International Journal of Latest Trends in Engineering and Technology
Vol.(7)Issue(3), pp. 343-353
DOI: http://dx.doi.org/10.21172/1.73.546
e-ISSN:2278-621X

Design and Implementation of Secure Data Aggregation in Wireless Sensor Networks

Dr. S. Siva Ranjani[1] and M.M. Yamuna Devi[2]

Abstract- Wireless sensor network (WSN) is an idea of sensing and controlling any remote area or system using sensor nodes. Because of the open deployment, sensors are vulnerable for security threats. In this paper we address the energy and security issues together. In our approach, we modify our Energy efficient Cluster Based Data Aggregation (ECBDA)[1] scheme to provide secure data transmission. The survival of any sensor node depends on its residual battery power which is having a direct impact on the communication between nodes. Since, sensors nodes are low powered in nature, it is not viable to apply standard cryptography methods. Cluster head performs data aggregation and Bayesian fusion algorithm to enable security. Trust is the directional relationship between two sensor nodes. By checking the trustworthiness of a node, we can enable secure communication. Bayesian fusion algorithm calculates the trust probability of a sensor based on the behavior of the node. The simulation results show that our approach effectively detects the untrustworthy nodes with minimum energy consumption.

Keywords – Aggregation; Clustering; Security; Bayes' Theorem

I. INTRODUCTION

Wireless sensor networks have thousands of inexpensive, low-powered sensing devices with limited resources. These networks are used in both military and civilian applications. Limited use of the battery power is the key challenge in all types of remote sensing applications. In most of the real time applications, sensor nodes are deployed in the open and remote environments. While data aggregation can assure energy conservation, authenticity and integrity are not guaranteed. As such, wireless sensor networks are highly vulnerable to security threats; methods are needed for assuring the secure transmission with minimal energy requirements.

Numerous cryptographic algorithms are proposed by several researchers to control malicious behavior in the wireless medium. But each algorithm needs the complex hardware resources to execute it. Because of the limited resources in sensor devices, it is impossible to use conventional cryptographic algorithms to ensure security. Key management and cryptographic algorithms lead to high computation cost in the individual sensors. Key exchange and new key generation lead to communication overhead, and hence it increases energy consumption in the overall network.

[1] *Department of Information Technology Sethu Institute of Technology, Virudhu Nagar, Tamil Nadu, India*
[2] *Department of Information Technology Sethu Institute of Technology, Virudhu Nagar, Tamil Nadu, India*

A. *Security threats*

In WSN, security threat is a harmful event performed by any vulnerable attackers in the network. Since WSNs are applied in several real time applications, secure communication is the essential issue. According to Du and Chen [2] and Giruka et al. [3], attacks in WSN can be grouped into three types: 1) Attack on authentication, 2) Attack on availability and 3) Attack on Integrity. Authentication and confidentiality of received information are the crucial parts in any network. In WSN, the receiving sensor node has to ensure that the data is initiated from the trusted source node. Authentication allows sender node and receiver node to be sure that they are talking with those with whom they really want to communicate. Attacks on authentication can be performed by capturing an existing node in the field or injecting a new malicious node in the field for spoofing of packets or to generate false information and rendering the receiver to believe the malicious node as the real source.

Purposefully making the sensor node resources or the communication medium busy is known as the attack on availability. It is generally called the Denial of Service (DoS) attacks. Attacker node can generate a jam signal or flooding the traffic to make network resources unavailable. DoS attacks will totally degrade the network functionally. Any real sensor node cannot report the sensed information to the sink node. So, the overall purpose of the application is distorted when any DoS attacks happen in the network. Attack on integrity is the action performed by the attacker node to capture and modify the contents of any forwarding packet. Hence the receiver gets wrong information and reacts wrongly. If the sink node receives the modified information from any attacker in the network, then the decision will be wrong and it will lead to many dangerous issues.

B. *Trust models*

Trust is the directional relationship between two sensor nodes. As proposed by Lopez et al. [4], secure communication can be assured by checking the trustworthiness of a node. Computation cost is low for finding the trust of a node when compared to the regular cryptographic methods. Bayesian fusion algorithm is used to measure the trust of a node. It tells the probability of the trustworthiness of a sensor. Shaikh, R et al. [5] proposed a new lightweight group-based trust management scheme (GTMS) for wireless sensor networks, which employs clustering. This approach reduces the cost of trust evaluation. A set of best practices about a node is included in the design of trust management system so the success rate is increased. Also, theoretical as well as simulation results show that this scheme demands less memory, energy, and communication overheads as compared to the current state-of-the-art trust management schemes, and it is more suitable for large-scale sensor networks.

As given by Dhulipala et al. [6], trust is defined as the belief or confidence about the nodes based on their past interactions with the neighbors. The authors proposed a Heuristic Approach-based Trust Worthy Architecture for WSN which considers the challenges of the system and focuses on the collaborative mechanism for trust evaluation and maintenance. They use a high power network monitoring node to evaluate the heuristic algorithm for the trust calculation with the assumption that it has no resource constrains, and it is free from attacks. Trust is calculated in the node level or in the cluster level. In node level trust calculation, each node has to send a trust request about a neighbor node to the network monitoring node via its cluster head. Network monitoring node will find the trust value of that neighbor using the pre-recommendations and replay the trust factor to the requesting node.

In cluster level, network monitoring node initiates the trust calculation for a group. The trust state of a cluster is defined based on the trust state of every individual node in the same cluster. This approach provides the reliable communication in secured medium. But the trust of a node is calculated based on the pre recommendations. Cluster formation and maintenance are not discussed. Momani et al. [7] proposed Bayesian fusion algorithm for inferring trust in wireless sensor networks. This work analyses the results from the communication trust component and the data trust component. The authors calculate both communication and data trust from the neighbor node information. So, the significance of the total trust which was calculated by the Bayesian fusion algorithm is poor. In this paper, the same Bayesian fusion

algorithm is applied but this proposed approach calculates the communication trust and the data trust with more effort.

The proposed approach detects the strange behavior of any node using this trust estimation. By avoiding the communication with any untrustworthy nodes, this proposed approach effectively controls the above-motioned attacks. The rest of this paper is organized as follows: Section II illustrates the details of secure cluster based data aggregation. In section III, we discuss the simulations. Finally we conclude this paper in section IV.

II. SECURE DATA AGGREGATION

A. Problem Statement:

Cluster-based data aggregation is the common solution for energy-efficient data transmission in wireless sensor networks. Because of the open deployment, sensors are vulnerable to security threats. Due to the limited resources of the sensors, it is highly impossible to apply formal cryptographic methods to provide secure communication in WSN. Trust estimation is used to detect the strange behavior of any node in the network. In cluster-based data aggregation, any cluster head is having the high power. Hence the cluster head nodes calculate the trust factor of their cluster members which is known as indirect trust. If any member has lower trust probability, then that member is treated as the malicious node. The aggregator node performs Bayesian fusion algorithm as proposed by Momani et al. [7] to manage the group-based trust.

B. Modified ECBDA

In this approach, the ECBDA algorithm is extended as Secure Cluster-based Data Aggregation (SCDA) to provide secure communication. It also includes five phases, namely

1. CLUSTER FORMATION
2. CLUSTER HEAD ELECTION
3. CLUSTER HEAD ROUTE DISCOVERY
4. SECURE DATA AGGREGATION
5. MAINTENANCE

Cluster formation phase is the same as ECBDA where the network is partitioned into layers based on the node's communication range. K clusters are formed in each layer using k-means algorithm. In the cluster head election phase, a node which is having the highest residual energy and the minimum communication cost will be elected as the CH of a cluster. In SCDA, all CHs have to perform secure data aggregation so the energy requirements to perform the Bayesian fusion algorithm is considered in defining the communication cost.

$$E_c = \left\{ \begin{array}{l} Energy\ \ to\ transmit \\ b\ bits\ \ of\ \ data\ \ to \\ its\ \ next\ \ hop \\ cluster\ \ head \end{array} \right\} + \left\{ \begin{array}{l} Energy\ \ to\ receive \\ b\ bits\ \ of\ \ data \\ from\ \ it's \\ cluster\ \ members \end{array} \right\} + \left\{ \begin{array}{l} Energy\ \ to\ compute \\ the\ Trust\ \ probabilit\ \ y \\ for\ \ Every \\ Cluster\ \ Member \end{array} \right\} \quad (1)$$

$$E_c = E_{tx}(b_{sn,CH_{nh}}) + \sum_{j=1}^{n(C)} [E_{rx}(b) + \varepsilon] \quad (2)$$

where ε is the energy to compute the trust probability. Once a node is elected as a CH, it broadcasts the CH message to its cluster members, to other CHs and to BS. In route discovery phase, BS finds the shortest path for all CHs using Dijktra's algorithm. E_{cost} of the edge between two vertices i and j is

$$E_{cost}(i,j) = \frac{\Delta \times [E_{tx}(i,j) + \varepsilon]}{E_r(i)} \quad (3)$$

$E_{tx}(i, j)$ is the energy required to transfer the data from the node i to j. Δ represents the load of the node i, and the load is defined as the in degree of the node i. In the data aggregation phase, all cluster members send their sensed data to their CH during their allotted time slot. Cluster head nodes are calculating the total trust. So the computation power consumption in the low energy cluster members is avoided. Once the cluster head identifies untrustworthiness of any node inside the cluster, it immediately alerts the remaining cluster members about the malicious node in the cluster. The cluster member discords the data from the malicious node. Security threat is identified by the highest power node, and the malicious node is detected in a small group, thus the energy consumption for detecting the untrustworthy node is reduced in this approach.

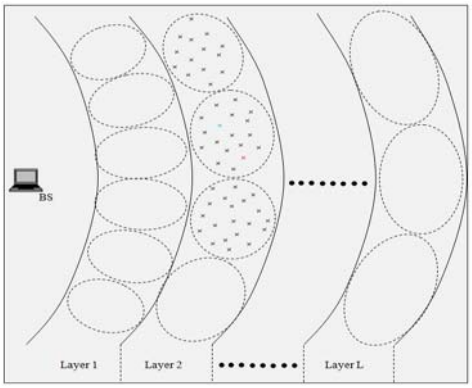

Figure 1. Cluster with one malicious node

Figure 1 shows the cluster formation, and the blue node is the CH of that cluster. Red node indicates the malicious action. Figure 2 shows the communication of the cluster members to their CH. After receiving the data from all cluster members, CH calculates the trust probability. Now, from the probability factor, CH can identify the malicious action of a particular node.

Figure 3 shows that CH broadcasts the alert signal to its entire cluster members. Then any node in that cluster drops the signals from the malicious node and no node will communicate the malicious node. Maintenance phase checks the CH's residual energy at each round. If the residual energy is less than the required threshold value, then new cluster head will be elected from the same cluster. Data aggregation phase of ECBDA is updated to calculate the trust value based on

- Communication trust
- Data trust

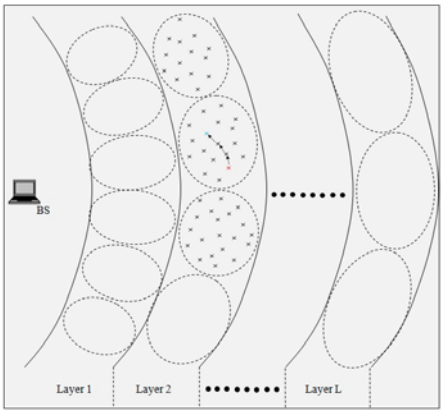

Figure 2. Cluster members communicating the CH

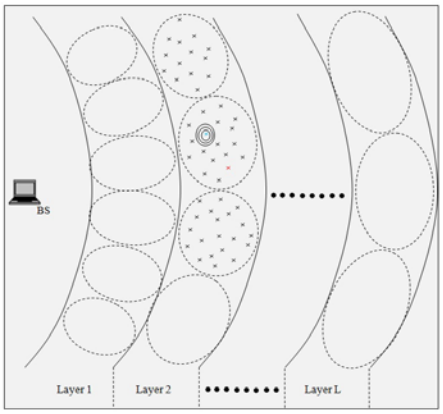

Figure 3. CH alters its cluster members about malicious node

Communication trust value is calculated based on the behavior of any cluster member in the routing message transmission to the cluster head. Data trust is calculated based on the difference between the sensed data from a couple of nearby sensors. Bayesian fusion algorithm calculates the probability of total trust using the prior probabilities of the communication trust and data trust. Communication trust is obtained by the beta distribution. Beta probability density function is used to find the probability of any binary events. Gaussian distribution function is used to calculate the data trust. Bayesian fusion algorithm follows the Bayes' theorem to find overall trust using the two evidences, communication trust and data trust.

C. *Communication trust*

Suppose a cluster has n number of cluster members such as cm_1, cm_2, cm_3, …… cm_n, cluster head (CH) finds the communication trust using the successful and unsuccessful delivery of the sensed data by a cluster member cm_i. Communication trust is defined using the beta density function.

$$f(x_i; \alpha_i, \beta_i) = \frac{\Gamma(\alpha_i + \beta_i)}{\Gamma \alpha_i \, \Gamma \beta_i} x_i^{\alpha_i - 1} (1 - x_i)^{\beta_i - 1} \qquad (4)$$

where α_i represents the number of successful transmissions, and β_i represents the number of failed transmissions of the cluster member cm_i. Here, the Beta function is expressed in terms of Gamma function (Γ). It is defined as

$$\begin{aligned} &\Gamma(t) = (t-1)! \quad \textit{Where} \quad t \textit{ is the whole number} \quad ; \\ &\therefore \ \Gamma(\alpha_i + \beta_i) = ((\alpha_i + \beta_i) - 1)! \\ &\Gamma(\alpha_i) = (\alpha_i - 1)! \\ &\Gamma(\beta_i) = (\beta_i - 1)! \end{aligned} \qquad (5)$$

CH waits for a TDMA frame to collect the data from each cluster member. It finds the successful and unsuccessful delivery of any cluster member based on the current transmission and from the history of the particular cluster member. With these factors, CH finds the current communication trust value of cm_i.

D. *Data trust*

Cluster head finds the data trust using the difference between data from the nearby sensors in its cluster. All close by sensors are grouped into the single cluster. Any event that occurs inside the cluster is monitored and reported by the cluster members. Hence in most of the instances, several sensors report the similar data to the CH. Gaussian distribution function is used to calculate the data trust.

$$f(x_i; \mathrm{M}, \sigma_v) = \frac{1}{\sigma_v \sqrt{2\pi}} e^{-(x_i - \mathrm{M})^2 / 2\sigma_v^2} \qquad (6)$$

$$M = \frac{1}{n} \sum_{i=1}^{n} x_i \quad (7)$$

$$\sigma_v = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - M)^2} \quad (8)$$

where M and σ_v are the mean and variance of the sensed data from the group of sensors in a cluster. x_i is the sensed value of cm_i. If there is no malicious node in the cluster, then all cluster members report similar data. So, Gaussian distribution gives good data trust. If any malicious node forwards some random data, it deviates with the data received from other cluster members. So, it gives poor data trust. Thus, the data trust factor supports to detect any malicious action in the cluster.

E. Bayesian fusion algorithm

This algorithm combines the communication trust and the data trust to find the overall trust of a cluster member. The posterior probability of the total trust of the cluster member $P(T_{cm})$ is obtained with the prior probability of communication trust $P(C_{cm})$ and the prior probability of the data trust $P(D_{cm})$.

$$P(T_{cm}) = P(T_{cm} / C_{cm}) * P(T_{cm} / D_{cm}) \quad (9)$$

where $P(T_{cm}/C_{cm})$ and $P(T_{cm}/D_{cm})$ are the conditional probabilities. These two probability values are calculated based on the Bayes' theorem.

$$P(T_{cm} / C_{cm}) = \frac{P(C_{cm} / T_{cm}) * P(T_{cm})}{P(C_{cm})} \quad (10)$$

$$P(T_{cm} / D_{cm}) = \frac{P(D_{cm} / T_{cm}) * P(T_{cm})}{P(D_{cm})} \quad (11)$$

Based on this calculated total trust value and the threshold value, CH decides if the current cluster member cm is the regular node or the malicious node. Probability threshold value P_{Th} is estimated based on the average of communication trust and average of data trust factors. If $P(T_{cm})$ is less than P_{Th}, then the cluster member cm_i is declared as the malicious node.

Figure 4 shows the overall working flow of SCDA. In this approach, the cluster head which is having high power is doing the trust calculation. The node which is having the most ability to perform the data collection and trust evaluation for a whole round is identified as cluster head. The computation overhead is avoided in all the relay nodes which are participating in the communication. Once the cluster head identifies any malicious behavior by measuring the trust value, it instantly indicates its cluster members about the existence of the malicious node in the cluster. So, cluster members drop the packet from the malicious node, and it will update its routing path to the cluster head by avoiding the malicious node in its routing. Hence the secure transmission is ensured by using the trust factor of the cluster member. And the energy consumption for the computation of trust factor is reduced by using cluster-based security. Thus, the proposed approach provides secure, energy-efficient in-networking process for wireless sensor networks.

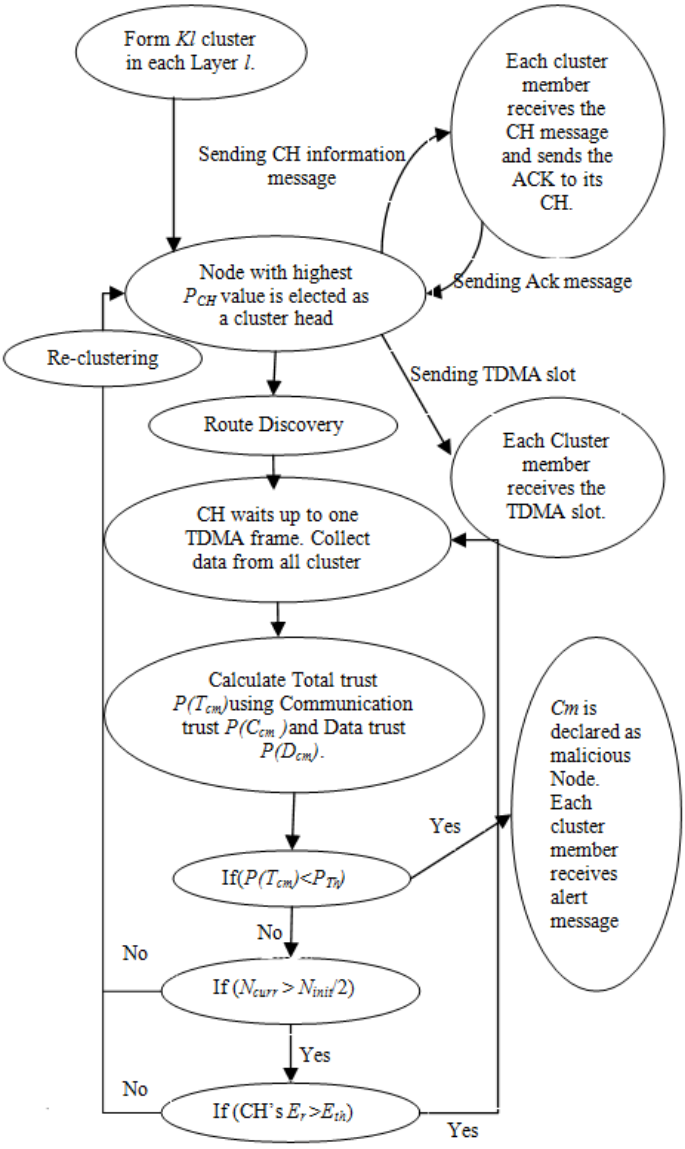

Figure 4. SCDA Flow diagram

III. EXPERIMENT AND RESULT

A. *Simulation parameters*

This approach is validated through NS2 simulator. Random deployment model is used for the sensor network topology setup. For the simulation scenario, the malicious nodes are set to generate the packets with random information whereas the regular sensor nodes are set to generate the packets with a uniform range. Initially, 5 malicious nodes were defined to measure the energy consumption by varying the number of regular nodes. For a few analyses, the number of malicious nodes is also varied. Table 1 shows the complete parameter settings used for this approach.

Table -1 SCDA Simulation Parameter Settings

Parameter	**Value**
Number of nodes	100 - 300

Parameter	Value
Number of malicious nodes	5 - 25
Receive Power	22.2 mW
Transmit Power	31.2mW
Data packet size	40 bytes
Traffic Type	CBR

B. *Algorithms used for comparison*

The efficiency of SCDA is compared with Gaussian Reputation System for Sensor Networks (GRSSN) which was proposed by Momani et al. [7]. The authors strongly suggest and prove that one trust component by itself cannot fully decide the trustworthiness of nodes in WSNs. The individual trust components are calculated from the fellow node. The results of the proposed approach are also compared with Hierarchical Trust Management (HTM) protocol proposed by Bao et al. [8]. HTM maintains two levels of trust: sensor node-level trust and CH-level trust. Each sensor node evaluates the other sensor nodes in the same cluster while each CH evaluates the other CHs and sensor nodes in its cluster. The peer-to-peer trust evaluation is periodically updated based on either direct observations or indirect observations.

When two nodes are neighbors within radio range, they evaluate each other based on direct observations. Each sensor node sends its trust evaluation results toward the other sensor nodes in the same cluster to its CH. Each CH performs trust evaluation toward all sensor nodes within its cluster. Similarly, each CH sends its trust evaluation results toward the other CHs in the WSN to a "CH commander" which may reside on the base station if one is available, or on a CH elected if a base station is not available. The CH commander performs trust evaluation toward all CHs in the system. Trust is evaluated based on intimacy, honesty, energy, and unselfishness of a node. Clusters are formed based on a hybrid, energy-efficient, distributed clustering approach (HEED) which was proposed by Younis and Fahmy [9].

C. *Results and discussion*

Energy dissipated in the network is analyzed with fixed number of malicious nodes. Figure 5 shows the comparison of energy dissipated. In GRSSN, trust calculation is performed in every relay node. Despite regular sensor nodes having lesser energy, all nodes in the communication path are forced to compute the total trust hence the average energy consumed by the GRSSN is higher than HTM and SCDA. In HTM, trust is evaluated in relay nodes and the CHs. It consumes lower energy when compared to GRSSN. But it changes the cluster head over a time period. So, it needs more energy for setting the clusters once in a round. SCDA uses the modified ECBDA, and CHs alone are calculating the total trust. Hence the average energy dissipated in SCDA is less.

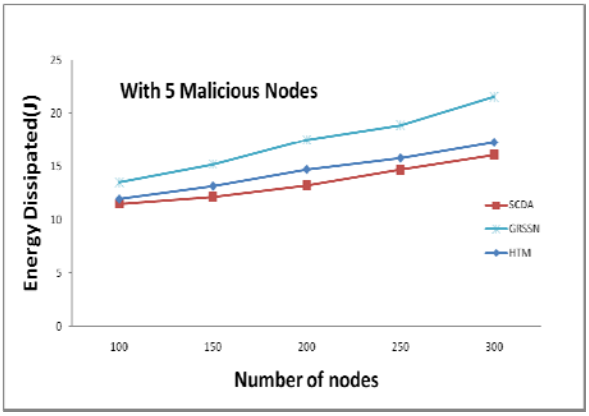

Figure 5. Comparison of energy dissipated with fixed malicious nodes

Comparison of trust probability calculation is shown in Figure 6. GRSSN and SCDA use the second hand information for calculating the trust factor. So, the trust probabilities in both approaches are close to similar. But in HTM, trust evaluation is performed on the communication parameter, not in data factor. So, the trust probability is lower in HTM when compared to SCDA and GRSSN.

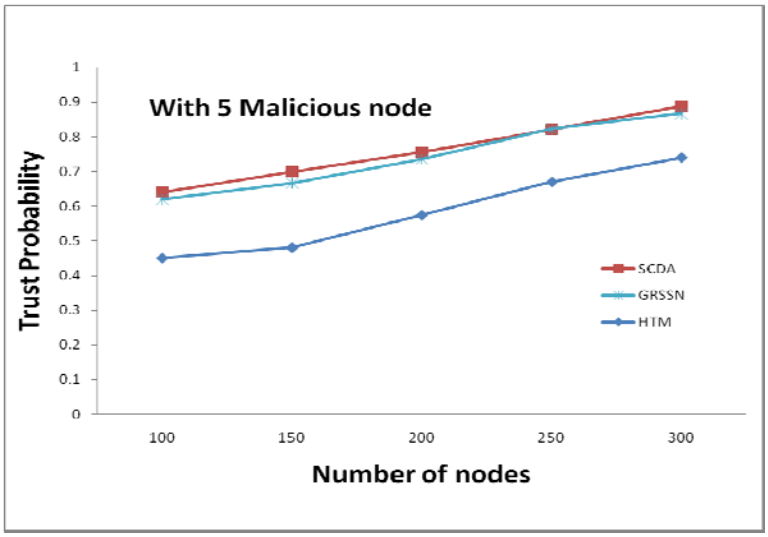

Figure 6. Comparison of trust probability with fixed malicious nodes

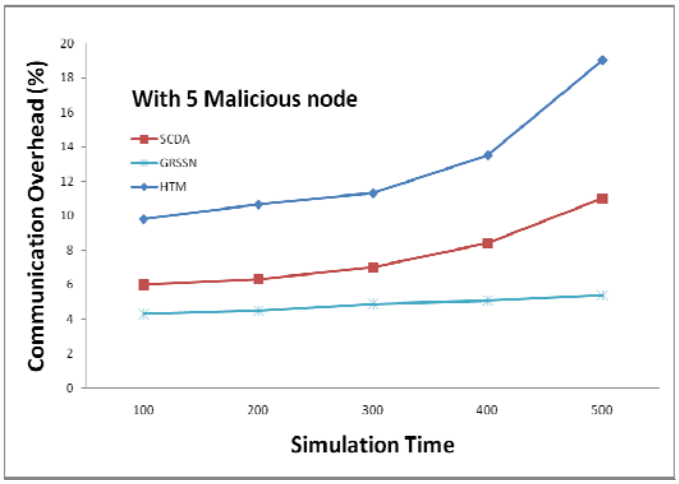

Figure 7. Comparison of communication overhead with fixed malicious nodes

Figure 7 shows the comparison of communication overhead. GRSSN uses tree-based trust calculation so the communication overhead is minimal. But it does not effectively detect the malicious node so it consumes more energy because of the security attacks. HTM finds the trust factor based on the geographical routing. Overhead is very high in HTM for routing message transmission to find the relay node. But in the proposed approach, clusters are formed initially, and re-clustering is performed only when a cluster loses 50% of the cluster members. Hence in SCDA communication overhead is lower than HTM. It effectively detects the malicious node with minimum energy requirement.

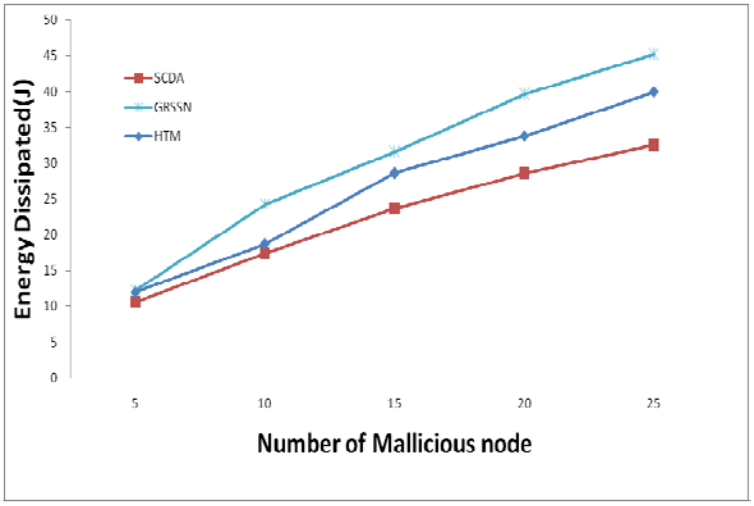

Figure 8. Comparison of energy consumption with fixed regular nodes

The performance of SCDA is analyzed by varying the number of malicious nodes. Figure 8 shows the comparison of the number of malicious nodes and average energy consumption. GRSSN and HTM drain more energy with increased number of malicious nodes. In SCDA, CHs effectively identify the malicious nodes and they immediately alert their cluster members about the presence of malicious node in the cluster. The cluster members avoid any communication to and from the malicious node. Hence the overall energy consumption is reduced in SCDA.

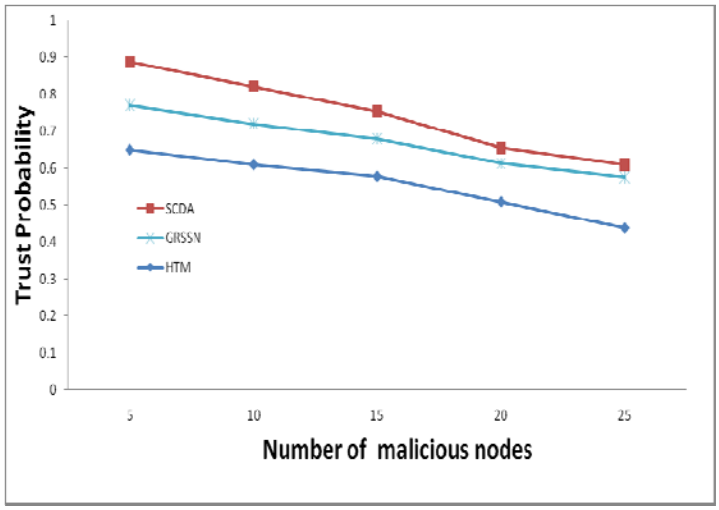

Figure 9. Comparison of trust probability with fixed regular nodes

Figure 9 exhibits the trust probability by varying the number of malicious nodes. In GRSSN and HTM, many nodes are losing their total energy soon. Consequently, with reduced number of regular nodes and increased number of malicious nodes, GRSSN and HTM could not calculate the trust probability well. But with the proper cluster maintenance of SCDA, the proposed approach calculates the trust probability with maximum number of regular nodes.

IV.CONCLUSION

In this paper, secure cluster-based data aggregation is proposed. Wireless sensor networks are applied in a range of vital civilization applications in which most of the applications carry out remote monitoring. Due to its open nature, WSN is prone to security attacks. Sensor nodes have limited resources and hence trust-based secure communication method is designed. SCDA forms and maintains the cluster using ECBDA approach with a few changes. CHs, the nodes with highest energy, are calculating the total trust

probability. Thus, the energy of every node which is involved in the data communication is saved. Bayesian fusion algorithm is applied to calculate the trustworthiness of a sensor node. More nodes are giving supporting information to the CHs for a long time so the fairness of trust probability is excellent in SCDA. Simulation results show that the proposed approach outperforms the existing methods in security as well as energy.

REFERENCES

[1] Ranjani, S. Siva, et al. "Achieving Energy Conservation by Cluster Based Data Aggregation in Wireless Sensor Networks." *Wireless personal communications* 73.3 (2013): 731-751.

[2] Du, X., and Chen, H. H, Security in wireless sensor networks, *IEEE Wireless Communications*, 15(4), (2008), 60-66.

[3] Gargano, L., and Rescigno, A. A, Collision-free path coloring with application to minimum-delay gathering in sensor networks, *Discrete Applied Mathematics*, 157(8), (2009), 1858-1872.

[4] Lopez, J., Roman, R., Agudo, I., and Fernandez-Gago, C, Trust management systems for wireless sensor networks: Best practices, *Computer Communications*, 33(9), (2010), 1086-1093.

[5] Shaikh, R. A., Jameel, H., d'Auriol, B. J., Lee, H., Lee, S., & Song, Y. J, Group-based trust management scheme for clustered wireless sensor networks, *IEEE Transactions on Parallel and Distributed Systems*, 20(11), (2009), 1698-1712.

[6] Dhulipala, V. S., Karthik, N., and Chandrasekaran, R. M, A novel heuristic approach based trust worthy architecture for wireless sensor networks, *Wireless personal communications*, 70(1), (2013), 189-205.

[7] Momani, M., Challa, S., and Alhmouz, R, Bayesian fusion algorithm for inferring trust in wireless sensor networks, *Journal of networks*, 5(7), (2010), 815-822.

[8] Bao, F., Chen, R., Chang, M., and Cho, J. H, Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection, *IEEE Transactions on Network and Service Management*, , 9(2), (2012), 169-183.

[9] Younis, O., and Fahmy, S, HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks, *IEEE Transactions on Mobile Computing*, 3(4), (2004), 366-379.