An Efficient Key Management Scheme in Multi-Tier and Multi-Cluster Wireless Sensor Networks

Manivannan Doraipandian[1], P. Neelamegam[2]
(Corresponding author: Manivannan Doraipandian)

School of Computing, SASTRA University[1]
Tirumalaisamudram, Thanjavur, Tamil Nadu 613401, India
School of Electrical and Electronics Engineering, SASTRA University[2]
(Email: dmv@cse.sastra.edu)

(Received Apr. 16, 2013; revised and accepted Aug. 15 & Nov. 26, 2013)

Abstract

Wireless Sensor Network is a collection of autonomous sensor nodes placed spatially. Unlike wired networks the sensor nodes here are subject to resource constraints such as memory, power and computation constraints. Key management and Security are the area of research in WSN. To ensure high level security encryption is necessary. The strength of any encryption algorithm depends upon the key used. So Key Management plays a significant role. The proposed KMS using LLT matrix achieves both Node-to-Node communication and Group communication. The main objective of the scheme is to strengthen the data transferring security mechanisms and also to ensure efficient key generation and management along with authentication. The main feature of this proposed system is 100% Local-connectivity; efficient node revocation methodology, perfect resilience; three-level authentication cum key generation and the most importantly reduced the storage. The scheme and its detailed performance analysis are discussed in this paper.

Keywords: Cholesky decomposition, key connectivity, resilience, WSN

1 Introduction

WSN [12] is a collection of nodes from hundreds to thousands. Each node has processing units, sensing unit and power source usually the battery. Sensor nodes are resource constrained in terms of computation, memory. Because of its transmission nature and also because of its deployment in hostile environments, security mechanisms available for wired ad-hoc networks are not applicable for WSN. So new security mechanisms [9] should be introduced but satisfy the security requirements such as authentication, confidentiality, integrity and availability. Though many cryptographic algorithms are available, but the strength of the algorithm purely depends on the key used. For eg. If AES is incorporated, whoever involved in building up the security mechanisms knows about the AES. So the importance will be on key and also the size of the key. If 128 bit key is used, a possible set of key will be in 2128. So to establish a secure communication key management plays a vital role. Key management includes key generation, distribution and storage of keys. The attackers usually made an attack on the key management level rather than cryptographic algorithm level. Since the sensor node is resource constrained designing a key management scheme for WSN is challenging issue. In recent years, many key management schemes are proposed. Key management schemes are broadly classified into three categories: key pre-distribution, arbitrated key mechanisms and self-enforcing mechanisms. Arbitrated keying mechanisms depend upon trusted third party agent. Of that if the node gets compromised all information about the network will get revealed. Self-enforcing mechanism is a public key cryptography method. Since sensor nodes are resource constraining this method is not preferable.

Almost all key management schemes [1, 2, 3, 6, 7, 10, 16] are based on key pre-distribution method in which keys are loaded into sensor nodes before deployment. Designing a suitable key management scheme for all kinds of WSN organization such as hierarchical or distributed is another challenging issue. Based upon applications and architecture used KMS has to be defined. Once after designing the KMS, the metrics [12] to be evaluated against KMS is security (Authentication, resilience, node revocation), efficiency (memory, processing, bandwidth, energy, key connectivity) and flexibility (deployment knowledge, scalability). Satisfying all the metrics in a single key management scheme is difficult. If suppose group key communication is incorporated, periodic updating of group key is necessary. This increases expenses on rekeying. Thus in this situation key connectivity is not an issue rather rekeying is. The evaluation metrics are mainly based on the architecture and keying mechanism used.

2 Related Works

Some schemes follows partial pairwise key methodology [5]. For a network with n nodes, it is not necessary to store n-1 keys in each node to achieve a connected graph. The degree of each node is determined by the probability of connectivity. Usually it is expected to be high. Basic Scheme [9], Q-composite [3] are based on this method. These schemes undergo three steps: Key Pre-Distribution phase; shared-key phase and path key establishment phase. Bloms [1] scheme; Du-et-al [6] multiple space, LU [12] schemes are also pair-wise schemes. Instead of storing the keys directly [18], corresponding rows and columns of the matrix are stored [14] and pairwise key is generated using vector multiplication whenever two nodes want to communicate. Most of the hierarchical network schemes [8] use group key mechanisms. Resilience and node revocation becomes an issue. Many schemes [17] are introduced without deployment knowledge. As a result, the probability to nodes to be within each others communication is less. Thus the connectivity is less. Considering all these factors, the PROPOSED scheme is a mixture of pairwise, group, matrix-based, hierarchical network with deployment knowledge.

3 Our Contribution

The proposed scheme is a matrix based scheme. A symmetric matrix is decomposed into two matrices using CHOLESKY factorization. It is almost similar to LU scheme. The reason for choosing CHOLESKY factorization is that: the two matrices are lower triangular matrix and its transpose. This reduces STORAGE, COMPUTATION and COMMUNICATION overhead to a large extent. It is enough to store only the row values unlike LU Scheme where in row and corresponding columns are stored. This reduces the MEMORY CONSUMPTION to half of that consumed in LU [12, 16] scheme.

The proposed scheme uses the HIERARACHICAL NETWORK STRUCTURE to enhance DIVISION OF LABOUR. Processing and work decrease down the group. This gives a clear idea of what type of nodes to be used at which level. Two types of keys used in the proposed scheme: pairwise key [3, 5, 10] and the other Group key. Group key is mainly used for commenting purposes. Pairwise keys serve two purposes. Firstly, they are used in message passing then its for node revocation. The corresponding group head will initiate the node revocation. This involves deleting records pertaining to the captured node and replacing the old group row with new one. This cannot be multi-cast because even the captured node will receive the message. Thus the pairwise key between a node and the group head is used for it.

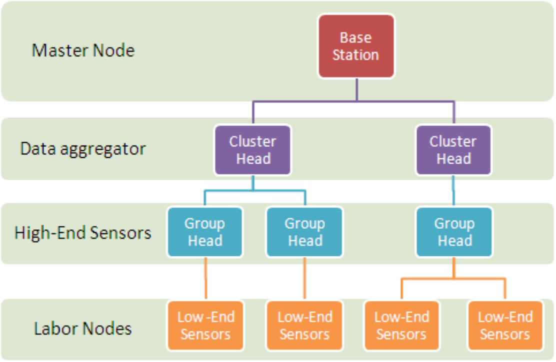

Figure 1: Layered clustered architecture - WSN

Proposed scheme also uses message passing efficiently by providing three level authentications cum key generation mechanism within three steps.

4 The Proposed Scheme

4.1 Architecture

The four layered clustered architecture comprises of Base station at the top level, Cluster heads at the second level, High end sensors as the group head in the third level and Low end sensor nodes at the bottom level (see Figure 1).

The main advantage of a multi-layered clustered architecture [11] is that the number of keys loaded in each sensor nodes will be appreciably less compared to distributed sensor networks. The hierarchical architecture enhances the scalability of the system. Further it provides Division of Labor system where in each node is loaded with optimal work it can perform [7]. Thus, hierarchical clustered architecture [13] gives a clear idea of what type of nodes to be used at different levels. The main objective in WSN is to achieve 100% connectivity at low power consumption. To achieve maximum communication range the node consumes maximum transmission power and thus the range of communication is traded-off with energy consumption. Typically, WSN nodes are expected to work efficiently at low power consumption. Hence decreasing the power consumption cuts down the communication range.

The nodes communicate with other nodes without any nodes intervention without regarding the transmission power of communication [15]. The main advantage of this is the security which is 100%, further; the data received is a primary data. Also there is minimal possibility of data loss. But still this is not welcomed in WSN because of its high energy consumption. Here for 100% connectivity all the nodes should be in the communication range of other nodes, this limits the network coverage.

These multi-hop techniques are used. The communication range of a node is reduced subject to the energy consumption constraint. In order to achieve 100% connectivity it is not necessary for all nodes to be within the

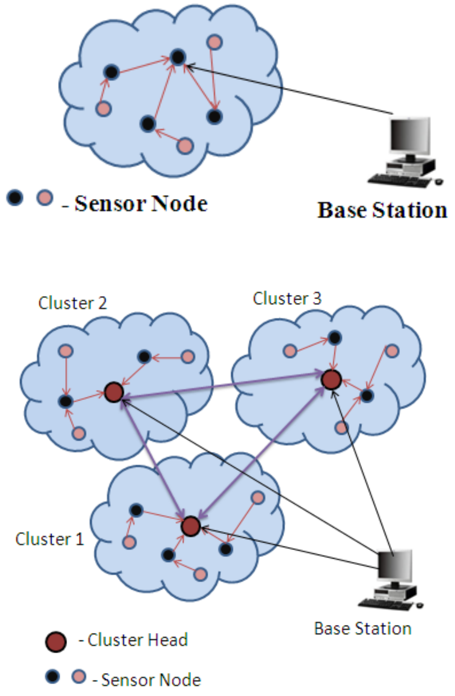

Figure 2: Direct and multi hop communication - WSN

communication range of nodes (see Figure 2).

It can be achieved using multi-hopping where in the two nodes, which are out of their communication range, communicated via the nodes present within the range. The identification of such nodes leads to path establishment phase. The main disadvantage of this is data loss and insecurity. Since the communication range [17] of a node is reduced keeping in mind the power consumption, multi-hop techniques are used for long range communication. Here low end sensor nodes are grouped under High end Group heads (nodes). If one node in a group wants to communicate with the node in the other group multi-hopping is done via their respective group heads. By doing this data is more secure as the receiving node knows the source of the message, also always there exist exactly two nodes (the respective group heads) in between the sender and the receiver. Thus the proposed scheme uses the optimized connectivity with minimal power consumption as the criterion for grouping nodes. The same criterion is followed for clustering group heads under powerful cluster heads. The cluster heads communicates directly with the base station. The proposed multi-layered clustered architecture is hence the best architecture.

4.2 Communication Flow

Base Station: Full duplex communication between base station and cluster heads.

Cluster Head: Full duplex communication between cluster heads (inter), base station and group heads (intra).

Group Head: Full duplex communication between group heads (inter), cluster head and sensor nodes (intra).

Sensor Nodes: Full duplex communication between nodes and group heads (within a group) [13, 14, 15, 18].

4.3 Outline of the Scheme

All the sensor nodes are loaded with the programs and data before deployment. Based on the locality of deployment, the sensor nodes are grouped under High end sensor nodes. Further the groups are clustered (depending upon the structure of deployment) under cluster head. Thus the knowledge about the locality of sensor nodes is known in advance. The base station is fed with the details of all sensor nodes, group as well as cluster heads such as Number and IDs of all nodes belonging to a group; Number and IDs of all group head belonging to a cluster; Number of clusters and ID of each cluster head. Lower Triangular matrices decomposed from a symmetric matrices form the basis of key generation. The symmetric decomposition is done using CHOLESKY decomposition [15].

Assume there are c clusters, g groups, and n nodes. Thus $c \times c$ symmetric matrix is allotted for inter-cluster communication along with the base; $c(g \times g)$ symmetric matrices for inter grout (intra cluster) communication within a cluster; $g(n \times n)$ symmetric matrices for inter node (intra group) communication. The trick of the trade is that the values of the order of the symmetric matrices are kept as large as possible. This is done to achieve better scalability. Using separate sets of matrices for different layers of architecture, different sets of keys are generated for each layer. Each layer is a completed graph with m nodes (m is appreciably less than the order of the symmetric matrix allotted to it). For commanding purpose say from base station to cluster heads or from cluster heads to its group heads or from group heads to its nodes a unique key is generated at each level. A key array consisting of possible keys with which a node can communicate is stored in its memory. This ensures authenticated communication between nodes. A Common hashing array for generating indices is used for encrypting the message. Periodically checks are made by the respective heads to test whether a node is alive or dead.

4.4 System Components and Functionalities

Base Station. This is the master node of the network. It is at the topmost level of the architecture [18]. It commands and controls all its co-ordinate nodes. It receives the aggregated data from various cluster heads and processes it [7, 18]. It stores cluster IDs, group IDs, number of clusters, number of groups in a cluster, number of nodes in a group along with their

IDs. Further it stores one row of the $c \times c$ matrix for establishing pairwise key between cluster heads for inter cluster communication; one common row from the $c \times c$ matrix for broadcasting. It also stores Key pool, hashing array [11].

Cluster Head. This node serves two purposes: one is that it reduces the burden of the base station by performing data aggregation and distribution of messages from/to various group heads; the other is that it aids inter-group communication by acting as a mediator [7, 18]. Moreover it initiates group head revocation. It stores Cluster IDs, Group IDs, and number of groups under its control. One row of $c \times c$ matrix for inter cluster communication; one row from the allotted GXG matrix for intra- cluster (inter-group) communication; the common row stored in base station (to receive message broadcasted by the base station) and one common row of $g \times g$ matrix for broadcasting (to group heads) purpose. Further it also stores the hashing array and key pool list.

High End Sensor Nodes/Group Heads. This node plays the role of cluster heads at this level, i.e., it performs data aggregation and distribution of messages from/to its nodes. This also takes the role of initiating node revocation [7, 18]. It stores IDs of node belonging to it, IDs of group heads belonging to same cluster. One row of GXG matrix for inter group communications; one row of NXN matrix for intra-group (inter-node) communication; the common row of GXG matrix stored in its cluster head (to receive the message broadcasted by cluster head); one common row of NXN matrix for broadcasting (to nodes) purpose. Further it also stores the hashing array and key pool list [4, 13].

Low End Sensor Nodes (simple called nodes). This is the working node of the system, which senses and transmits sensed data to its group heads. Group ID, one row of the $n \times n$ matrix for communicating with group head; the common row stored in its group head (to receive the message broadcasted by it). Further it also holds the hashing array and a key list with two elements one the key value for communicating to group head and the other for receiving the broadcasted message [4, 13].

NOTE: Different sets of matrices are dedicated to different cluster heads. Though the hashing array stored in the nodes is same for all, the key pool list varies in accordance with the matrices allotted to it.

4.5 Key Management

4.5.1 Symmetric Matrix Decomposition

The methodologies used in this scheme are listed as follows:

Cholesky factorization. LU decomposition constructs both lower and upper triangular factors **L** and **U** Cholesky decomposition constructs a lower triangular matrix **L** whose transpose $\mathbf{L}^T$ itself an upper triangular matrix such that A=LL^T.

Cholesky Factorization Algorithm. If the order of the A matrix is N then,

1) Set $k = 1$;
2) Repeat the following until $k <= N$;
3) For K^{th} $N \times N$ Matrix:
   a. $a_{k,k} = \sqrt{a_{k,k}}$;
   b. $a_{k+1:N,k} = a_{k+1:N,k} / a_{k,k}$;
   c. $a_{k+1:N,K+1} = a_{k+1:N,K+1} - a_{k+1:N}, \ k * a_{k+1,k}$;
   d. $a_{k+2:N,k+2} = a_{k+2:N,k+2} - a_{k+2:N,k} * a_{k+2,K}$ and so on;
   e. Increment k by 1.

4.5.2 Pre-deployment Phase

All the parameters that are mentioned in the system and component phase are loaded to the appropriate nodes.

4.5.3 Key-establishment Phase

After successful deployment of nodes establishing connectivity is the crucial step. This is done using keys. In simple words two nodes can communicate if and only if they share a common key.

4.5.4 Pair-wise Key Establishment

Steps involved in pair-wise key establishment [10] between two nodes:

1) The sender node A sends its row R_{na} in format I node to the receiver node B.
   Message Format I: NodeID$_B$ || row_values || hashing_index (base) || hashing_index (shift) || NodeID$_A$.

2) Node B receiver the messages and retrieves the row values of A. It computes the Key K_{AB} and checks it presence in the Key pool. If it is present then Node B sends its row, checked bet, hash of the key in format 2 to A.
   Message Format II: NodeID$_A$ || row_values || hash(key) || checked_bht || hashing_index(base) || hashing_index(shift) || NodeID$_B$.

3) Node A receives the message and retrieves row values R_{nb} of B, key value K_{AB} and computes the key value, K_{BA} using R_{na} and R_{nb}. Then it checks whether K_{BA} is present in its key pool and also r_{eA} matches with K_{AB}. If it matches node A sends the message to B using shh computed key.

4.5.5 Group Key Establishment

Group key are mainly used for commanding and controlling the nodes. There is only one group key at any level. Here broadcasting technique is followed.

Steps involved in group key establishment:

1) The group head broadcasts the message to all its nodes using message format I mentioned below. **Message Format I:** Group ID || row_values || hash on key || cipher message || hashing_index (base) || hashing_index (shift).

2) Appropriate nodes receive the message and retrieve the necessary command.

The process of decryption is as same as the process depicted above in pair wise key establishment.

4.5.6 Cluster Key Establishment

This process is similar to that of group key establishment.

5 Performance Analysis

The following are the factors (affecting performance of the system) that are analyzed in this phase. These are (1) Key connectivity; (2) Efficiency (Computation, Storage, Communication); (3) Scalability.

5.1 Key Connectivity

It is a measure of the possibility of communication between two nodes in a network; this is usually referred to as local connectivity [1, 3, 6]. Global connectivity is a measure of connected components in the entire network. For system with high performance key connectivity should be high. This is because with high connectivity probability of multi-hopping reduces. This reduces unnecessary intermediate communications which in turn reduce the transmission power. Thus battery power (power source of sensor nodes) is reserved for processing and hence performance increases with key connectivity.

In the proposed scheme, 100% Key connectivity is achieved at each tier of the hierarchy, i.e., the network is a completely connected graph at each level of hierarchy (completely pairwise) as shown in Figure 3. The proposed network (structure) is a connected (not a fully connected) graph. As mentioned the connected components of the graph are fully connected. Generally, a lot of communication happens only within nodes of the same level, i.e., the number of intra-level communications is more when compared to inter-level communication. Thus it is enough if 100% key connectivity is assured within a level and inter-level communication can be achieved using secondary or ternary neighbors. The proposed scheme uses this strategy.

Random pairwise scheme: In order to reduce the Key storage when compared to EG [9] scheme, the entire graph is divided into several overlapping connected components (nodes). Though this scheme does not support 100% connectivity but ensures the network is connected. Since all the nodes perform the same tasks, frequent communication between them is required. Thus for two nodes, which are far apart, to communicate lots of hopping has to be done, this increases communication overhead. Thus high key connectivity is achieved at the expense of transmission Power.

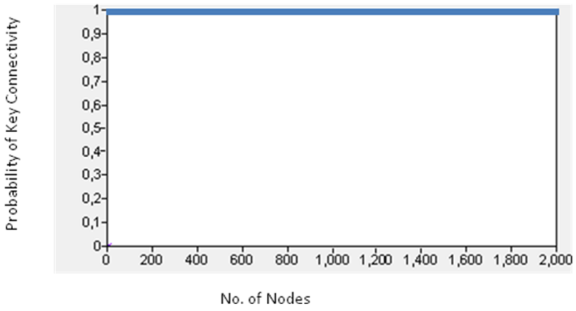

Figure 3: Key connectivity vs. Number of nodes

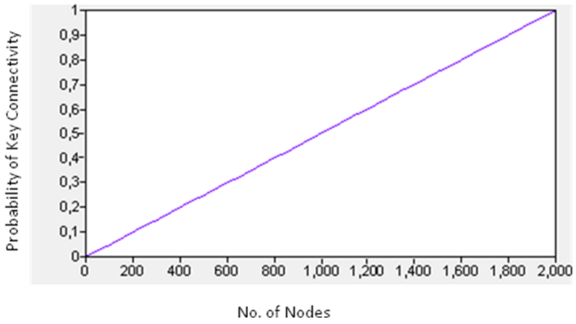

Figure 4: Key connectivity vs Number of hops; RP $n = 2000$, p vs m

For RP scheme the key Connectivity will be $p = (1/n) * m$ where p denotes a probability of connectivity; n denotes number of nodes; m denotes a degree of each node. The key connectivity for RP scheme is shown in Figure 4.

Asymmetric Pre-distribution scheme: The key connectivity is not 100% initially. Whenever two non-connected nodes want to communicate, they first establish a pair-wise key between them with their first degree H sensor node. Thus Key connectivity gradually reaches 100% at the expense of memory, i.e., the storage memory in L sensor nodes inner-cases.

The Key connectivity for AP scheme is $1-((p-m)!(p-l)!/p!(p-m-l)!))$ where p = pool size; e = number of keys in H - Sensor node; l = number of keys in L- Sensor node. In the proposed scheme 100% key connectivity is achieved between primary neighbors, unlike AP scheme as shown in Figure 5. Thus a balance is stroked between Key connectivity and Key storage.

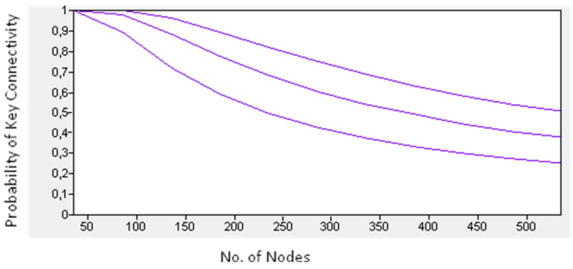

Figure 5: key Connectivity for AP scheme

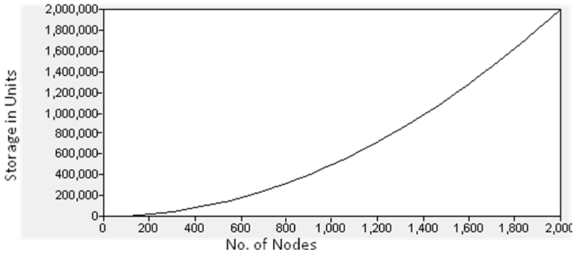

Figure 6: Storage in proposed scheme

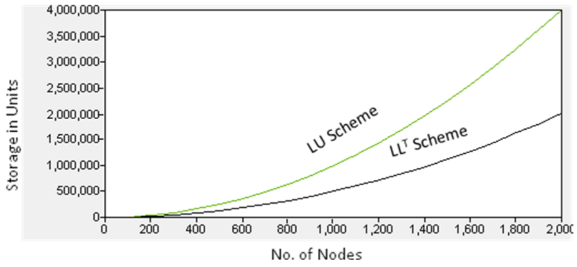

Figure 7: Storage in LU vs. Proposed Scheme

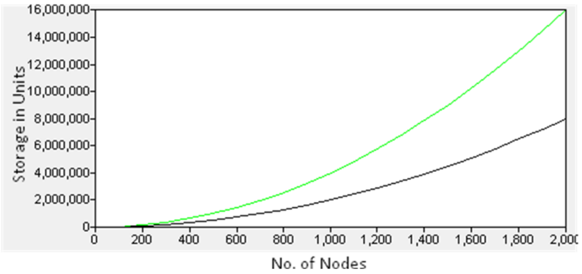

Figure 8: Number of nodes vs Memory

5.2 Efficiency

5.2.1 Storage

A typical node is subject to memory constraints for better performance, i.e., a maximum storage capacity of a node is generally small. Also high end nodes have better storage capacity compared to low end sensor nodes. The proposed scheme uses this fact and stores data accordingly, i.e., the storage decreases down the hierarchy. Matrix generation and other major storing activities are limited with the top level itself.

Proposed scheme: The Storage will be $(x/2)(x+1)$ where x is the number of Nodes and the respective graph is shown in Figure 6. In LU [10] Decomposition each and every node stores one row of Lower Triangular matrix and corresponding column of upper matrix to generate keys by matrix multiplication. In the proposed scheme the upper triangular Is the transpose of the lower triangular matrix ($U = L^T$). This reduces the number of rows to be stored in each node to one. Let minimum size of one row be on an average 4 bytes and say there are 5000 nodes (as in a typical network); LU utilizes 40000 (2*4*5000) bytes whereas the proposed scheme consumes only half the above value, i.e., 20000 bytes (4*5000). The remaining reserved memory is efficiently for authentication and computational purposes.

The storage for LU will be x*(x+1) whereas in LL^T it will be $(x/2)(x+1)$ where x be the number of nodes (see Figure 7).

In Random Pairwise [10] scheme the voting keys, that are stored in low end sensor nodes, used for node revocation increases the storage in the nodes. In the proposed scheme any node revocation within a group or a cluster is initiated and taken care by their corresponding group heads of cluster heads, imposing no additional memory consumption. Head nodes being a high end sensor node can store additional information. Thus the network objectives are achieved subject to memory constraints without any degradation in performance.

In Du et al. scheme τ distinct keys spaces from the possible choices (say w) are randomly loaded into the nodes. The size of one row is λ+1, thus for each node (λ +1) τ units are required.In the Proposed scheme many entries in lower triangular matrix are zero thus size of one row is far less than that used in Du et al. scheme. This strategy helps to reduce memory consumption to a large extent.

In LEAP each and every node is loaded with individual key, pairwise key, group key and cluster key to achieve high connectivity between different levels of hierarchy. The proposed scheme uses only pairwise key and group key to achieve the connectivity that LEAP achieves. This reduces the memory consumption to almost half of that in LEAP (see Figure 8).

5.2.2 Computation

Computation is done at the expense of power consumption. Since a node is expected to work with minimal power consumption too much computation degrades nodes performance. The proposed scheme basically involves three computations multiplication, one-level base conversion, shifting. Multiplication is done for key generation. The base for conversion is chosen in such a way that it terminates at one level itself, thus restricting the number of divisions to one. Shifting being a bit twiddling operation does not consume much power. Thus the computational power consumption is relatively less compared to many schemes and also the proposed scheme does not perform any computation for node authentication. Also since most of the row entries are zero computation becomes simple.

In Polynomial based scheme the nodes are supposed to compute their key using n-degree polynomial functions with two variables which involve computing exponential powers of those variables and their summation. This consumes a lot power. Proposed scheme limits the number of arithmetic computations to one or two and mainly performs simple bit twiddling operations and hence consumes relatively less power. In Blom [10] and Du et al. scheme the nodes compute keys by multiplying rows and columns. To reduce storage on each node, only the seed of the column (Vander monde matrix with seed s) is stored. But this imposes computational overhead in generating the column which happens at the expense of power consumption. The proposed scheme has no such overhead in generating column as only rows are stored.

5.2.3 Communication

Communication is directly related to transmission power. Thus for high performance unnecessary communications should be avoided. In the proposed scheme the nodes are loaded with all the possible keys with which it can communicate, keeping in mind the transmission power, before deployment. Many schemes have shared-Key discovery phase and path-key establishment phase. This involves a lot of communication between nodes. The proposed scheme being completely pre-deployed does not involve any communication of this type. Thus saving a lot of transmission power. Further the proposed scheme involves only two communications for key generation and node authentication.

In the AP scheme [7], Du et al. matrix scheme, Blundo, Liu and Ning scheme, q-composite scheme involves both shared key establishment And path key establishment phase which increase communication. In LU scheme the key computation involves three steps whereas proposed scheme uses only two steps, hence 33% transmission power is saved. Also in LU if there is a key mismatch then the authentication mechanism (μ TESLA) is initiated which consumes computation power. The proposed scheme does node authentication and key establishment within the two steps and hence is efficient.

5.3 Scalability

This measures the performance of the network in addition of new nodes. For a typical network the performance should not be affected while adding new nodes. In the proposed scheme the order of the matrix is set to the maximum having a futuristic view on the scalability. When a node is added to a particular group, unused row of the allotted matrix is loaded into it along with all other remaining necessary data before deployment. The strategy used here for accommodating a large number of new nodes is to group them under new group. This strategy handles scalability to a large extent.

In Hierarchical LU the rows and columns are randomly loaded into the nodes. When more and more new nodes are added the possibility of two nodes having same rows increases. Thus probability of link compromising increases. In RP scheme each node is loaded with m identifiers in its vicinity. When a new node is added, its key identity must be updated in that connected component of the network. This imposes communication overhead.

6 Security Analysis

6.1 Resilience

A network should be secure enough so that the entire message passing is done secretly. This ensures no data leakage. Thus a malicious user cannot hack the information from the nodes in the network. Resilience is a measure of how quickly the system recovers upon node capturing. The recovery depends on the impact of node capturing on the system, i.e., it indirectly measures how much remaining nodes and links get compromised on node capturing. A good wireless sensor network must definitely be resilient, otherwise, the entire system will be attacked and all the data can be hacked out of it.

6.2 Message Interception

This is a situation where malicious users intercept messages (brute force attacks) in the network by snoops, traffic analysis, modification, masquerading, repudiation, replaying, and denial of service and many other security threats and attacks. A good network is supposed to ensure high data integrity, confidentiality and availability.

The proposed scheme produces cipher messages which are highly encrypted. Thus any malicious user will not be able to retrieve any information from it. By doing this the proposed scheme overcomes the traffic analysis and spoofing threats. Here the node ID and the computed key values which are checked against a key pool list acts as the digital signature to provide authentication. Further, the proposed scheme uses encipherment and thus ensures data integrity and overcomes masquerading, modification, etc.

Assuming the awkward situation where in the attacker retrieved the content of the message and found the key. In the proposed scheme each and every node has a unique pairwise key with each node within a level. The attacker remains helpless with one key as he will not be able to masquerade with other nodes. Thus no link gets compromised. The only link that gets compromised is the link from which the attacker retrieved the information.

In such a situation the corresponding group or cluster head initiates node recovery mechanism after identifying malicious network interfaces. Heads replace the row values and key pool list of all the nodes in the vicinity of the alien interface with new rows from its allotted matrix and a corresponding key pool list. This is done by unicasting all this information to respective nodes.

6.3 Node Compromising

Here the attacker uses any physical attacks to directly capture nodes. On capturing a node the attacker will get to know about all the information that is stored in it.

Considering an awkward situation where in a node in the proposed network is captured physically and all the information that is stored in it are known to the attacker. Using this node the attacker can easily communicate with all the other nodes in its communication range. If the node is a group head then the attacker will be able to retrieve information from the entire cluster in which it belongs. In the proposed scheme the respective group head or cluster head immediately initiates recovery mechanism after identifying the attacked node, giving no room for the attacker to extract information from other nodes. Once the attacked node is known by the group head it initiates node revocation mechanism. Its first and foremost task is to transmit the node ID of the attacked node to all the remaining nodes in the cluster. This is done by unicasting to the nearest node, the attacked node ID, the new row for group communication and change in hashing index in hashing format (mentioned above). The nearest node retrieves all the necessary information and performs three basic operations. Firstly it passes this message to its neighbor node which is not the attacked one. Secondly, it deletes the attacked node ID and its corresponding key value from the list it stores. It then changes the row for group communication and updates the old key value for group communication in the key list with the new computed group key. Thirdly, the node changes the range of base and shifting number generated by the pseudo random generator. Thus though the attacker who knows the message format can't hack it because he doesn't know to what number the index is referring to. All the other nodes also do the same. Further this ensures resilient property in the network.

In the AP scheme, Hierarchical LU scheme, Du et al. [17] matrix scheme the probability of using the same key to establish links between different nodes are more because the rows are randomly loaded into the nodes, i.e., two or more nodes may have same row values for shared key establishment. Thus when one link gets compromised it will also affect all the other links which used the same row for generations. In RP scheme the voting keys play a crucial role in node revocation. This increases storage in every node. Thus resilience is achieved at the expense of storage. Also voting leads to communication overhead.

In LU scheme [16] the probability of two nodes to have same row value increases with an increase in the number of nodes and number of keys chosen for the scheme. Thus number of link compromising increases with increase in the number of keys in a node as shown in Figure 9. strategy helps to reduce memory consumption to a large extent.

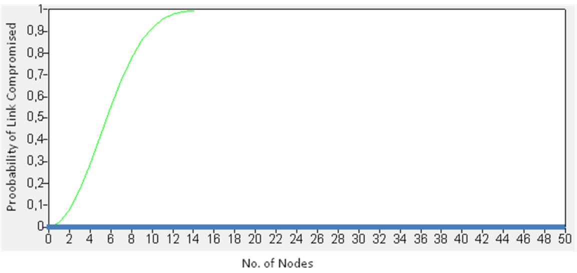

Figure 9: Number of nodes vs probability of link compromised

6.4 Node Authentication

Authentication [13, 14] ensures that both the parties at the ends of the communication are authenticated. In the proposed scheme, no additional authentication mechanism, like μ-Tesla, is used. Instead a part of the key generation mechanism is used for authenticating. First the receiver node that extracts the ID from the message checks with its node ID. If it matches then it is confirmed that the message is from authenticated node. Further confirmation is done by checking the computed key value with the key pool list it possesses. If there is a match then authenticated communication takes place between them. In cases of Mismatch in either of the steps, the corresponding group or cluster head is alerted by the node in which mismatch occurred. The heads probe into this issue and finds whether mismatch is due to loss of data or due to malpractices. Thus two level authentications cum key generation reduces communication to a large extent.

In many schemes such as q-composite scheme, SHELL, the keys are directly deployed in the nodes. The nodes there directly send messages using those keys. In order to authenticate nodes some additional mechanisms are needed. But only a few schemes incorporate such mechanisms. Thus attacking such networks with less or authentication is simple. Authentication ensures data integrity and confidentiality in a network.

7 Summary and Conclusion

An Efficient Key Management scheme for WSN with multi-tier and multi clustered architecture using LLT is discussed LLT matrix will play a vital role to achieve FULL local key connectivity with less communication and less computation overhead. This Proposed protocol is an efficient, secured, scalable and multilevel authenticated between nodes, nodes to group head, group head to cluster head and cluster head to cluster head. In this architecture, Choleskey decomposition constructs a lower triangular matrix L, whose transpose LT itself an upper triangular matrix such that A=rL^T. Using this technique pairwise key establishment phase, Group key establishment phase, cluster key establishment phase is achieved. Performance analyzes in terms of key connectivity, efficiency, scalability are done and the results are noted. The security analysis are made related to resilience and node authentication and the results are noted finally the comparison is made between proposed scheme with an existing key management scheme in terms of performance and security analysis. The summary of the results is discussed in Table 1. The results indicate that the proposed scheme is well suited for dynamic homogeneous and heterogeneous sensor networks.

Table 1: Summary and result

Metrics	Achievements
Key Connectivity	100% Local key Connectivity
Storage	Less Storage;
Communication	No shared key and path key establishment phase
Computation	Bit twiddling operation reduces computation overhead
Scalability	Unused rows from the matrix is loaded
Resilience	Changing the range of base and shift number
Authentication	Multi-Tier authentication, where node ID's acts as a digital signature

References

[1] R. Blom, "An optimal class of symmetric key generation systems", in *Advances in Cryptology (EUROCRYPT'84)*, LNCS 209, pp. 335–338, 1985.

[2] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, M. Yungcet, "Perfectly secure key distribution for dynamic conferences", *Information and Computation*, vol. 146, no. 1, pp. 1–23, 1998.

[3] H. Chan, A. Perrig, D. Song, "Random key predistribution schemes for sensor networks", in *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, pp. 197–213, 2003.

[4] C. C. Chang, L. Harn, and T. F. Cheng, "Notes on polynomial-based key management for secure intra-group and inter-group communication", *International Journal of Network Security*, vol. 16, no. 2, pp. 143–148, Mar. 2014.

[5] H. Dai, H. Xu, "Key pre-distribution approach in wireless sensor networks using LU matrix", *IEEE Sensors Journal*, vol. 10, no. 8, pp. 1399–1409, Aug. 2010.

[6] W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, A. Khalili, "A pairwise key predistribution scheme for wireless sensor network", *ACM Transactions on Information and System Security*, vol. 8, no. 2. pp. 228–258, 2005.

[7] X. Du, Y. Xiao, M. Guizani, and H. H. Chen, "An effective key management scheme for heterogeneous sensor networks", *Ad Hoc Networks*, vol. 5, no. 1, pp. 24–34, 2007.

[8] M. Eltoweissy, H. Heydari, L. Morales, H. Sadborough, "Combinatorial optimization of key management in group communications", *Journal of Network and Systems Management*, vol. 12, no. 1, pp. 33–50, 2004.

[9] L. Eschenauer, V. D. Gligor, "A key management scheme for distributed sensor networks", in *Proceedings of the 9th ACM Conference on Computer and Communication Security (CCS'02)*, pp. 41–47, 2002.

[10] A. A. Kamal, "Cryptanalysis of a polynomial-based key management scheme for secure group communication", *International Journal of Network Security*, vol. 15, no. 1, pp. 68–70, 2013.

[11] D. Macedonio, M. Merro, "A semantic analysis of key management protocols for wireless sensor networks", in *Science of Computer Programming*, pp. 53–78, Elsevier Science, 2014.

[12] D. Manivannan, R. Ezhilarasie, P. Neelamegam, K. R. Anuj, "An efficient and hybrid key management scheme for three tier wireless sensor networks using LU matrix", in *Proceedings of the First International Conference on Advances in Computing and Communications (ACC'11)*, pp. 111–121, Kochi, India, 2011.

[13] O. K. Sahingoz, "Large scale wireless sensor networks with multi-level dynamic key management scheme", *Journal of System Architecture*, vol. 59, pp. 801–807, 2013.

[14] Qi Shi, N. Zhang, M. Merabti and K. Kifayat, "Resource-efficient authentic key establishment in heterogeneous wireless sensor networks", *Journal of Parallel and Distributed Computing*, vol. 73, no. 2, pp. 235–249, 2013.

[15] M. Turkanovic, B. Brumen and M. Hölbl, "A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the internet of things notion", in *Ad Hoc Networks*, pp. 96–112, Elsevier Science, 2014.

[16] Mi Wen, Y. Zheng, H. Li, K. Chen, "A hierarchical composition of LU matrix-based key distribution scheme for sensor networks", in . *Emerging Technologies in Knowledge Discovery and Data Mining*, LNCS 4819, pp. 608–620, Springer, 2007.

[17] M. F. Younis, K. Ghumman, M. Eltoweissy, "Location-aware combinatorial key management scheme for clustered sensor networks", *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 8, pp. 865–882, 2006.

[18] S. Zhu, S. Setia, and S. Jajodia, "LEAP: Efficient security mechanisms for large-scale distributed sensor networks", *ACM Transactions on Sensor Networks*, vol. 2, no. 4, pp. 500–528, 2006.

Manivannan Doraipandian obtained Ph.D. degree from SASTRA University, Thanjavur in 2013. Since 1996, he has been in the teaching profession and currently he is a Senior Assistant Professor in the Department of Computer Science, School of Computing, SASTRA University, Thanjavur, Tamil Nadu, India. His area of interest include Cryptography, Security in Embedded Systems, Wireless Sensor Networks using ARM processors and Embedded Communication Systems.

P. Neelamegam obtained Ph.D. degree from Bharathidasan University, Tiruchirappalli in 1992. Since 1971, he has been in the teaching profession and currently he is a Professor in the Department of Electronics and Instrumentation, School of Electrical and Electronics engineering, SASTRA University, Thanjavur, Tamil Nadu, India. His research interests include Microprocessors, Microcontrollers, embedded system based Instrumentation, wireless sensor network, neural network and fuzzy logic.