International Journal of Applied Science and Engineering

An adaptive multilevel location based key management system for dynamic wireless sensor networks

Venkatesh Arumugam*, Asha Seshasayanam

School of Computing Science and Engineering, VIT University, Chennai Campus, India

ABSTRACT

A Key management system plays an important role in the process of wireless communication between the nodes of a Wireless Sensor Network (WSN). Unlike the wired networks, WSNs are more vulnerable to attacks from the malicious nodes. To overcome the shortcomings of the existing key management systems, this paper proposes an adaptive multilevel location-based key management system (AML-KBS), in which the keys are generated dynamically and shared among the nodes of the wireless networks. Since the proposed approach follows a location-based system for key management, the attackers can be differentiated based upon their location. Also, the proposed system has proven its withstanding against node capture attacks. Comparing with the existing approaches, the memory requirement of the proposed system has shown better improvement. Moreover, the proposed methodology provides better security mechanism than the existing key management systems.

Keywords: Key management, Wireless sensor networks, Node capture attack, Local node attacks.

OPEN ACCESS

Received: September 4, 2020

Accepted: November 29, 2020

Corresponding Author:
Venkatesh Arumugam
venky.jec@gmail.com

Publisher:
Chaoyang University of Technology
ISSN: 1727-2394 (Print)
ISSN: 1727-7841 (Online)

1. INTRODUCTION

Wireless Sensor networks (WSN) are collection of nodes which are called sensor nodes (SN). They have batteries as energy source and having limited computing power with lesser storage capabilities. WSN are useful in many important applications such as automation process of many of the residences and commercial applications. They possess the capability of communicating shorter distances with which they can send or receive small amount of data. Since the sensor nodes may contain very crucial information and are vulnerable for various attacks by spurious attackers, it is very essential to secure the access of the nodes from the attackers. For instance, any of the spurious node can observe the wireless communication between the sensor nodes and they can get the information that is been shared. So, it is really important to follow some kind of crypto-oriented approach to change the form of the information that is exchanged among the nodes. But, it is very much equally important to share how the encrypted information can be accessed by the intended wireless sensor nodes. Here comes the need of securely exchanging the secret keys between the nodes before the secure communication begins (Chan et al., 2003).

Many of the WSN currently in use are following the concept of secret key cryptography, in which any of the sensor nodes that are participating in the communication are using same key for crypto-conversion of data and also for authenticating the communication. The keys that is used for the purpose can also be called as symmetric key and the process of sharing them amongst the participating sensor nodes is called as key management (Lee et al., 2007; Zhang and Varadharajan, 2010). Another challenge faced by the sensor nodes is lack of prior knowledge about the topology and structure of the network before communication begins. It can be overcome by placing the key inside the sensor node prior to the deployment of the nodes. Moreover,

it is very important to choose a key for the adjacent nodes of a wireless network which would share the communication messages. Even though sharing between two neighbor nodes is not precise, they can be shared among the nodes that are present in a particular path called key path. During the design phase of the WSN, it is very crucial to determine the length of the key path, so that the performance of the network can be assessed easily (Eschenauer and Gligor, 2002). But this kind of key distribution during the design phase can only be used in static WSNs. In mobile or ad-hoc WSNs, they are not helpful. Also, it is very much space consuming, if each of the nodes are storing the keys of the neighboring nodes. If the base station is given with the responsibility of distributing the keys to the sensor nodes, it will create another problem of overloading communication channel. So, the alternate approach can be following a public key cryptosystem or asymmetric key cryptosystems. Even though public key crypto systems proved to be more secure, they pose new problems such as requiring computational overhead as well as high energy utilization (Gura et al., 2004; Wang and Li, 2006). So, bootstrapping as well as distribution of keys by base station are posing problems in key management.

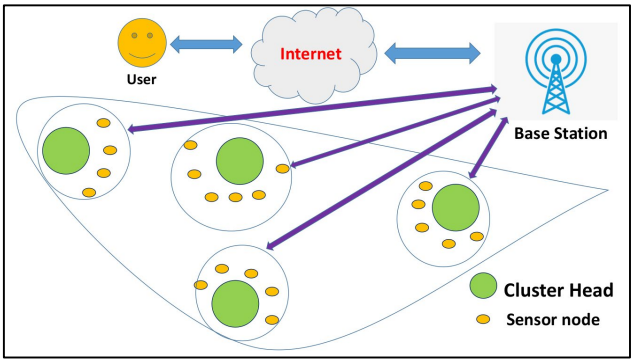

Fig. 1. Basic architecture of a WSN

In Fig. 1, basic architecture of a wireless sensor network has been shown. Base station controls all the nodes that are associated with a cluster. The group of nodes that constitute a cluster are having a node as their cluster head. All the other nodes which are coming under the control of a cluster head are called as sensor nodes. Basically the WSN adapts the OSI model, which consists of five layers (Application, Network, Transport, Data-link and Physical layers) for data processing and transmission. In addition, a WSN has three cross layers which takes care of the power, mobility and task management. It is essential to follow the layered approach to ensure the combined operation of various sensor nodes to improve the performance of a sensor network.

There are mainly two basic kind of key management schemes. The problems of bootstrapping can also sort out with the help of either a plain global key (PGK), in which all the sensor nodes are sharing a same key or full pairwise keys, where each node possess the key of its neighboring node (Simpl´ıcio-Jr. et al., 2010). The former approach does not provide the expected level of security at all over the open sensor network, whereas the later cannot be applied in larger networks even though it is comparatively more secure.

The data about each of the sensor nodes and their positions are very crucial in determining the approach to be followed in sharing the keys among themselves. So, it is mandatory to design a key management system with the help of the location information of the sensor nodes. In many of such approaches, the grid-based approaches are helpful in determining the position of the nodes and appropriately sharing the keys based on the placement of the nodes. But in certain critical applications which are related to national security, we cannot rely on the grid-based location mapping of the sensor nodes. Also, this issue can be overcome by hiding certain crucial implementation information (Anjum, 2010).

There are approaches which provides access to a collection of keys in a key repository, from which the keys will be shared among the participating sensor nodes. They established a chain of all existing keys that are applicable for the particular node that can be designed for secure data transfer between them. But the main issue with this approach is determining the size of the key repository or the length of the linked key list to accommodate all the sensor nodes in a specific location. Because, the size of the linked key list is difficult to maintain because of the storage restriction in the sensor nodes. Also, if the size of the repository gets increased, will obviously reduce the possible sharing among a pair of sensor nodes. Also, the sharing of keys between two nodes are basically affected due to the increase in size of the repository (Eschenauer and Gligor, 2002).

Alternatively, thousands of keys can be distributed in a randomized way during the manufacturing of the nodes, before they get implemented real time. Out of the available keys in a repository, a certain number of keys are chosen randomly to assign to any of the designed sensor node. The collection of the keys that are assigned to any pair of nodes are being compared among themselves (Merkle, 1978).

2. RELATED WORKS

Many more research works have been carried out for improving the security while key sharing among the sensor nodes. In this section, various key management schemes are briefly discussed along with their shortcomings in perspective of security and efficiency. Some kind of protocols or rules must be followed while exchanging key of the participating nodes during communication to ensure security. It is very important to handle the problems in key management such as creating appropriate keys, sharing the keys among the nodes and implementing encoding and decoding operations (Abdollahzadeh and Navimipour, 2016).

As discussed in the introduction part of this paper, there are primarily two ways of managing keys in WSNs. They

are known as static and dynamic key management. The keys are fixed and won't get changed once they have been assigned with a pair of sensor nodes, whereas they may get changed in course of time if they were created using any of the dynamic key management. Comparing with the dynamic schemes, static keys are more vulnerable to network attacks (Afsar and Tayarani-N, 2014). Since the nodes' assigned dynamic keys keep on changing, the lifetime of the network as well as the confrontation against the network attacks are improved quite a lot. So, comparing the two approaches existing, it is always better to follow a dynamic approach for key management in a wireless sensor network. Many of the algorithms which are focusing on key management are presented in Bekara and Laurent-Maknavicius (2009). From Bekara and Laurent-Maknavicius (2009), we understand that the key management systems can be categorized based on the types of keys, ways of estimating keys, ways the generated keys are shared and how the coding is performed over the generated keys.

In Ferng et al. (2014), a multilevel dynamic key management scheme has been discussed. This can be applied for the different models in the WSNs such as hierarchical model as well as the peer–to–peer models. Here, in this paper, the authors have discussed an approach which consisted a private key, public key combination for securing data. The distribution has been handled by a certification server which is a centralized one and all the nodes are able to access the server. To ensure the security, the sensor node that uses the certification server receives the identifier of the node and it accesses the key repository to access the private key of its counterpart it wants to establish communication. The keys are created and distributed prior to the establishment of the wireless network. The main disadvantage of this approach is the minimum level of security achieved despite less amount of energy is required for the processing. But there is a provision for improvement under this approach when similar keys are used at both the end of the communication with the cost of higher energy consumption.

Node cryptanalysis and adversary crypto attacks are defended with an efficient key management system which ensures the secure communication. There are many approaches which handles hierarchical model-based networks in which the neighbors of a particular node are in part of communication. An efficient key management system for hierarchical networks as shown in Fig. 2 has been proposed with Thevar and Rohini (2017). In the proposed approach, the group of nodes have been categorized as sectors. For every group of nodes, a node has been identified as a leader for the group. The nodes in each group are sending and receiving messages with each other members of the same group. Similarly, each of the group members are combined and viewed as a sector, whereas sector of one group is communicating with the sector of another group of sensor nodes. The nodes are also allowed to move from one sector to another as well as another group. Whenever such a movement takes place the key values that are associated with that particular moving node are removed from the repository correspondingly. This has been accomplished with the help of an angular function and the group in which the node is present determines its private key.

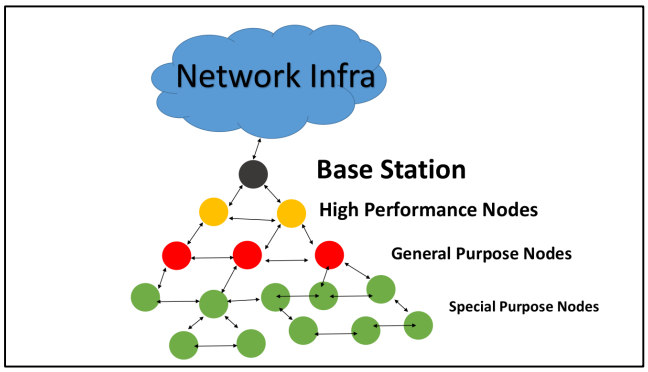

Fig. 2. Hierarchical wireless sensor network model

The position of the node in movement at each of the time frame has been followed up by the leader of the group using a data structure such as a table. When it finds some node is moving outside the group or the sector it updates the table with the new position of the node. Also, this table can be very useful for protecting outside nodes moving inside a sector. The key assignment is done only during a node gets joined in the group. Also, the communication is happening between the adjacent nodes, it is evident that the energy required for communication has been greatly reduced. All the operations corresponding to any of the group or the sector are handled by the leader of that particular group.

A novel key management scheme which does not rely on the certificates for secure communication has been proposed in Seo et al. (2015). The authors have named the proposed approach as a Certificate-less Effective key management protocol which primarily focuses on the WSNs which are classified as dynamic ones. It provides the relaxation for the nodes to roam inside the network. It can be considered as a heterogeneous and hierarchical model which uses pairwise keys for asymmetric encryption, a node key, communication key and a key which supports communication within a group. The key values are expected to be modified often to withstand against code analysis and cryptanalysis. Considering a situation in which a node is been captured by the attacker which also get the information about the group keys and pairwise communication keys of the particular node, the encoding operation of the captured node will prevent cryptanalysis of the attacker. So, the attacker loses the chance of disturbing the communication between any other pair of nodes inside the group or the network.

In Huang et al. (2013), a method which primarily focuses on the protection of data in sensor network which protects the storage network based WSNs has been proposed. This network has been called as privacy enhanced one, as it improves greatly the safety of the data being stored. Here, the network has been scattered with groups or clusters of square structure. This approach supports assigning keys for

the communication before the nodes are distributed throughout the network. In this system, there are a combination of master key, pairwise key, cell key, row key and global key are used to achieve data security (Huang et al., 2013). In this approach, the received data are always placed inside a cell which is called as storage cell. If some changes are observed by the sensor node at any point of time, it clearly identifies the particular storage position with the help of a hash mapping function. The cell key plays an important role of making the received data encrypted and the encrypted data has been stored in the identified storage cell. The same process has been carried out whenever a particular data stored in some cell is required by any of the processes. The main issue with the proposed approach, it is very clear that during the process of storing and retrieving the data in the cell, the communication channel has been overloaded. This will lead to wastage of energy and greatly reduces the network lifetime (Huang et al., 2013).

A multitier key distribution has been proposed in Annapurna and Siddappa (2015). The main focus of the proposed solution is to ensure security when a data broadcasting is done to a particular group of sensor nodes. Also, the proposed system withstands the active attacks on the WSNs such as Sybil attacks with the help of the multitier structure. The keys are shared between within clusters are handled by the top layer of the system. When a particular node is part of a group, it will always receive a data and forward the same towards the head of the group. The next level ensures secure data transmission from a sensor node of one group to another node of another group. Also, the system ensures one additional level of encryption at each node of transmission in the path to the storage. The bottom most layer protects the system from rekeying, a situation happens when the node itself is improperly using a key. If there are two different groups, say A and B are communicating, a key k_i that has been shared by them has to be identified as $k_i = K_A \cap K_B$. So, whenever a node initiates communication the node requires key k_i (Annapurna and Siddappa, 2015).

There are some approaches which combines the sharing of keys before and after deployment of the network. Such a system can be found in Erfani et al. (2015). In this system, the primary requirement is at least one secret key must be shared among the communicating sensor nodes. Whenever a sensor node is included or removed from a network, devising a new key has been explained in this approach. In this system, the keys generated before and after creating a network are maintained in two different areas of the storage space of the particular node. A sensor node is created initially with a secret key before it gets deployed. That key will be compared with pre-deployment key of the node that is to be communicated. If they are not similar, then the keys after deployment are generated. Another important feature of this approach is assigning keys again after creating and deploying a node. But this system restricts the maximum number of nodes that can be connected with the network at one point of time. Advantages of this approach includes easy to implement and the level of secrecy it provides.

Implementation of a key management system greatly affects the power consumption associated with the sensor node. So, it is important to review the approaches, which are conscious on energy efficiency. One such a system that focuses primarily on the energy efficiency has been proposed in Messai et al. (2015). In Energy aware Hierarchical Key management (EAHKM) approach as proposed in Messai et al. (2015), which uses same key for both encryption and decryption. Also, this approach combines both the key distribution approaches before and after deployment of sensor nodes. The proposed system protects the cluster of nodes from the attackers, with a novel cluster establishment process that contains two distinct steps namely the key pre-distribution and key generation and cluster formation. The first step has been accomplished even before the development of the sensor nodes, where the keys are loaded into the storage part of the sensor nodes. One of the keys are used to establish connection between the sensor node and the base station whereas another key which is shared among all the nodes that forms a cluster. The later key is deleted immediately after the cluster has been created. Since this approach can be useful in saving energy during communication, this could be useful in many of the real time applications. But it has a disadvantage of possessing a network key which can be attacked and the whole key management system will be in danger. Also, since this approach is very useful in hierarchical models, in certain scenarios, the distance from a node to the base station may be extremely high which will require higher energy to send and receive messages to and from the deeper sensor nodes respectively.

A hierarchical key management approach has been proposed in Zhang et al. (2017). This approach cannot be applied with dynamic networks. It is assumed that all the sensor nodes are having similar amount of processing capacity, storage and power. Using pre-deployment scheme, the key has been loaded in the memory of every sensor nodes. These keys are only useful during the sector establishment in a network. Once a sensor node has been included in a sector, then the corresponding key would be removed from the memory of that particular sensor node. Here, the number of intermediate nodes present in any cluster determines the message availability of any sensor node in the cluster. In the rekeying process the sector head node inside a cluster has been chosen and a new key has been identified.

Another key management system that concentrates on energy efficiency has been proposed in Chakavarika et al. (2017), which could be considered as a dynamic key management approach. The proposed scheme follows hierarchical network model. Keys are shared between the sectors are implemented in this approach. The sensor nodes and the base stations are also sharing pairwise key, in addition to the cluster members. Initially, a pre-distribution key has been generated with the help of the bivariate

polynomial which will further be distributed to different sectors. The values extracted from these bivariate polynomials are placed in the memory of the sector head nodes. Based on this key value, the members of the sector are assigned with the private keys which will be stored in their memory later.

One of the most important approach that follows a master key based approach has been discussed in Zhu et al. (2006). This approach uses a transitory master key based approach. This key will not be used throughout the distribution process, only during the initial distribution period instead. So, it will get deleted, once the nodes were utilized them for a connection. When the nodes are moving into subsequent operations, whereas they can be utilizing the resources to function in a sector and send and receive messages as well as data. LEAP+ scheme has been proposed in Zhu et al. (2006) that explains how the transitory master key approach can be helpful in providing security during communication. The different keys used in LEAP+ scheme includes: individual, global, cluster and pairwise key. Each kind of key is assigned to different component of a sensor network. Based upon the master key and a random function chosen by the network dynamically, the network activities are determined. The random function is applied with the master key of a sensor node, and its private key has been obtained. The keys that are generated have been propagated towards the neighbor nodes, which in turn will generated their private keys by following the process as discussed earlier. The keys that were created by the intermediate valued ones are removed at the end of the process and the final obtained key will be retained by the storage of the sensor node. But, the main disadvantage of this scheme is, the network has been assumed to be static to apply this scheme for a particular network. In dynamic networks, this approach will lead towards overhead in terms of maintaining the key information and message communication. No additional knowledge can be implemented other than the very basic information about the members of a particular sensor network is possible using this approach.

Conversely, in Gandino et al. (2016), the knowledge deployment has been included which are implemented prior to implementation of the network in real time. Pairwise keys in this approach were distributed in the earlier stage of distribution, since the adjacent nodes may be sharing the keys with the assumption that they are neighborhood nodes. During the movement of the nodes, after the keys were distributed, the position of the nodes have been stored and they are maintained in the memory of the sector head node. Also, their earlier keys would be deleted by the corresponding sensor nodes.

3. PROPOSED SYSTEM

In this paper we have proposed a novel approach for secure key management approach which is named as Adaptive Multilevel Location based AML-KBS that implements an effective key management system which ensures the security of node to node communication in a wireless sensor network. The different types of keys provided by the proposed system are: an asymmetric key set, a sensor key, a neighborhood key and a sector key.

3.1 Construction of a WSN

In this section, the process of constructing a WSN which is dynamic and dissimilar sensor nodes has been explained. In this network, the sensor nodes that are fixed and moving ones are controlled by a base station (BS). This BS is also responsible for collecting data and messages from various sensor nodes associated with it. Some of the sensors which are capable of high performance (S_h) and some others are showing low performance (S_l) in the network. There are T total sensor nodes are present in the network. T_1 is the total S_h sensors and T_2 is the total S_1 sensors where $T_1 \ll T_2$. Since the network we are considering dynamic, the nodes are essentially added and removed over the network due to the nature of the real time sensor network. The entire network has been subdivided into sectors and in our case, S_h sensors are acting as the sector heads and the remaining S_l sensors are the members of the sectors. Every sector has a direct connection to the BS of the network through the connection established from the path manipulated by high performance sensors as shown in Fig. 3.

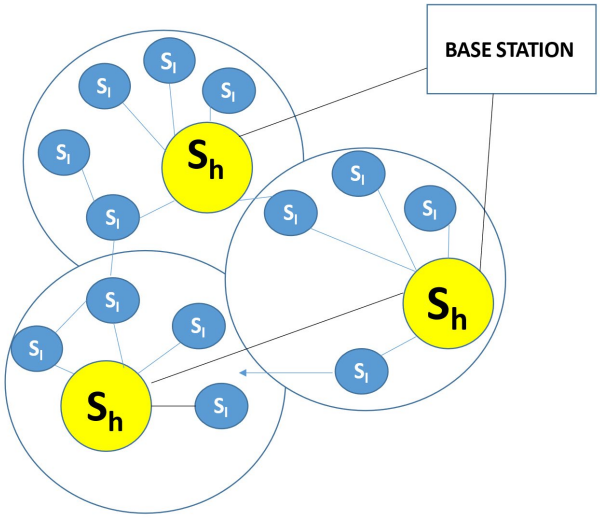

Fig. 3. An example WSN with high performance and low performance sensor nodes

Every S_h sensor is trying to create its own sector by trying to identify the S_l sensors which are physically present near to the S_h sensor by sending and receiving beacon messages around the network. Every addition and removal of a S_l sensor will be updated periodically with the BS by sector head sensor node. It is the role of BS to maintain a data structure called nodelist (nl) which possesses the information about the nodes that are present currently with the sector which are considered to be genuine.

The data structure of BS has a status field that maintains the status of every sensor node newly registered. If any of node is left from the network or it is found that some nodes are not genuine, then the status of the corresponding sensor nodes are updated. To identify a sensor node uniquely the BS node list assigns a unique ID for each of the sensor nodes. Every base station contains a Centralized Key Manager (CKM) that holds responsibility for creating values required by BS key sharing process and generates asymmetric key sets for every sensor node pairs of the network. The BS and the sensor nodes are sharing a sensor key and all the nodes of a particular sector are sharing a key called as sector key.

3.2 CKM Structure

In the proposed system, as mentioned earlier, the CKM generates and shares four different types of keys for enabling secure communication between nodes of the sensor network as shown in Fig. 4. The asymmetric key set has been generated by the CKM of the base station before the node is being actually implemented. Every node pairs can be using this key set for encryption and decryption of messages and data between them respectively. Every sensor node is sharing a unique key with the base station which is known as sensor key. For instance, a node S_h can be using its own sensor key while sending a command to one of its S_l or to the base station. Since the base station itself has a sensor key, it can use the same for encrypting any important command message to be communicated with any other sector heads or the nodes directly. Every sensor node of a network could get one individual unique sensor key from the base station CKM before it starts sending or receiving data.

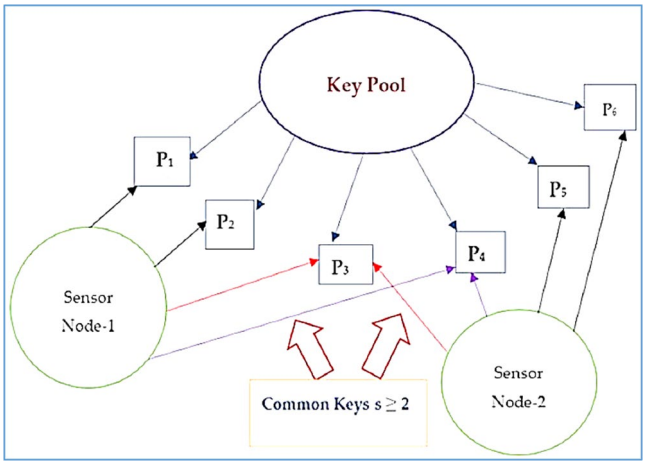

Fig. 4. Overall architecture of key management

The neighborhood keys are exchanged between the sensor nodes which are physically adjacent for ensuring the intended neighbor and protect their data from adversary nodes. So, in a normal scenario, a S_l shares its neighborhood key with some S_h to join in the sector headed by S_h. With the acquired neighborhood key, a sector key could be generated by S_h and can be shared with the S_l node which recently joins into the sector. So, S_l can use this key for sharing sensitive data with its sector head secretly. Whenever a common message is to be shared among all the nodes in the sector, a sector key would be used by the sector key. Whenever a sector head S_h wants to communicate any important command with all the sensor nodes of the sector, such information would be encrypted by S_h using the sector key which is unique for each of the sectors.

3.3 Sensor Network Design

This subsection explains the process of designing the sensor network based on the information that a base station possesses. The network design process begins with addition of a new sensor node in the network by updating the list that is maintained by the base station. The base station assigns every new node with unique identifier S_{L_i} or S_{H_j} before it updates the node information in the node list. For instance, the S_l nodes are having identifiers as mS_{L_i} and S_h nodes are assigned with $mS_{H_j}$where $1 \leq i \leq T_1$ and $1 \leq j \leq T_2$ while $T = T_1 + T_2$. The base station chooses any node S_{L_i} and the corresponding key value can be calculated by $S_{L_i} = y_{L_i} Q$, where Q is a random number and y_{L_i} is the private key of the asymmetric key for the selected node. So, the asymmetric key for a particular sensor node has been calculated by the CKM by using the public and private keys $(\mathrm{K_{S_i}}, \mathrm{r_{S_i}})$ as follows:

$$K_{S_i} = y_{L_i} Q$$

$$r_{S_i} = y_{L_i} + h_0(L_i, y_{L_i}, Q_{L_i}) \ mod \ n$$

By following the procedure, the node list has been updated whenever a new node has been included in the network. So, at any moment the node list contains the identifiers of the sensor nodes and their corresponding key values generated. Before implementation of the network, the keys generated by the above explained process has been stored in the memory of the sensor nodes and the list has been updated accordingly.

Once the node has got a place in a particular network, it has to look for its adjacent sensor nodes throughout the sector with their location information. So, they will make the asymmetric key pair can be generated among the neighborhood nodes adversely. The sensor nodes are broadcasting a message with their identifier and public key to create the private key pairs (Seo et al., 2015). The key pairs can be used for the encryption process of the data that was obtained in each of the sensor node. Consider two nodes P and Q. Let us assume that node P receives a message from Q, then the asymmetric key K_{PQ} can be generated with the following process as explained in Seo and Bertino (2013). The generated key has been encapsulated in the node's memory as $\theta_P = (X_P, Y_P)$. The value of $m_P \in_R Z_q^*$ has been chosen for finding out $X_P = m_P R$, where R is a random generator. Then the neighborhood key has been

calculated as $K_{PQ} = l_1(S_P, T_P, m_P \ . R_Q, Q, R_Q)$. A neighborhood key pair K_{PQ} has been obtained after the nodes P and Q are generating an authentication hash code. The hash code will then be regenerated to verify the authentication of the nodes.

After this process, each of the nodes must be classified into any of the member of the sectors. This sector creation has to be taken care of the base station as well as the sector head sensor nodes. Each of the S_h nodes are sending beacon messages to identify the S_l nodes which are adjacent to each other and then they are authenticating the nodes with the process explained earlier. The S_h sensors are formulating a sector based on the S_l nodes which have the authentication properly. For every sector node S_l, the sector head S_h maintains a sector key and the key has been shared with the nodes falling under same sector. A sector key $SS\ \ Ki$ has been created by the head and share with the sensor node.

3.4 Sensor Node Movement-Removal

It is very essential for S_h of a particular sector to maintain the sector key, when the sensor node of one sector travels to another sector. Whenever this happens, it is the role of S_h to make note of the movements and report the same to the base station, so that the BS makes necessary changes over the node list it maintains security in front and reverse direction. We can say that a node in movement can be represented as S_m. There are various reasons for the sensor node may move from one sector to another, that includes the disruption in the link, the sensor node drop etc. Some of the node movements are intentional. The situation where the sensor node S_m can be intimating its willingness to move from one sector to another, so that the sector head can update the status of the particular node accordingly. In this particular scenario, the sector head has to intimate the movement with the base station with a message. Otherwise in some other cases, the node movement may happen due to the connection failure between S_h and S_m. This situation may arise, if the S_m node may power off or controlled by an attacker. If a particular timer value expires, till then there is no reply from S_m node, the sector head S_h can assume that the sensor node could have been moved out of its sector or unreachable. So, S_h has to update about the node lost information with the base station with the error message by specifying the node identifier in the message. Once it is found that some S_m in its sector, S_h will have to recalculate sector key immediately. And it will send the updated sector key to all the S_l nodes within its sector currently present. So, all the S_l nodes will decrypt the message with their asymmetric counterpart and update their sector key accordingly.

3.5 Sensor Node Movement-Inclusion

In most of the cases, S_m will be included in some neighborhood sectors once it intentionally moving out of its current sector. So, S_m has to ask the S_{h1}, for instance, to get added as part of the new sector region. Once S_{h1} receives the information about the newly joined sensor node, it must recalculate the asymmetric key pair for its own sector and intimate the inclusion of new node to the base station with a control message. The base station will decide based on the message received which includes the sensor key of S_h about the genuineness of the newly joined node to be included as part of the sector of S_{h1} or not. The node list is also updated accordingly. Now S_{h1} will receive positive reply from BS and it will update sector key with all of its current members. If the BS identifies that the new member just now added is not a genuine one, it has to intimate the same to S_h and will reverse the asymmetric key generation process, and existing keys are restored.

When the S_m node again comes back to its earlier sector, S_h will look for the timeout value t_v maintained for S_m which started by the moment it leaves the sector. If t_v expired, S_m is considered as the new node, otherwise the earlier key values are restored.

3.6 Location Based Node Management

In the location-based node management, we have devised an approach that can enable the sector head to assign the key for S_l based on their location in their sector. The main advantage we have achieved from this system, is its withstanding against the malicious nodes and adversaries present as part of the sector members, which is tedious to detect. As the earlier discussion suggests, the keys may be distributed prior to the implementation of the WSN. In the proposed system, the hashing approach has also been adapted to implement key management. The various keys present in the sensor node include sector key, sensor key and a hash function and the positional information of the sector. The neighborhood nodes are identified by sending a broadcast beacon message and waiting for the response from other nodes. The positive reply from the adjacent nodes are containing the sector positional information that helps recording the location of the sensor. Every time the node S_h receives a positive response from any S_l nodes, a sector has been framed with location information (Choi et al., 2018). Whenever a communication begins, the encrypted form of location information has been shared with the sender and the receiver nodes. Using the hash function the node position in the sector can be managed. All the keys used by this approach are using any authentication algorithm for authenticating their genuineness.

In WSN, there is always a danger by a compromised node in a completely secure network which exactly mimics the operations of a normal node. The attacker could use the compromised node to access the network information (Choi et al., 2018). This kind of insider attacks should also be addressed. The process begins with inclusion of a node in the network after it sends acknowledgment for the beacon message by S_l. For every new node added, the base station usually send a confirmation message after it finds that the

new node S_l is authenticated one and assigns with an identifier. Also during the node insertion process, the corresponding sector key has been updated securely with the help of the sensor key and the hash code function. An efficient data forwarding protocol that is secured by the network key is essentially implemented in the sectors to identify the best path between the sensor node that sends data and the sector head node. Since the sector key was not shared with all the nodes on the network, and shared only with the n number of neighborhood nodes that are part of the sector that comes under a sector head node S_h. So, considering any of the path from the node to the sector head has a maximum of n hops. In our proposed approach, the protection against the compromised node S_c can be implemented as follows. Every data segment is initially assigned with a serial key value to identify its position while transmitting data. Every time the data passes a new sensor node, the serial key value of the corresponding data has been updated with the encryption of new sensor key information. A cut off limit has been maintained to limit the serial key value, and if the value goes beyond the cut off limit then the serial key will be removed from the registry. Also, the key value has been verified by a data structure maintained by each of the sensor nodes. Once the value does not change for a certain period of time, then it will be deleted. Then the information about deletion of the key has been informed with the node actually initiated the same. The message for enabling the node to regenerate the serial key has been once again forwarded.

Alternatively, the sensor nodes may be placed in any of the pre-determined location in the sector to retrieve the location information effectively. The sector can be split into fixed number of cells based upon the size of the sector earlier decided by the base station. The cells are later assigned with the newly deployed node which will facilitate the security and performance without any extra operations being carried out. The security provided by the above said approach could be assessed with the probability of attack P_{att} which is found very little value. It can be defined as follows:

$$P_{att} = \sum_{j=r}^{l} \left(1 - \left(1 - \frac{l}{|Q|}\right)^{y}\right)^{j} \frac{q(j)}{j} \quad (1)$$

In the above mentioned Equation (1), $|Q|$ is the size of the serial key number collection, l is the size of the key and $q(j)$ is the chance of a possible link creation in hop-by-hop manner from the sensor node to the sector head that contains j number of serial keys and $q = q(r) + q(r+1) + \cdots + q(l)$. The conditional probability that a particular node can be compromised, if the key set was accessed by an adversary would be,

$$P\{M|D_y\} = \sum_{\forall k} P\{M_k|D_y\} = \sum_{\forall k} P\{m_k|m\}P\{E_k|D_y\} \quad (2)$$

In Equation (2), M_k represents when a link with the serial key k has been captured. m represents the key has protected a particular link between the sensor node and the sector head. m_k is any link that possess the key k. E is some action that protects a particular data block from the adversary. D is any action in which y number of data are compromised. M_k is an action that causes the key k being compromised.

4. SECURITY ANALYSIS

In this section, we discuss about the level of security that is achieved from the proposed system and comparing the same with some of the existing similar approaches. The proposed system achieves secrecy and cannot be compromised by adversary nodes. The level of encryption used in the proposed system ensures that the data cannot be accessed from any attackers.

Consider the following scenario: a node S_{l_c} has been compromised by an attacker in a sector i. The sensor key of S_{l_c} would be accessed by the attacker, the sector key S_{k_i} and asymmetric key shared by the sector head H_{k_i} are also obtained. But, since the computation of the asymmetric key generation is not possible, because the sensor key computation along with the key pair generation requires the unique neighborhood key specified in the sensor node, which is not possible to obtain.

The proposed system withstands against outsider attacks such as eavesdropping, node capture and replay attacks. The sensor nodes are sharing asymmetric key pairs with the sector head nodes, which make it impossible for an outside attacker to monitor a data packet exchanged between two nodes by eavesdropping attack. Every data transmission channel is using different set of keys for encryption supports the WSN to withstand against the outsider attack. Since the encrypted information is communicated, even the attacker watches the communication, no useful information is obtained by the eavesdropper.

Normally, in WSNs, the node capture attacks make severe damage and data can be theft easily. But, in the proposed system, when the packet is attacked by the attacker, the entire system assumes a packet loss and refreshes the key generation process completely. Since the identifier of the captured node has already been removed from the node list of the base station, the asymmetric key pair generation will generate new set of keys which are not matching with each other. Replay attacks are faced effectively with the introduction of serial key value, so that repeated data packet could be easily identified by the sensor nodes and they will be discarded. By this method, the wastage of memory and overload of communication channel are avoided.

As the proposed system effectively handles outsider attacks, it has also capable of handling insider attacks such as black hole attack and packet drop attacks as well. Since our proposed system is always changing the neighborhood nodes every time it generates a new neighborhood key, the node will not be allowing to remove every received data from its memory. If there is no reply for a beacon message after the communication failure, the network itself identifies

a black hole attack and it will reset all its keys and reshuffle the position of the neighborhood nodes based on the location information as we discussed in the earlier subsections.

Similarly, when the network is realizing a set of data drop from the network, under certain probability the node is checked whether it has communication link in existence or not. Serial key numbers have been introduced to withstand this kind of data dropping attacks. Since the master table entries for each of the data sequence has been maintained, the drop of data packets is easily identifiable and also the same data cannot be replayed by the attacker.

5. RESULTS COMPARISON AND DISCUSSION

In this section, the experimental setup and the various simulation results obtained based upon the different experiments are elaborated. The proposed system has been compared with the existing systems and the simulated results have been obtained using MATLAB.

5.1 Experimental Setup

The implemented network contains 16 sectors and they are divided into high and low performance nodes. It was assumed to have one high performance sensor node in each of the sectors. The values used for plotting the diagrams are calculated as the average value obtained from 10 experiments. From the existing public key cryptographic systems, the encryption process has been adopted as in most of the research explained in literature. We have assessed the performance of the proposed system by asymmetric key pair generation process, rekeying after each of the node has been attached by the hacker and the energy level required by the nodes of a sector, when the number of sensor nodes are increasing. Also the time taken for computing the pairwise key with the energy required by the process has also been considered. Because of the power consumption nature of S_h sensor nodes, we are varying the computational ability by each of the nodes by varying the speed of clock cycle of the sensor nodes.

5.2 Simulation Results

In our experimental setup, we are calculating the connectivity of the sensor nodes when the number of nodes is growing higher. In Fig. 5, the connectivity of sensor nodes with the number of sensor nodes have been shown. In the shown diagram, the proposed system works better than of the other two existing approaches.

In the same experimental setup, the average of ten different numbers of sensor nodes have been tested for their connectivity. When we have similar sized sectors, it is evident that they are having identical connectivity. The energy consumed in the proposed approach has also been proven to be very optimal when the sector range is smaller.

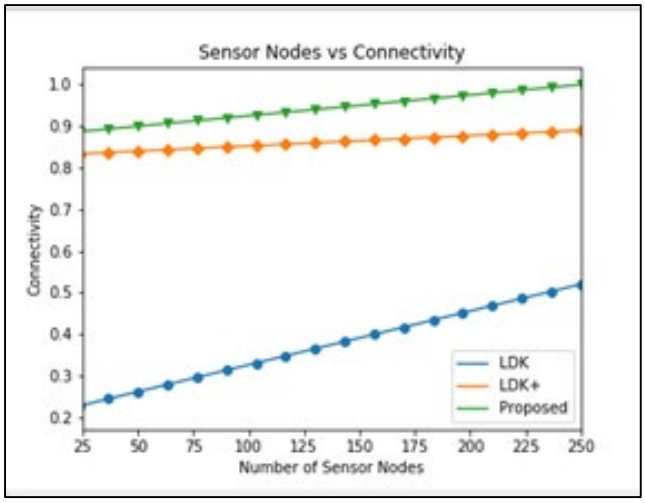

Fig. 5. Connectivity change over number of sensor nodes

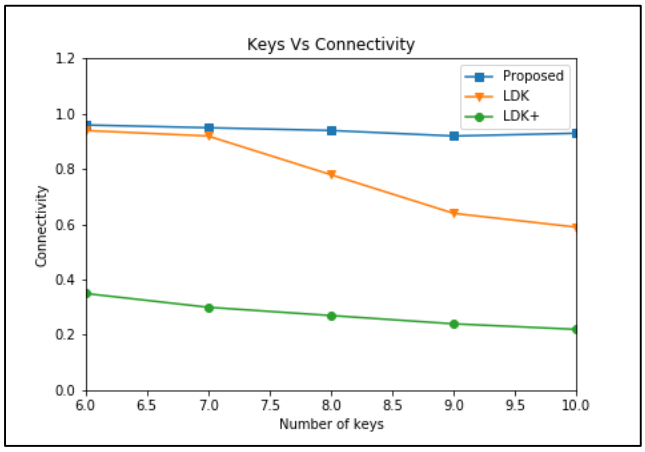

Fig. 6. Average keys generated for the active connections

In Fig. 6, the average number of keys generated with the number of active connections has been plotted. Here, the connection of LDK is not better than the proposed approach, and hence it is evident that the connectivity of the sensor nodes is increasing with reduction of the key pairs generated. The range of the communication in a sector determines the implementation of better connectivity. Since the connectivity is growing with the reduction of the sensor keys, it is obvious that the cost of the sensor network is reducing. So, the proposed approach is achieving efficiency in energy consumption and connection.

In Fig. 7, the attack probability plot explains while the sensor nodes are growing, the proposed approach is reducing the attack probability comparing with the existing systems. Since the pairwise asymmetric key generation process makes the attacking mode difficult, the proposed approach outperforms the existing systems.

In Fig. 8, the key regeneration process, if some of the nodes are captured by the attacker. Whenever a key has been caught by the attacker, the regeneration process makes it difficult to capture the sensor node again. Both the LDK and LDK+ approaches are not supporting many of the features

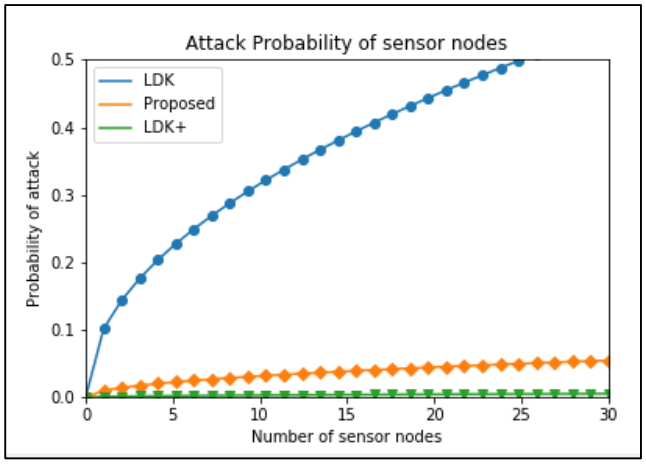

Fig. 7. Attack probability estimation

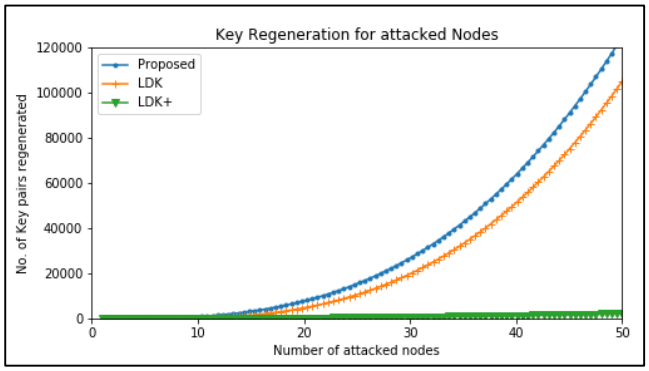

Fig. 8. Key regeneration for attacked nodes

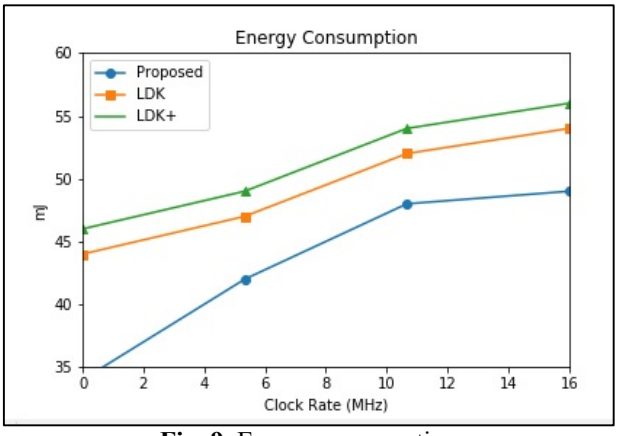

Fig. 9. Energy consumption

supported by the proposed system including rekey generation during the node capture attack.

Also, we have calculated the energy consumption for various key management approaches. The proposed approach explained in this paper is working better than that of the existing methods as plotted in the Fig. 9. The total energy consumed by different kind of sensors based on their clock cycle value are compared with the help of the energy that was spent. The rate at which the nodes are migrating from one sector to another is also affecting the energy required by the sensor node. From the experiments, we have also found that the S_h nodes are very energy consuming than of the S_l nodes.

Also, it is very clear that the mobility of the nodes from one sector to another has affected the energy consumption. But there are also cases where the energy required is decreasing once the sensor nodes are getting back to the earlier sector. Similarly, the relationship between the energy consumption and the security has also been understood.

6. CONCLUSIONS AND FUTURE WORKS

In the proposed system, an adaptive multilevel location-based approach used for key management (AML-KBS), in which the keys are generated dynamically and shared among the nodes of the wireless networks. Since the proposed approach follows a location-based system for key management, the attackers can be differentiated based upon their location. The key updating process obviously improves the security of the proposed system, and the complexity of the multiple key generation power when the sensor node is added and removed in a sector makes it tougher for an attacker to crack the key. From the experimental results, we have computed various parameters which proves that the proposed approach is better than of the any other existing key management system. In future, the proposed system can also be applied for multiple sector dynamic WSNs and the level of protection can be increased by introducing additional level of keys.

REFERENCES

Abdollahzadeh, S., Navimipour, N.J. 2016. Deployment strategies in the wireless sensor network: a comprehensive review, Computer Communications 91, 1–16. doi:10.1016/j.comcom.2016.06.003.

Afsar, M.M., Tayarani-N, M.-H. 2014. Clustering in sensor networks: A literature survey, Journal of Network and Computer Applications 46, 198–226. doi:10.1016/j.jnca.2014.09.005.

Alotaibi, M. 2018. An enhanced symmetric cryptosystem and biometric-based anonymous user authentication and session key establishment scheme for WSN, IEEE Access, 6, 70072–70087.

Anjum, F. 2010. Location dependent key management in sensor networks without using deployment knowledge, Wireless Netw., 16, 1587–1600.

Annapurna, H., Siddappa, M. 2015. A technique for multi-tier key distribution for securing group communication in wsn, in: Emerging Research in Computing, Information, Communication and Applications, Springer, 273–279. doi:10.1007/978-81-322-2550-8_26.

Bekara, C., Laurent-Maknavicius, M. 2009. Key management in wireless sensor networks, Wireless and Mobile Network Security: Security Basics, Security in On-the-shelf and Emerging Technologies, 613–648.

Chakavarika, T.T., Gupta, S.K., Chaurasia, B.K. 2017. Energy efficient key distribution and management scheme in wireless sensor networks, Wireless Personal Communications 97, 1059–1070. doi:10.1007/s11277-017-4551-2.

Chan, H., Perrig, A., Song, D. 2003. Random key predistribution schemes for sensor networks, Proc. IEEE Symp. Security and Privacy (SP '03), 197–213.

Choi, J., Bang, J., Kim, L.H., Ahn, M., Kwon, T. 2018. Location-based key management strong against insider threats in wireless sensor networks, IEEE Systems Journal.

Erfani, S.H., Javadi, H.H., Rahmani, A.M. 2015. A dynamic key management scheme for dynamic wireless sensor networks, Security and Communication Networks 8, 1040–1049. doi:10.1002/sec.1058.

Eschenauer, L., Gligor, V.D. 2002. A key-management scheme for distributed sensor networks, in Proc. ACM Conf. Computer and Communication Security, 41–47.

Eschenauer, L., Gligor, V.D. 2002. A key-management scheme for distributed sensor networks, in Proc. ACM Conf. Computer and Communication Security, 41–47.

Ferng, H.-W., Nurhakim, J., Horng, S.-J. 2014. Key management protocol with end-to-end data security and key revocation for a multi-bs wireless sensor network, Wireless Networks 20, 625–637. doi:10.1007/s11276-013-0627-4.

Gandino, F., Ferrero, R., Montrucchio, B., Rebaudengo, M. 2016. Fast hierarchical key management scheme with transitory master key for wireless sensor networks, Filippo Gandino; Renato Ferrero; Bartolomeo Montrucchio; Maurizio Rebaudengo IEEE Internet of Things Journal, 3, 1334–1345

Gura, N., Patel, A., Wander, A., Eberle, H., Shantz, S. 2004. Comparing elliptic curve cryptography and RSA on 8-Bit CPUs, Proc. Sixth Int'l Workshop Cryptographic Hardware and Embedded Systems (CHES '04), 119–132.

Huang, J.-M., Yang, S.-B., Dai, C.-L. 2013. An efficient key management scheme for data-centric storage wireless sensor networks, IERI Procedia 4, 25–31. doi:10.1016/j.ieri.2013.11.005.

Lee, J.C., Leung, V.C.M., Wong, K.H., Cao, J. 2007. Key management issues in wireless sensor networks: Current proposals and future developments, IEEE Wireless Communications, 14, 76–84.

Merkle, R. 1978. Secure communication over insecure channels, Commun. ACM, 21, 294–299.

Messai, M.-L., Seba, H., Aliouat, M. 2015. A new hierarchical key management scheme for secure clustering in wireless sensor networks, in: International Conference on Wired/Wireless Internet Communication, Springer, 411–424. doi:10.1007/978-3-319-22572-2_30.

Nikooghadam, M., Amintoosi, H. 2020. Secure communication in CloudIoT through design of a lightweight authentication and session key agreement scheme, International Journal of Communication Systems, p. e4332.

Seo, S., Bertino, E. 2013. Eliptic curve cryptography based certificateless hybrid signcryption scheme without pairing, CERIAS Technical Report 2013, https://www.cerias.purdue.edu/apps/reports and papers/. Seung-Hyun.

Seo, S.-H., Won, J., Sultana, S., Bertino, E. 2015. Effective key management in dynamic wireless sensor networks, IEEE Transactions on Information Forensics and Security 10, 371–383. doi:10.1109/TIFS.2014.2375555.

Seo, S.-H., Won, J., Sultana, S., Bertino, E. 2015. Effective key management in dynamic wireless sensor networks, IEEE Transactions on Information Forensics and Security, 10, 371–383

Shin, S., Kwon, T. 2019. A lightweight three-factor authentication and key agreement scheme in wireless sensor networks for smart homes, Sensors, 19, 2012.

Simpl´ıcio-Jr., M.A., Barreto, P.S., Margi, C.B., Carvalho, T.C. 2010. A survey on key management mechanisms for distributed wireless sensor networks, Comput. Networks, 54, 2591–2612.

Thevar, G.K.C., Rohini, G. 2017. Energy efficient geographical key management scheme for authentication in mobile wireless sensor networks, Wireless Networks 23, 1479–1489. doi:10.1007/s11276-016-1228-9.

Wang, H., Li, Q. 2006. Efficient implementation of public key cryptosystems on mote sensors, Proc. Eighth Int'l Conf. Information and Comm. Security (ICICS '06), 519–528.

Zhang, J., Varadharajan, V. 2010. Wireless sensor network key management survey and taxonomy, J. Netw. Comput. Applications, 33, 63–75.

Zhang, Y., Li, X., Liu, J., Yang, J., Cui, B. 2017. A secure hierarchical key management scheme in wireless sensor network, International Journal of Distributed Sensor Networks 8, 547471. doi:10.1155/2012/547471.

Zhu, S., Setia, S., Jajodia, S. 2006. Leap+: Efficient security mechanisms for large-scale distributed sensor networks, ACM Trans. Sensor Netw., 2, 500–528.