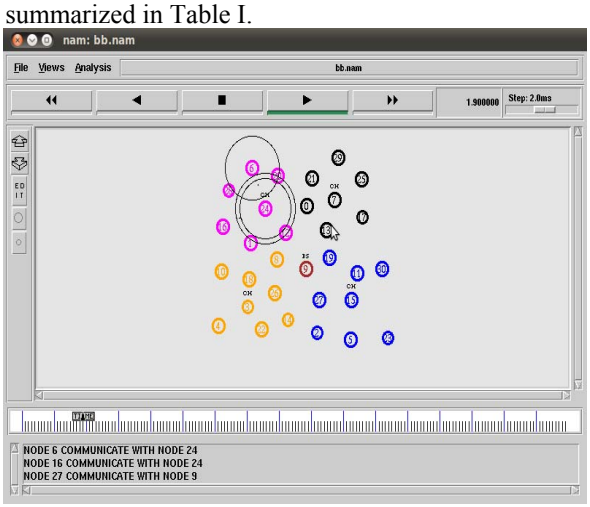

Fig. 5 Simulation of CDAMA Scheme in NS-2

TABLE I

PERFORMANCE OF CDAMA

Data Aggregation Scheme	Packet Loss (packets)	Packet Delivery Ratio	Throughput Transferred (KB/S)	Lifetime (seconds)
CDAMA	568	0.72	88	15.93

2) Experiment for TCMSDA Data Aggregation Scheme:

The TCMSDA scheme is implemented in NS-2. The performance of the TCMSDA scheme is evaluated and analysed using NS-2 Visual Trace Analyser. Packet Loss, Packet Delivery Ratio, Throughput Transferred and

Lifetime are used as performance metrics for the TCMSDA scheme.

The simulation of TCMSDA scheme in NS-2 is shown in Fig 6. Performance of the TCMSDA scheme is summarized in Table II.

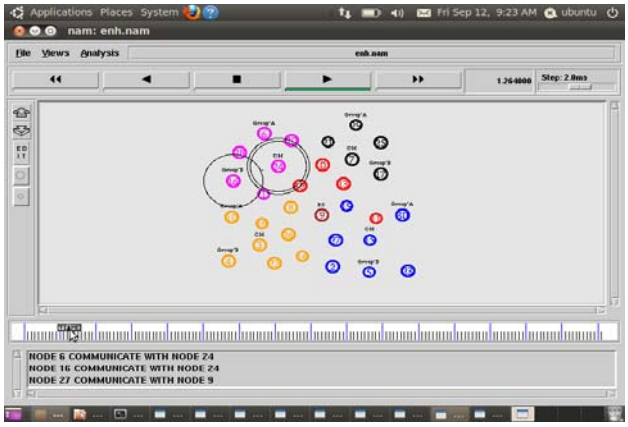

Fig. 6 Simulation of TCMSDA Scheme in NS-2

TABLE II

PERFORMANCE OF TCMSDA

Data Aggregation Scheme	Packet Loss (packets)	Packet Delivery Ratio	Throughput Transferred (KB/S)	Lifetime (seconds)
TCMSDA	537	0.73	90	16.70

3) *Comparative Results of the CDAMA and TCMSDA*:

Comparative results of two experiments carried out by implementing the CDAMA and TCMSDA schemes are summarized in Table II.

TABLE III

COMPARATIVE RESULTS OF CDAMA AND TCMSDA

Data Aggregation Scheme	Packet Loss (packets)	Packet Delivery Ratio	Throughput Transferred (KB/S)	Lifetime (seconds)
CDAMA	568	0.72	88	15.93
TCMSDA	537	0.73	90	16.70

E. Comparative Charts

The comparison charts for the above performance metrics for CDAMA and TCMSDA are shown below.

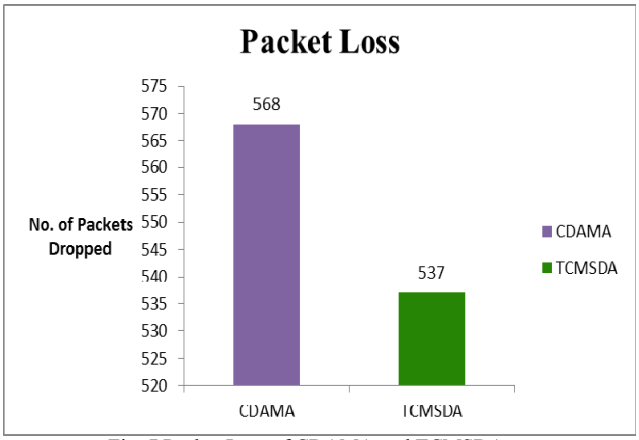

Fig. 7 Packet Loss of CDAMA and TCMSDA

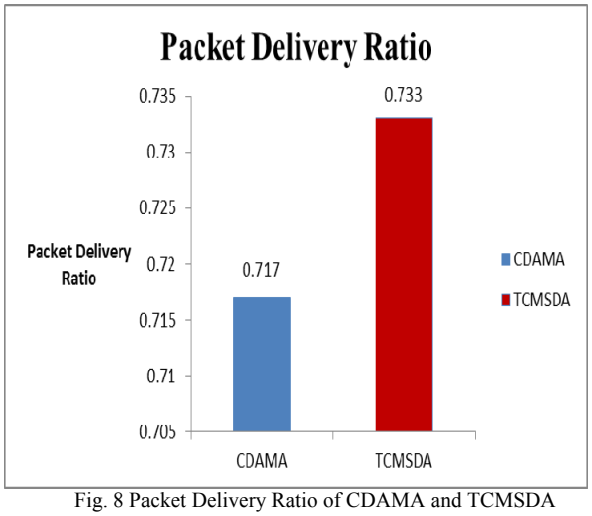

Fig. 8 Packet Delivery Ratio of CDAMA and TCMSDA

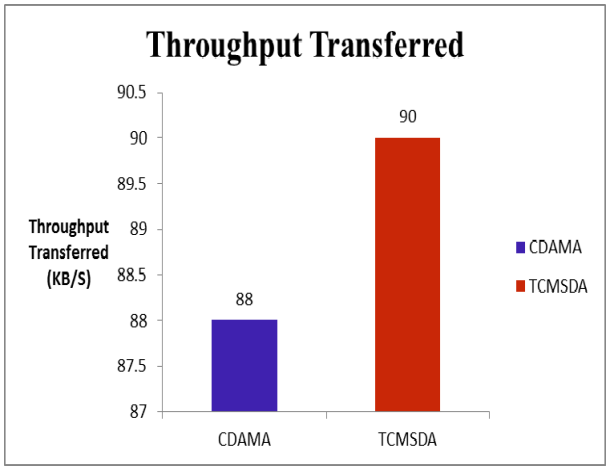

Fig. 9 Throughput Transferred of CDAMA and TCMSDA

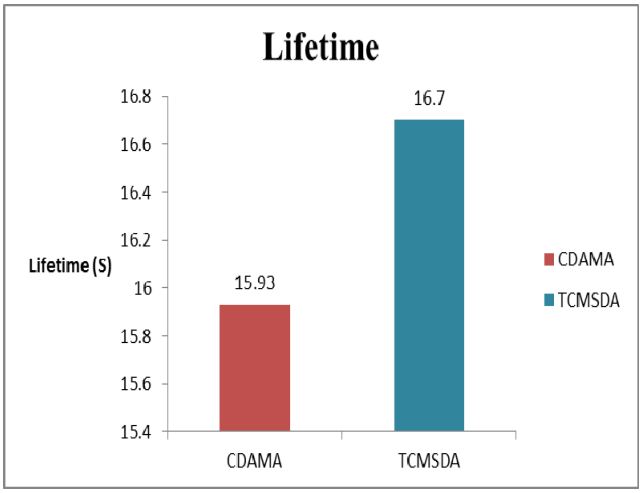

Fig. 10 Comparison of Lifetime of CDAMA and TCMSDA

It is observed from the results that the performance of the TCMSDA scheme based on trust computation is more efficient and secure compared to CDAMA data aggregation scheme.

VI. CONCLUSION

In the wireless sensor networks, for a multi-application environment, CDAMA is the first CDA scheme. By using CDAMA, the cipher texts from distinct applications can be aggregated, but not mixed. For a single-application environment, CDAMA is still more secure than other CDA schemes. When compromising attacks occur in WSNs, CDAMA mitigates the impact and reduces the damage to an acceptable condition. Besides the above applications, CDAMA is the first CDA scheme that supports secure counting. The base station would know the exact number of messages aggregated, making selective or repeated aggregation attacks infeasible. But it is more complex and less efficient. So Trust Computation Model for Secure Data Aggregation (TCMSDA) scheme is introduced for wireless sensor networks. In this method, the trust value of the node is computed. Based on this, illegal and normal nodes are differentiated and the malicious nodes are filtered out in the network. In the trust computation model, each node identifies trustworthiness of sensor nodes.

In future, this model can be extended as Hierarchical Concealed Data Aggregation method which allows concealed aggregation of data that are encrypted with different keys. This method will virtually partition the network into several regions and will employ a different public key in each region. Due to the privacy homomorphic encryption scheme of this method, the data collected in a region can be encrypted using the public key of the region and the encrypted data of several regions can be hierarchically aggregated into a single piece of data without violating data confidentiality.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," IEEE Comm. Magazine, vol. 40, no. 8, pp. 102-114, Aug. 2002.

[2] R. Min and A. Chandrakasan, "Energy-Efficient Communication for Ad-Hoc Wireless Sensor Networks," Proc. Conf. Record of the 35th Asilomar Conf. Signals, Systems and Computers, vol. 1, 2001.

[3] B. Przydatek, D. Song, and A. Perrig, "SIA: Secure Information Aggregation in Sensor Networks," Proc. First Int'l Conf. Embedded Networked Sensor Systems, pp. 255-265, 2003.

[4] A. Perrig, J. Stankovic, and D. Wagner, "Security in Wireless Sensor Networks," Comm. ACM, vol. 47, no. 6, pp. 53-57, June 2004.

[5] L. Hu and D. Evans, "Secure Aggregation for Wireless Networks," Proc. Symp. Applications and the Internet Workshops, pp. 384-391, 2003.

[6] H. Cam, S. O ¨ zdemir, P. Nair, D. Muthuavinashiappan, and H.O. Sanli, "Energy-Efficient Secure Pattern Based Data Aggregation for Wireless Sensor Networks," Computer Comm., vol. 29, no. 4, pp. 446-455, 2006.

[7] H. Sanli, S. Ozdemir, and H. Cam, "SRDA: Secure Reference-based Data Aggregation Protocol for Wireless Sensor Networks," Proc. IEEE 60th Vehicular Technology Conf. (VTC '04-Fall), vol. 7, 2004.

[8] Y. Wu, D. Ma, T. Li, and R.H. Deng, "Classify Encrypted Data in Wireless Sensor Networks," Proc. IEEE 60th Vehicular Technology Conf., pp. 3236-3239, 2004.

[9] D. Westhoff, J. Girao, and M. Acharya, "Concealed Data Aggregation for Reverse Multicast Traffic in Sensor Networks: Encryption, Key Distribution, and Routing Adaptation," IEEE Trans. Mobile Computing, vol. 5, no. 10, pp. 1417-1431, Oct. 2006.

[10] J. Girao, D. Westhoff, M. Schneider, N. Ltd, and G. Heidelberg, "CDA: Concealed Data Aggregation for Reverse Multicast Traffic in Wireless Sensor Networks," Proc. IEEE Int'l Conf. Comm. (ICC '05), vol. 5, 2005.

[11] E. Mykletun, J. Girao, and D. Westhoff, "Public Key Based Cryptoschemes for Data Concealment in Wireless Sensor Networks," Proc. IEEE Int'l Conf. Comm. (ICC '06), vol. 5, 2006.

[12] J. Girao, D. Westhoff, E. Mykletun, and T. Araki, "Tinypeds: Tiny Persistent Encrypted Data Storage in Asynchronous Wireless Sensor Networks," Ad Hoc Networks, vol. 5, no. 7, pp. 1073-1089, 2007.

[13] D. Boneh, E. Goh, and K. Nissim, "Evaluating 2-DNF Formulas on Ciphertexts," Proc. Second Int'l Conf. Theory of Cryptography (TCC), vol. 3378, pp. 325-341, 2005.

[14] C. Castelluccia, E. Mykletun, and G. Tsudik, "Efficient Aggregation of Encrypted Data in Wireless Sensor Networks," Proc. Second Ann. Int'l Conf. Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous '05), pp. 109-117, 2005.

[15] S. Peter, D. Westhoff, and C. Castelluccia, "A Survey on the Encryption of Convergecast-Traffic with In-Network Processing," IEEE Trans. Dependable and Secure Computing, vol. 7, no. 1, pp. 20- 34, Jan.-Mar. 2010.

[16] R. Cramer and V. Shoup, "A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attack," Proc. 18th Ann. Int'l Cryptology Conf. Advances in Cryptology, pp. 13-25, 1998.

[17] J. Domingo-Ferrer, "A Provably Secure Additive and Multiplicative Privacy Homomorphism," Proc. Fifth Int'l Conf. Information Security, pp. 471-483, 2002.

[18] N. Koblitz, A. Menezes, and S. Vanstone, "The State of Elliptic Curve Cryptography," Designs, Codes and Cryptography, vol. 19, no. 2, pp. 173-193, 2000.

[19] P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes," Proc. 17th Int'l Conf. Theory and Application of Cryptographic Techniques, pp. 223-238, 1999.

[20] T. Okamoto and S. Uchiyama, "A New Public-Key Cryptosystem as Secure as Factoring," Proc. Int'l Conf. Theory and Application of Cryptographic Techniques, pp. 308-318, 1998.

[21] L. Oliveira, D. Aranha, E. Morais, F. Daguano, J. Lopez, and R. Dahab, "TinyTate: Computing the Tate Pairing in Resource-Constrained Sensor Nodes," Proc. IEEE Sixth Int'l Symp. Network Computing and Applications (NCA '07), pp. 318-323, 2007.

[22] L. Washington, Elliptic Curves: Number Theory and Cryptography. Chapman & Hall/CRC, 2008.

[23] S. Zhu, S. Setia, and S. Jajodia, "LEAP+: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks," ACM Trans. Sensor Networks, vol. 2, no. 4, pp. 500-528, 2006.

[24] A. Perrig, R. Szewczyk, J. Tygar, V. Wen, and D. Culler, "SPINS: Security Protocols for Sensor Networks," Wireless Networks, vol. 8, no. 5, pp. 521-534, 2002.

[25] S. Bhattacharya, A. Saifullah, C. Lu, and G. Roman, "Multi-Application Deployment in Shared Sensor Networks Based on Quality of Monitoring," Proc. IEEE 16th Real-Time and Embedded Technology and Applications Symp., pp. 259-268, 2010.