ISSN: 2321-7782 (Online)
Volume 3, Issue 5, May 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Maximizing the Lifetime and Data Security of WSNs using HEF Algorithm and Paillier Homomorphism

Bhavana D[1]
PG Scholar
Dept. of ISE
NIE, Mysore
Karnataka – India

Chinnaswamy C N[2]
Asso. Professor
Dept. of ISE
NIE, Mysore
Karnataka – India

Dr. T H Sreenivas[3]
Professor
Dept. of ISE
NIE, Mysore
Karnataka – India

Abstract: Wireless Sensor Networks have gained wide popularity in the recent years for its high-ranking applications such as remote environment monitoring, target tracking, safety-critical monitoring etc. However Wireless Sensor Networks face many constraints like low computational power, small storage, and limited energy resources. Two of the chief issues associated with Wireless Sensor Networks are network lifetime and data security which we aim to address here. In our proposed system we aim to maximize the network lifetime of Wireless Sensor Networks by choosing High Energy First (HEF) clustering algorithm as a design reference model for cluster head selection, which is to the best of our knowledge proved to be an optimal clustering policy under certain ideal conditions, together with a sleep/active algorithm to avoid any unnecessary energy dissipation by the sensor nodes. We also aim to address the issue of sensor data security by opting for Paillier homomorphism encryption, which is an end to end data security plan that can guarantee end-to-end data confidentiality with less transmission latency and computation cost contrasting with the hop by hop data security which involves more consumption of battery power due to several decryptions at every hop.

Keywords: Data security in WSNs; Efficient cluster head selection using HEF; HEF algorithm; Lifetime maximization of WSNs; Paillier Homomorphism for data security.

I. INTRODUCTION

Wireless sensor networks (WSNs) have predominantly become a new trend in the technology. WSNs are a collection of wireless nodes having limited energy capabilities, deployed arbitrarily over a dynamically evolving environment, may be mobile or stationary, for observing physical phenomena like humidity, temperature sensing, health monitoring, seismic events, motion etc. It is a resource constraint network, in which all sensor nodes have finite resources. Hence it becomes essential to save energy and resources wherever it is possible for its longer functioning, especially in time critical networks. One way of doing it is by implementing data aggregation. Data aggregation can be done with the help of a clustering plan. Clustering reduces the amount of traffic occurring in the network by means of organizing sensor nodes into groups, collecting and compressing the sensed data together and then transmitting only the compact data to base station, considerably reducing the battery power used by every sensor in WSNs.

The architecture of the sensor network also plays important role in determining the performance and lifetime of WSNs together with data aggregation. Here we have opted Hierarchical Cluster Architecture of WSNs (HC-WSNs). A typical HC-WSN comprises of a Base Station (BS), several Cluster Head/Aggregator nodes (CH) and Regular sensor nodes. All the nodes are organized into clusters and a cluster head/aggregator is chosen to head every cluster. Since the aggregator is accountable for most of the operations such as data aggregation from regular nodes and forwarding the same to the BS, it necessarily needs to possess more energy than the regular nodes. Hence cluster head selection becomes be an important process here.

Various algorithms are proposed for the selection of aggregators for maximizing the network life time. Although most of them try to achieve maximum lifetime of the network they do not necessarily provide predictability for the same. For the best of our knowledge our chosen High Energy First clustering (HEF) maximizes the network life time & schedulability bounds under Ideal Conditions for Optimality of HEF (ICOH).

Another issue that is related to the WSNs is the security of data. Many sensor networks have mission-critical tasks and thus require the security aspect to be considered. In a WSN, there are usually certain nodes, called aggregators, which help to aggregate information requested by user queries. When an aggregator node's security is compromised, it is easy for the adversary to inject false data into sensor networks. Thus, the aggregators are prone to attacks. Another possible way attackers can pose a threat is by compromising a sensor node and injecting forged data through it. Without authentication, the attackers can fool the aggregators into reporting false data injected at the sensor nodes to the base station.

Secure data aggregation requires authentication, confidentiality, and integrity. Moreover, it also requires the cooperation of sensor nodes to identify the compromised sensors. Improper utilization of the sensed and aggregated information or using forged information may cause information leakage and provide inaccurate results.

Confidentiality requires the data to be transmitted in encrypted text while data aggregation is usually based on plain text data. In hop-by-hop encrypted data aggregation, all the intermediate sensor nodes have to decrypt the received encrypted data and apply aggregation function on it. Due to many decryptions performed by the intermediate nodes it consumes more battery power and does not provide end-to-end security. But in end-to-end encrypted data aggregation, intermediate nodes can aggregate the encrypted text directly without decrypting the messages. Compared to the hop-by-hop encryption, it guarantees the end-to-end data confidentiality and results in less transmission latency and computation cost. One such approach that provides such an end-to-security is a cipher text based data aggregation scheme called Paillier homomorphism that we have chosen here. It is based on a certain encryption transformation that is performed on the data without necessarily decrypting the data at every node and hence saving energy as well as imparting security.

II. Literature Survey

A. Related Works

Several solutions to maximize network lifetime are available, and each approach provides different magnitudes of energy savings and levels of efficiency. Among the network lifetime optimization research works, clustering of sensors into groups is a popular strategy to save energy and bandwidth, whereby cluster heads act as routers relaying all packets from sensors to the base station. Hence the cluster head selection (CHS) phase plays the most dominant role with respect to the optimality and predictability of the entire network operation. A smart cluster head selection strategy can significantly reduce energy consumption, which in turn prolongs the network lifetime.

In previous years, many cluster head selection algorithms for HC-WSNs papers have been published, and they are mostly driven by their own cluster head selection criteria such as prolonging network lifetime and energy dissipation, and require extensive a priori information of each node.

The election of CHS algorithms without energy awareness using the Lower Identity (ID) heuristic approach has been proposed by [1], where in it uses the static node ID scheme to choose the node with the minimum node ID as a cluster head. Another CH election process using secret ballot votes [2] has been proposed to identify a node that receives the majority vote of those seated in a cluster as a new cluster head. The node which gets the second highest number of votes would be elected as the vice cluster head.

A method that uses back off mechanisms [3] in the contention window was proposed to guarantee that a sensor node will be elected as a cluster head at least a certain number of times so that all nodes have an opportunity to work as a cluster head at each round.

A framework for dynamically organizing mobile nodes and electing a dominating-set in highly spontaneous large-scale mobile ad hoc networks was given by [4] with an aim to support location-based routing protocol. This geographically-based approach selects a node as a cluster head that has the highest spatial-associativity with respect to a specific cluster. This geographically- oriented approach forces the cluster head to stay close to its Virtual-Cluster-Centre (VCC).

Clustering and In-network Processing Routing Algorithm (CIPRA) [5], which is another method for cluster head selection, is based on the total number of sensors, its unique ID and the current round number. In CIPRA, each node decides whether or not to elect itself as a cluster head by dividing the product value of its own unique ID, and the current round number by the modulus of N for every round. But using CIPRA technique, only one CH can be selected at a time to reduce energy requirement. To select multiple CHs, residual energy should be considered.

The popular Low-Energy Adaptive Clustering Hierarchy (LEACH) which selects cluster heads based on a static probability function without energy awareness was proposed by [6]. The selection is based on the desired percentage of CHs for the network and number of times the node has been a CH so far. LEACH is a widely known cluster head election algorithm with four key advantages of reduction in energy dissipation, distribution of energy-usage, enhancement of network lifetime, and management of network coverage on the set of Hierarchical wireless sensor network nodes. However, though LEACH works well to enhance network lifetime statistically, it does not offer deterministic optimization. There are many papers that propose to improve LEACH, but they tend to be statistical without reliable lifetime assurance.

The cluster head selection processes mentioned above for Cluster head selection Algorithms Without Energy Awareness clustering do not require sensors to be aware of any a priori energy information about the nodes. But without the awareness of the energy information, cluster heads cannot be swapped, and also the traffic loads cannot be divided. Hence, it is difficult for sensors to select the most appropriate cluster heads to increase their network lifetime, and hot-spot cluster head sensors die rapidly. Some of the algorithms that consider having a priori information are mentioned below.

An algorithm to avoid non-uniform distribution of cluster heads was proposed by [7] wherein the cluster heads are selected according to their residual energy, and a predefined energy level difference is used to enforce the cluster head rotation inside the cluster.

A particular cluster head selection strategy [8] says that all nodes contain a cluster head probability and that is recomputed based on each round, and node priority (such as residual energy and node ID) will take effect in case of a tie.

Another CH selection algorithm Hybrid Energy-Efficient Distributed clustering (HEED)[9] periodically selects cluster heads based on a hybrid of residual energy, and a secondary index (such as node proximity to its neighbors or node degree). Residual energy is used to set the initial set of cluster heads. Intra cluster communication cost is used for deciding to join a cluster or not. The secondary index will be considered if two nodes have the same residual energy.

There are also some algorithms that try to get as much information as possible to compute the best clustering, and to maximize the overall network lifetime. These algorithms are mostly centralized.

In these algorithms, in addition to only collecting data from sensors, the base station or a centralized centre will also determine the working status of the sensors.

LEACH-Centralized (LEACH-C), an enhancement to the previously mentioned LEACH was given by [10]. A centralized base station using the LEACH-C algorithm chooses a cluster head based on a hybrid of location information and energy levels. LEACH-C maintains enough separation distance to keep cluster head nodes separate from each other.

However, none of the above cluster head selection algorithms addresses the schedulability analysis issue in their proposed algorithms. Although some of their approaches are optimal, the predictability of optimality is stochastic (non-deterministic).

On the other hand our chosen HEF cluster head selection algorithm [11], chooses the optimal Cluster Head, maximizes the network life time & schedulability bounds also derived under ICOH. The core idea is to choose the highest-ranking energy residue sensor as a cluster head and Cluster formation is similar to LEACH. After each round, energy consumption of CH and regular nodes are calculated. From this calculated value, those nodes contain higher residual energy will be selected as CH. So the drain rate of the nodes will be linear and packet delivery rate is increased. At the end of each round, utilization of energy is better and hence network lifetime is prolonged in higher level compared to other algorithms. It also supports for deriving life time bounds to ensure predictability of the nodes. This prediction is very much helpful for real time WSN.

Another of the major concerns with respect to WSNs that we have taken into account here is the security of data. Since WSNs main aim is collect the sensed data, it has to protect this important data that it transmits across the network so that it reaches the BS unhampered and uncompromised. Since the aggregator node is the one that collects the sensed data from all of its cluster nodes, an attack at the aggregator node will affect severely when it comes to security of data. Several works have been done previously with regard to secure data aggregation in WSNs.

The work on secure data aggregation can be classified based on encryption of data at specific nodes.

A Hop-by-Hop encrypted data aggregation for WSNs with dynamic cluster-based architectures which is called Delayed Hop-by-hop Authentication (DHA) scheme [12] provides hop-by-hop data integrity and data freshness only using individual keys. The intermediate nodes decrypt every message received by them so as to get the plaintext using individual keys. Then aggregate the plaintext according to the aggregate function, and encrypt the aggregate result before transmitting it. In this all the intermediate sensor node has to decrypt the received data and apply aggregation function on it. Due to many decryptions performed by the intermediate node it’s consuming more battery power and does not provide end-to-end security.

A set of end-to-end encrypted data aggregation protocols were proposed by [13] [14] and [15] in order to overcome the drawbacks of the hop by-hop encrypted data aggregation. In those schemes, intermediate nodes can aggregate the cipher text directly without decrypting the messages. Compared to hop-by-hop, it can guarantee the end-to-end data confidentiality and also has less transmission latency and computation cost. Adversaries will not be able to recognize what data is being sent during the transmission. In terms of privacy, their schemes aim to eliminate redundant reading for data aggregating but this reading remains secret to the aggregator.

Our chosen scheme Paillier Homomorphism (PH) is an encryption transformation that allows direct computation on encrypted data. In homomorphic encryption certain aggregation functions can be calculated on the encrypted data. The data is encrypted and sent toward the base station, and the aggregation function is applied on the encrypted data by the sensors along the path. The base station receives the encrypted aggregate result and decrypts it. The encryption transformations can be either multiplicative or additive. Here we have chosen the additive transformation. This means that in order to calculate the SUM of two values, we can apply some function to their encrypted counterparts and then decrypt the result of the SUM operation at sink node. The data would be encrypted at the sensor node, the SUM or AVERAGE would be calculated as the aggregate result follows a path to the base station, and the final result would be decrypted at the base station.

B. *Existing System*

Since the main concern with the WSNs is the energy constraint and security, there have been several attempts made towards saving its energy together with delivering efficient service.

Most of the networks currently use Hierarchical clustering in WSNs, as it is proved to be contributing when it comes to energy saving. There are many HC-WSN algorithms that have been proposed for efficient clustering as well as Cluster Head selection such as LEACH, HEED, LEACH-C in order to save energy as much as possible. In the literature, most of these algorithms although try to achieve maximum network lifetime and optimality, none of the above cluster head selection algorithms addresses the schedulability analysis issue in their proposed algorithms for predictability of network lifetime.

Most routing protocols for HC-WSNs are vulnerable to a number of security attacks, including jamming, spoofing, replay, etc. However, because these are cluster based protocols, they rely fundamentally on the CHs for data aggregation and routing, and attacks involving CHs are the most damaging. If an intruder manages to become a CH, he can stage attacks such as sinkhole and selective forwarding, thus capturing, manipulating sensitive data and disrupting the workings of the network. There are many secure data aggregation protocols in the literature which aim at providing the necessary security for the data at the aggregator nodes such as Secure Aggregation [16], Secure Information Aggregation [17],Secure Differential Data Aggregation [18], but all these are plain text based therefore decryption of data is required at the nodes to apply these. Hence the protocol may be vulnerable if a parent and a child node in the hierarchy are compromised.

C. Problem Statement

Since wireless sensor networks are resource constrained, inefficient usage of sensor nodes battery power can lead to premature death of the network. Also without secure means for the transmission of sensed data to the base station, adversaries can attempt to pry on the network data at different points like sensor nodes, aggregators and gain knowledge or modify the data being transmitted.

Recently, improving energy efficiency of WSNs has gained a lot of popularity. There have been several schemes developed to increase the lifetime of a WSN as well as protect the sensed data during transmission.

We address these issues associated with WSNs by Maximizing the Lifetime and Data Security of WSN using HEF algorithm with sleep/awake and Paillier Homomorphism.

D. Proposed System

In our proposed system we have chosen High Energy First (HEF) Hierarchical Clustering Cluster Head selection algorithm.

Unlike most of the HC-WSN clustering algorithms like LEACH, CIPRA etc mentioned in the literature that do not take energy information of the sensor nodes into consideration, our chosen HEF gathers the residual energy of every sensor node in the network into account for cluster formation and also for cluster head selection. Although there are several other algorithms that work based on the energy consideration of sensor nodes with their approaches being optimal, the predictability of their optimality is stochastic (non-deterministic).

Since HEF is based on residual energy, at different rounds nodes that have maximum energy will be selected as CH, so the drain rate of the nodes will be linear and packet delivery rate is increased. At the end of each round, utilization of energy is better and hence network lifetime is prolonged in higher level compared to other algorithms. It also supports for deriving life time bounds for performing schedulability test to ensure predictability of the nodes under ICOH. This prediction is very much helpful for real time WSN.

As an enhancement to this we propose here cluster node sleep/active algorithm in which, based on the energy information collected, a threshold energy level is computed and those nodes whose energy are below this level will be sent to sleep by the aggregator and awaken only at the subsequent rounds where they would be useful. The optimality of HEF algorithm along with active/sleep ensures to balance the energies of all the nodes within a cluster thereby reducing energy depletion by efficiently forming clusters in set-up phase.

With the purpose of adding security at the data aggregation level, we have chosen the Paillier homomorphic cryptosystem which is an encryption transformation where certain aggregation functions can be calculated on the encrypted data. The data is encrypted and sent towards the base station, while sensors along the path apply the aggregation function on the encrypted data. This is much better than end-to-end approaches in the literature because the base station receives the encrypted aggregate result and decrypts it essentially allowing only the BS to know the gist of the data thus providing security to the data even at the aggregator level.

Thus with our proposed system we aim to maximize the network lifetime with predictability and energy savings along with the added advantage of security of the data at the aggregator level.

III. System Architecture

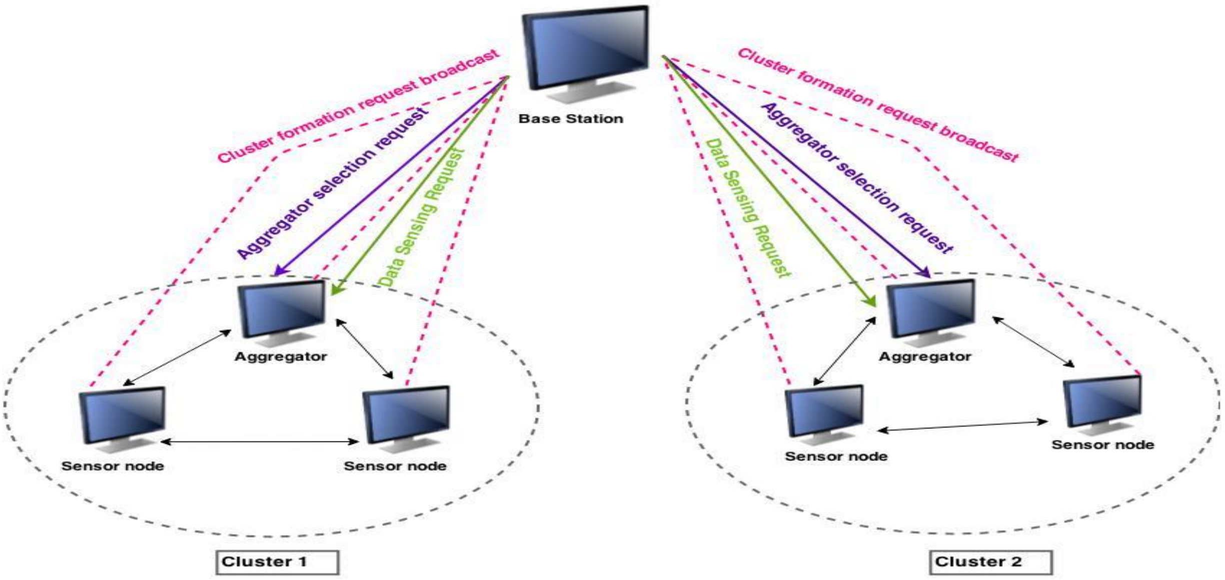


Fig.1 System architecture of proposed system

Fig.1 shows the architecture of the proposed system of a hierarchical wireless sensor network consisting of a controlling base station and several sensor nodes. The base station is the root node at the top of the hierarchy that controls all the activities and functions performed by the nodes in the network. The cluster heads/ aggregators form the second level of the hierarchy which receives instructions from the base station to control and manage the activities performed by the sensor nodes working under them. The sensor nodes working under the aggregators occupy the lowest level of hierarchy and perform as instructed by the aggregators.

Thus the base station is the controlling centre that controls the cluster head/aggregators which in turn manage the sensor nodes under them in their respective clusters.

IV. System Implementation

Base Station

The base station is the controlling centre, hence it initiates requests for cluster formation and aggregator selection processes to the sensor nodes. It also requests the aggregator for data sensing whenever required. It generates encryption keys using homomorphic encryption scheme and shares the same with the nodes for encryption and secure transmission of sensed data through the network. The base station also decrypts the encrypted data received from the sensor nodes, performs certain computations on it and stores it in a log file for the end user to read.

Sensor Nodes

The sensor nodes perform the operations requested by the base station and the aggregator. They can either function as an aggregator or a regular sensor node depending upon their role. The sensor nodes can receive three types of requests from the base station. It can be for cluster formation, aggregator selection or data sensing. They exchange battery values and IPs with the neighbors and form clusters when they are requested for cluster formation. They calculate threshold battery value, compare each other's battery values and elect aggregators using HEF algorithm when requested for aggregator selection. They sense, encrypt and transmit the data to the aggregator when requested for data sensing. The aggregator node will add its own sensed and encrypted data to the received data and in turn transmits it to the base station. The same process repeats for successive rounds until the nodes die or terminated by the user.

A. *Cluster Formation and Aggregator Selection*

The clustering of nodes is done by exchanging battery values with the neighbouring nodes and cluster heads/aggregators are selected using the High Energy First (HEF) cluster head selection algorithm. All the nodes in the network exchange their node IPs and battery power and form clusters, the nodes within the clusters compare each other's battery values and elect the one having the highest battery power as Cluster Aggregator using HEF algorithm.

1) *HEF algorithm:* The idea behind the HEF clustering algorithm is to choose the node with the highest residual energy within every cluster as the cluster head/aggregator respectively.

- HEF selects aggregator according to the energy remaining for each sensor node by comparing the battery values of every sensor node within the cluster. The node with the highest battery value will be elected as the aggregator for that cluster for that particular round.
- The aggregator of each cluster broadcasts its IP to the sensor nodes of its cluster announcing that it is the aggregator for the current round.
- Each aggregator acknowledges itself by sending its IP address to the BS and registers as the cluster aggregator for the current round.
- The aggregator forwards the time schedule sent by the BS to its cluster members for each round.

2) *Residual Energy of Sensor Node and Cluster Head:* Suppose the residual energy level of a sensor node i in the T-th round is denoted as $E_i(T)$ and the energy consumed by a cluster head and a regular node as W_c and W_r respectively.

If the residual energy of the node i is represented as $E_i(t)$ at the beginning of the t^{th} round and $E_i(t+1)$ at the very next round, then the residual energy of the node in the t+1 round is given as

$$Ei(t+1) = \begin{cases} Ei(t) - Wr & \textit{if node is regular node} \\ Ei(t) - Wc & \textit{if node is cluster head} \end{cases}$$

HEF comes with the Ideal Conditions for the Optimality of HEF (ICOH) [11] that has to be satisfied when choosing a node as the aggregator or cluster head for achieving longer network lifetime of WSNs. They are;

- All nodes must operate in a working-conserving mode. In other words, each node works as a cluster head, or a regular sensor in a round.
- The energy consumptions of W_c and W_r are constant during the entire operation, where $W_c \geq W_r$.

3) *Awake/Sleep Scheduling:* To conserve more energy of the sensor nodes, a threshold battery value is calculated by the aggregator at every round by taking into account all the battery values of the nodes in a cluster, and the nodes whose

battery values are below this threshold will be sent to sleep until their battery values become eligible to be included in a particular round. This process repeats for every round.

B. Key Generation and Distribution

In order to provide security to the data that is sensed by the sensor nodes in the network, the data has to be encrypted. Here, for encryption of the sensed data we have chosen a homomorphic encryption scheme called Paillier Cryptosystem.

Homomorphic encryption is a form of encryption that allows computations to be carried out on cipher text, thus generating an encrypted result which, when decrypted, matches the result of operations performed on the plaintext.

The Paillier cryptosystem, named after and invented by Pascal Paillier in 1999, is a probabilistic asymmetric algorithm for public key cryptography. The scheme is an additive homomorphic cryptosystem; this means that, given only the public-key and the encryption of m_1 and m_2, one can compute the encryption of $m_{1+}m_2$.

For every aggregation phase, the BS computes the public (encryption) keys (n, g) and the private (decryption) keys (μ, λ) and sends the public encryption key along with the data sensing request to the sensor nodes for data encryption.

The key generation is done using the following rules;

- Choose two large prime numbers *p* and *q* randomly and independently of each other such that $\gcd(pq,(p-1)(q-1)) = 1$. This property is assured if both primes are of equal length.
- Compute $n = pq$ and $\lambda = \operatorname{lcm}(p-1, q-1)$.
- Select random integer g where $g \in \mathbb{Z}^*_{n^2}$
- Ensure n divides the order of g by checking the existence of the following modular multiplicative inverse: $\mu = (L(g^\lambda \operatorname{mod} n^2))^{-1} \operatorname{mod} n$,

  Where function *L* is defined as $L(u) = \frac{u-1}{n}$.

Note that the notation $\frac{a}{b}$ does not denote the modular multiplication of *a* times the modular multiplicative inverse of *b* but rather the quotient of *a* divided by *b*, i.e., the largest integer value $v \geq 0$ to satisfy the relation $a \geq vb$. The public (encryption) key is $(n, g)$and the private (decryption) key is (λ, μ).

All aggregator nodes share a common key for confidential communication called as Common Aggregator Key (CAK) that is updated for every aggregation phase. To update CAK, BS broadcasts update message consisting of updated Common Updation Key (CUK). Upon receiving CUK every aggregator node updates CAK.

C. Data Sensing and Forwarding

Each node in the cluster senses the data upon receiving data sensing request, encrypts it using the secret public key sent by the BS and then forwards the encrypted data to the aggregator node. Here we have chosen a temperature sensor LM35 to sense the surrounding temperature.

The data thus sensed is encrypted using secret public keys (n, g) sent by the BS along with data sensing request as follows;

- Let m be a message to be encrypted where $m \in \mathbb{Z}_n$
- Select random r where $r \in \mathbb{Z}^*_n$

- Compute cipher text as: $c = g^m \cdot r^n \bmod n^2$

The node then forwards it to the cluster head/aggregator node.

D. Data Aggregation and Encryption

At the cluster head/aggregator node, data received from all cluster sensor nodes are added using homomorphic encryption scheme. Homomorphic encryption can be additive or multiplicative. We have adopted the additive scheme here. The additive homomorphic property is described below.

Homomorphic addition of plaintexts:

The product of two cipher texts will decrypt to the sum of their corresponding plaintexts,

$$\left(E(m_1, r_1).E(m_{2,} r_2) mod\ n^2\right) = m_{1+} m_2\ mod\ n$$

The product of a cipher text with a plaintext raising g will decrypt to the sum of the corresponding plaintexts,

$$\left(E(m_{1,} r_1).g^{m_2} mod\ n^2\right) = m_1 + m_2\ mod\ n$$

The data thus encrypted is forwarded to the base station by all the cluster aggregator nodes.

E. Data Decryption and Computation

The base station (Sink) receives the aggregate sensed data from aggregator nodes of all clusters along with the number of active nodes in a given cluster. It decrypts the received aggregate using its private keys (λ, μ).

The decryption is done using the following decryption rules;

- Let ***c*** be the cipher text to decrypt, where $c \in \mathbb{Z}^*_{n^2}$
- Compute the plaintext message as: $m = L(c^\lambda \bmod n^2) \cdot \mu \bmod n$

The average of the decrypted aggregate data is computed by dividing the aggregate sum by the number of active nodes in a given cluster which gives the average aggregate data sensed by every cluster. The BS gets to know the result/sum of the data sent but will not come to know about the values that were added to get the result/sum thus protecting its integrity.

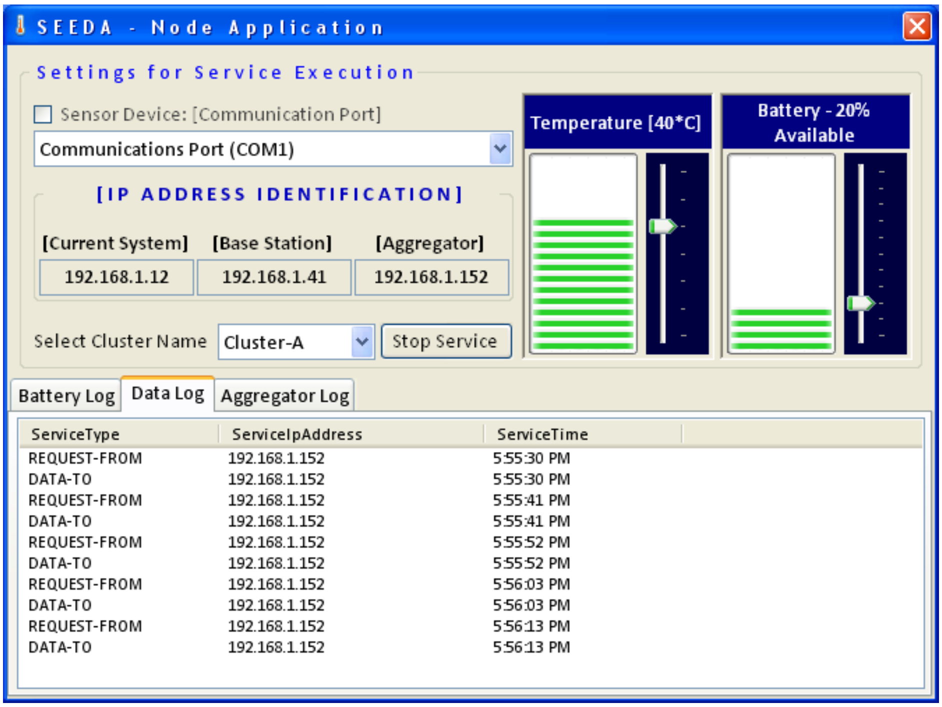
V. RESULTS

The node functioning as aggregator maintains a log of all the sensors that are under it so that it can get a fair idea as to which node is active, sleeping or dead based on their residual battery levels. This information is shown under the 'Aggregator log' of the aggregator. The nodes that are active are represented with a system icon colored green , the nodes that have gone dead after complete usage of battery power are indicated by system icon colored red and the nodes that have been sent to sleep are indicated by system icon colored cyan.

The battery levels of the sensors working under an aggregator are given under the 'Battery log'. This log is present even in the regular sensor nodes to show neighbor battery values. The battery levels in the battery log allows to predict which system is going to be dead soon so that the aggregator gets to know beforehand about it and need not send any further request to that system thus saving energy and providing predictability for WSNs lifetime.

The User Interface (UI) also shows the BS IP under which the aggregator is working and also its own battery levels and the temperature sensed via track bars.

The system under sleep state won't be sent any requests until it becomes active in future rounds thus avoiding unnecessary drain of its battery power. Hence it makes easier for the aggregator to know that this system won't be available and hence the request need not be sent to it for further rounds thus providing predictability.


SEEDA - Node Application

Settings for Service Execution

☐ Sensor Device: [Communication Port]

Communications Port (COM1)

[IP ADDRESS IDENTIFICATION]

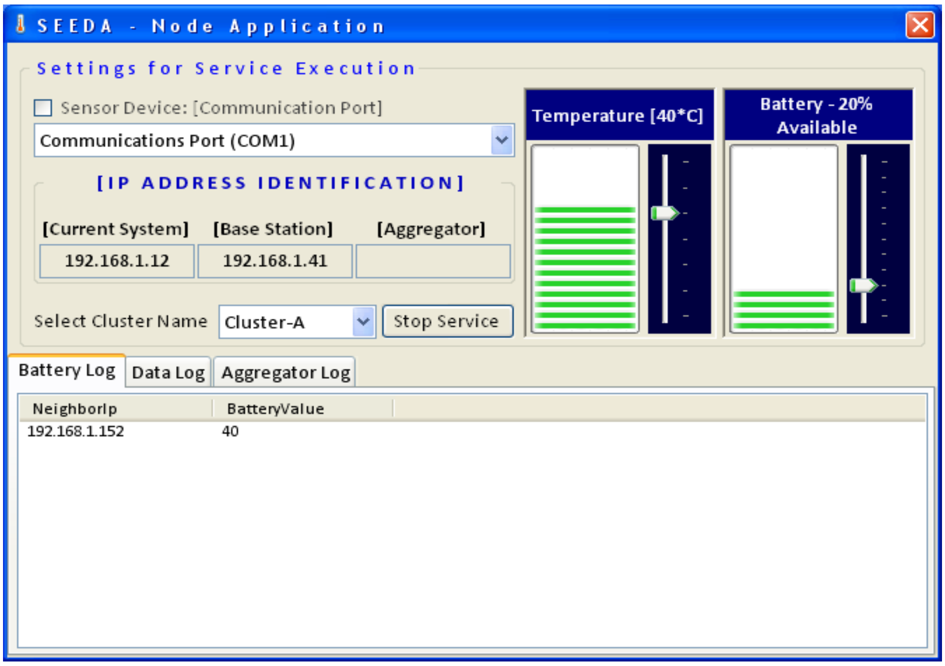
[Current System]	[Base Station]	[Aggregator]
192.168.1.12	192.168.1.41	192.168.1.152

Select Cluster Name: Cluster-A

ServiceType	ServiceIpAddress	ServiceTime
REQUEST-FROM	192.168.1.152	5:55:30 PM
DATA-TO	192.168.1.152	5:55:30 PM
REQUEST-FROM	192.168.1.152	5:55:41 PM
DATA-TO	192.168.1.152	5:55:41 PM
REQUEST-FROM	192.168.1.152	5:55:52 PM
DATA-TO	192.168.1.152	5:55:52 PM
REQUEST-FROM	192.168.1.152	5:56:03 PM
DATA-TO	192.168.1.152	5:56:03 PM
REQUEST-FROM	192.168.1.152	5:56:13 PM
DATA-TO	192.168.1.152	5:56:13 PM

Fig.2 Sensor node under Cluster A showing Data Log

Fig.2 above shows the 'Data Log' of application running as a sensor node under cluster A. The data log is common for all the sensor nodes including the aggregator. The data log shows the incoming and outgoing requests, their time and also the IP from which it is coming/going. This helps to know as to which node is active and participating in the data sensing and transfer.


SEEDA - Node Application

Settings for Service Execution

☐ Sensor Device: [Communication Port]

Communications Port (COM1)

[IP ADDRESS IDENTIFICATION]

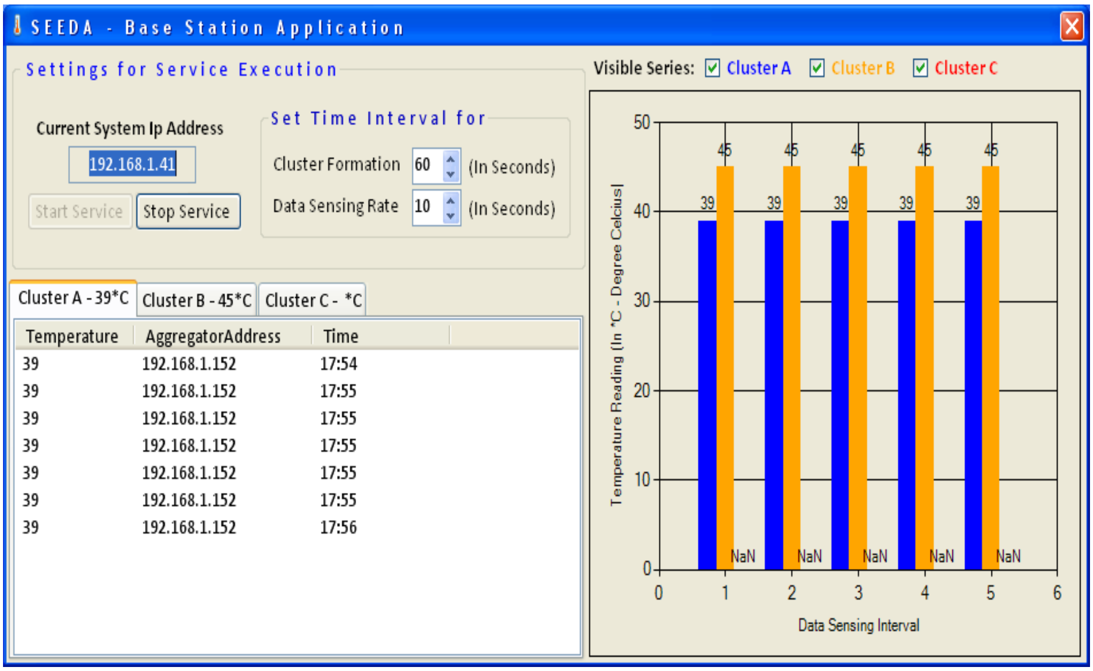
[Current System]	[Base Station]	[Aggregator]
192.168.1.12	192.168.1.41	

Select Cluster Name: Cluster-A

NeighborIp	BatteryValue
192.168.1.152	40

Fig.3 Sensor node under Cluster A showing Battery Log

Fig.3 above shows the 'Battery Log' which displays IP of neighboring sensor nodes that are active and their corresponding battery values. All the above results apply to all the aggregators belonging to different clusters running simultaneously under the BS.



SEEDA - Base Station Application

Settings for Service Execution

Current System Ip Address: 192.168.1.41

Set Time Interval for

Cluster Formation: 60 (In Seconds)

Data Sensing Rate: 10 (In Seconds)

Cluster A - 39*C | Cluster B - 45*C | Cluster C - *C

Temperature	AggregatorAddress	Time
39	192.168.1.152	17:54
39	192.168.1.152	17:55
39	192.168.1.152	17:55
39	192.168.1.152	17:55
39	192.168.1.152	17:55
39	192.168.1.152	17:55
39	192.168.1.152	17:56

Visible Series: ☑ Cluster A ☑ Cluster B ☑ Cluster C

Fig.4 Base Station showing Cluster Temperature Readings

Fig.4 above shows the base station application that monitors the function of all the aggregators and their respective clusters. The User Interface (UI) provides the user with a control to set the cluster formation and data sensing intervals that might be helpful for different scenarios.

The BS receives the aggregate temperature computed by the cluster aggregators from each cluster along with the time and the aggregators address. It then computes the average aggregate temperature for every cluster and displays it under the respective cluster log. The results obtained are plotted on a chart for the graphical view thus making it easier for the user to understand. This data also helps it to know which node is acting as the aggregator for which round. For every round the entire log of the previous round will be cleared to fill in the new data, however a log file of the data will be saved on the disk before the commencement of the next round for later study.

Thus we can conclude that our proposed system solution not only increases the lifetime of HC-WSNs by making energy savings at cluster head selection phase, data sensing phase and data aggregation phase but also provides predictability for the same with HEF algorithm. Also, with the use of a strong homomorphic encryption data aggregation scheme like Paillier's, it imparts strong data security solution especially at the aggregator nodes which contain valuable data and are more vulnerable to intruder attacks.

VI. Conclusion and Future Work

A novel attempt to achieve maximum network lifetime by reducing energy consumption at the cluster head selection phase and at the data communication phase is introduced. A hierarchical cluster head selection algorithm which selects a node with the highest residual energy as the cluster head in every round is implemented. The HEF works on the residual energy of the sensor nodes which is gathered in priori in every round for cluster head selection. Thus it provides better predictability and utilization of battery for the sensor nodes. The threshold battery value calculation for every round and sending those nodes below this threshold battery value to the sleep state for later resumption prevents unnecessary battery consumption leading to premature death of the nodes. Further, a homomorphic encryption scheme to provide secure transmission of the sensed data using Paillier Cryptosystem is implemented. This not only provides the much needed security to the sensed data from eavesdroppers and intruders but also allows encryption to be done on the cipher text eliminating the need to decrypt/encrypt at every node through which it is transmitted. Thus further reducing the energy consumed for encryption and decryption at every node.

The proposed solution is attempted for a single hierarchical network consisting of fewer systems. As a future enhancement it could be designed for larger networks with multiple hierarchies which could help in the reduction of energy consumption on a large scale for WSNs.

References

1. D. J. Baker and A. Ephremides, “The architectural organization of a mobile radio network via a distributed algorithm,” in IEEE Transactions on Communications COM-29(11), 1981.
2. G. V. Crosby, N. Pissinou, and J. Gadze, “A framework for trust-based cluster head election in wireless sensor networks,” in DSSNS 2006 : Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems : Proceedings, Columbia, Maryland, April 24-28, 2006, sponsored by IEEE, NASA, IEEE Computer Society.
3. L.-C. Wang, C.-W. Wang, and C.-M. Liu, “Adaptive contention window-based cluster head election algorithms for wireless sensor networks,” in VTC-2005-Fall. 2005 IEEE 62nd, September 2005, vol.3.
4. S. Sivavakeesar and G. Pavlou, “Associativity-based Stable cluster formation in mobile ad hoc networks,” in Proceedings of IEEE Conference on Consumer Communications and Networking Conference (CCNC2005), January 2005, IEEE.
5. E. Chu, T. Mine, and M. Amamiya, “A data gathering mechanism based on clustering and in-network processing routing algorithm: CIPRA,” in The Third International Conference on Mobile Computing and Ubiquitous Networking, ICMU, 2006.
6. W. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “Energyefficient communication protocol for wireless microsensor networks,” in System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on Publication Date: 4-7 Jan. 2000, vol. 2
7. W. Wang and A. Jantsch, “An algorithm for electing cluster heads based on maximum residual energy,” in Proceeding of the 2006 international conference on Communications and Mobile Computing, July 03-06, 2006.
8. H. Huang and J. Wu, “A probabilistic clustering algorithm in wireless sensor networks,” in VTC-2005-Fall. 2005 IEEE 62nd, September 2005, vol. 3
9. O. Younis and S. Fahmy, “Distributed clustering in ad-hoc sensor networks: A hybrid, energy-efficient approach,” in In: Proceedings of the IEEE Conference on Computer Communications (INFOCOM),Hong Kong, 2004.
10. E. Hansen, J. Neander, M. Nolin, and M. Björkman, “Energy-efficient cluster formation for large sensor networks using a minimum separation distance,” in In The Fifth Annual Mediterranean Ad Hoc Networking Workshop, Lipari, Italy, June 2006.
11. Bo-Chao Cheng, Hsi-Hsun Yeh, and Ping-Hai Hsu, “Schedulability Analysis for Hard Network Lifetime Wireless Sensor Networks With High Energy First Clustering”, IEEE TRANSACTIONS ON RELIABILITY, Vol. 60, No. 3, September 2011.
12. Xiaoyan Wang , Jie Li, Xiaoning Peng And Beiji Zou,”Secure And Efficient Data Aggregation For Wireless Sensor Networks”, IEEE Seventh vehicular Technology Conference Fall , 2010.
13. C.Castelluccia, E.Mykletun, G.Tsudik,”Efficient Aggregation of Encrypted data in wireless sensor network”, in: Proceeding of the conference on mobile and Ubiquitous System: Networking and Services, 2005.
14. D. Westhoff, J. Girao, M. Acharya,“Concealed Data Aggregation for Reverse Multicast Traffic in Sensor Networks: Encryption, Key Distribution and Routing Adaptation”, IEEE Transactions on Mobile Computing, Vol. 5, No. 10, 2006.
15. Shih-I Huang, Shiuhpyng shieh, J.D.Tygar,“Secure encrypted data aggregation for wireless sensor networks”, Springer, 2009.
16. L. Hu and D. Evans, “Secure Aggregation for Wireless Networks,” Wksp.Security and Assurance in Ad Hoc Networks, 2003.
17. B. Przydatek, D. Song, and A. Perrig, “SIA: Secure Information Aggregation in Sensor Networks,” SenSys '03: Proc. 1st Int'l. Conf. Embedded Networked Sensor Systems, New York: ACM Press, 2003.
18. H. Çam et al., Sensor Network Operations, Wiley, 2004, ch. “Secure Differential Data Aggregation for Wireless Sensor Networks”.

AUTHOR(S) PROFILE

Ms. Bhavana D, has received her B.E degree in Computer Science & Engineering in the year 2013 from Visvesvaraya Technological University, Belgaum, Karnataka. She is currently studying as a PG Scholar pursuing 4th Semester M.Tech in Computer Network & Engineering in The National Institute of Engineering, Mysore, Karnataka.

Mr. C. N. Chinnaswamy, is an Associate Professor in the Department of Information Science & Engineering at The National Institute of Engineering, Mysore, Karnataka. He has received his M.Tech from VTU and B.E. from University of Mysore. He is currently pursuing his Ph.D. His teaching and research interests are in the field of Computer Networks and Internet Communication.

Dr. T. H. Sreenivas, is Professor in the Department of Information Science & Engineering at The National Institute of Engineering, Mysore, Karnataka. He has received his Ph.D. from IIT, Madras, M.Tech from IIT, Kanpur and B.E. from University of Mysore. His teaching and research interests are in the field of Operating Systems, Networking, Schedule Optimization and Wireless Sensor Network.