Journal of Information Systems and Telecommunication
Vol.10, No.1, January-March 2022, 20-27
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Secure Key Management Scheme for Hierarchical Network Using Combinatorial Design

Siddiq Iqbal[1*], Sujatha B R[2]

[1].Department of Electronics & Telecommunication B.M.S. Institute of Technology & Management, Bengaluru, India
[2].Department of Electronics & Communication Malnad College of Engineering, Hassan, India

Received: 31 Jan 2021 / Revised: 09 Aug 2021/ Accepted: 09 Sep 2021
DOI: https://doi.org/10.52547/jist.15691.10.37.20

Abstract

The wireless sensor network (WSN) signifies to a gathering of spatially spread and committed sensors for observing and logging the physical states of the environment and for organizing the information gathered at the central Base station. Many security threats may affect the functioning of these networks. Security of the data in the system depends on the cryptographic procedure and the methods where encryption and decryption keys are developed among the sensors. Symmetric key foundation is one of the best applicable ideal models for safe exchanges in WSNs. The main goal is to improve and evaluate certain issues, such as node attack, to provide better key strength, connectivity, security for node interaction, and throughput. Uniform Balanced Incomplete Block Design (UBIBD) is used to generate the keys allocated by the base station to the cluster head. The cluster head distributes keys to its members using Symmetric Balanced Incomplete Block Design (SBIBD), and the keys are refreshed on a regular basis to avoid out-of-date entries. In wireless sensor networks, compromised nodes can be used to inject false reports. The concept of interacting between sensor nodes using keys and establishing a secure connection aids in ensuring the network's security.

Keywords: Wireless Sensor Networks; Combinatorial Design; Key Management; Key Distribution; Key Refreshment; Balanced Incomplete Block Design.

1- Introduction

A wireless sensor network has a massive number of minute nodes, each with its own set of energy limitations and computing difficulties. WSN offers a wide range of applications, including military, medical, and residential appliance management. In general, sensor devices have resource constraints; these constraints govern the design of WSNs and, as a result, influence the security level of sensor networks. Transmission distance, restricted processing power, battery lifetime, limited memory, random scattering of nodes, and bandwidth are examples of such constraints [3].

- Transmission distance: In sensor networks, sensors have transmission range which is restricted. Longer the distance of transmission more energy will be consumed. Thus, the communication range for sensor nodes is controlled due to the limited available resources.
- Limited processing power: A sensor node in a network has its own special hardware and software architecture. During the manufacturing process of a sensor node, the main aim is to save the energy consumed by the node, to achieve this the hardware and software architecture should be as simple as possible.
- Battery life: As the sensor devices are operating usually in outdoor environment, they are completely dependent on battery power. Sensor nodes rely solely on their battery for power, so once it runs out, the status of the sensor node is considered inactive.
- Limited memory: Sensor node just like any other device has its own unique operating system. This operating system is preloaded into the memory of the sensor and occupies a part of storage. The remaining memory of the node should be managed efficiently. For efficient utilisation of memory the storage space occupied by the cryptographic keys which are used in the network for secure transmission of data should be low. Less space occupied by the keys gives space for other processing operations to be carried out without any limitation.
- Prior deployment knowledge: In most of the applications, the deployment of sensor nodes is in high risk areas, such as war zones. It is critical to select the deployment method, as usually the deployment is random or carried out in real-time. This problem adds to location information of a sensor node being insufficient or most likely unavailable. During the start of key assignment

✉ **Siddiq Iqbal**
Siddiq@bmsit.in

process, the key management procedures should be independent of the sensor location.

• Bandwidth: In WSN, the key establishment process is constrained by sensor node constraints. As previously stated, the hardware architecture of sensor nodes is straightforward. Because such nodes' transmitters have limited capacities, they can't send a large amount of data all at once. As a result, the bandwidth capacity of sensor nodes is limited. As a result, key management techniques should consider this issue.

Because of these limits and the vulnerability of the WSN to attackers, protecting the sensor network is seen as a big task. To achieve this task, many algorithms have been suggested in the literature. WSN necessitates the use of a cryptography algorithm, which must be appropriately chosen, and the most crucial component for those algorithms is to resolve the Key agreement or management problem, which would be considered necessary provide the encrypted and authenticated data transmission among sensor nodes in order to have a channel with good security. Many security needs, such as confidentiality, integrity, authenticity, and availability, must be met by the cryptography algorithm (CIAA). To counteract attacks in WSN, a key management scheme is used [1]. Key distribution can be accomplished in a variety of ways, the simplest of which is to allot a single unique secret key to all the nodes in the network. However, if an adversary manages to obtain this secret key, the entire network's security is jeopardised. A more realistic way to key pre-distribution is to assign unique pair-wise keys per each link amongst sensor nodes in the network. The resiliency of such an approach is quite high, because compromising any of pair-wise key has no impact on the remaining network. However, because each node has to maintain keys with all of the other sensor nodes in the network, the pair-wise keys-based approach increases key storage overhead. Combinatorial design-based key pre-distribution is a kind of compromise, where we sacrifice network robustness in exchange for lower storage overhead. This architecture includes allocating a pool of keys to each of the sensor nodes, with each pair of key sets sharing some keys.

The remaining paper is structured as follows: Sect. 2 will go over some of the previous work in the field of key management. In Sect. 3, we will state the proposed scheme, which includes different phases of key management. In Sect. 4, we will explain the performance evaluation. In Sect. 5, we thoroughly explain the simulation results. In Sect. 6, Finally, we will present the conclusion of our paper.

2- Related Works

The process of keeping track of different cryptographic keys in a cryptosystem is known as key management. Key generation, exchange, storage, handling, crypto-shredding, and replacement are all aspects of key management. The generation and distribution of keys is carried out using BIBD [2] [4]. The essence of the key management scheme's major updation based on a unified design is illustrated, two main update techniques are proposed, and the results of the three approaches are analysed in two ways. Finally, the two strategies are extended to other combinatorial plan-based key management systems, and the second strategy is enhanced. Key management approaches based on combinatorial plans receive a lot of attention [16]. Key administration schemes that do not include key updates will be less protected in the long run [7]. Despite the fact that the NBIBD was first published in 1967 and sparked widespread interest, they were virtually not known in combinatorial writing outside of the competitive literature, and their entire combinatorial inferences were not sufficient. The reviews and extensions of numerical understanding on NBIBDs were discussed. For NBIBDs, isomorphism and auto orphisms are specified, as well as building methods. NBIBD is divided into several categories, each of which is defined and illustrated [8]. State-of-the-art cryptography procedures are used to secure computer networks. While it has been claimed that designing good cryptographic procedures is the simplest approach to secure a large scale network, it seems that security concerns in algorithms and their executions are frequently observed [10] [11]. The development of wireless data transmission in the initial twentieth century is credited with the first cryptographic explosion; An adversary may clearly read radio transmission just as easy as a true recipient. [9].

Block constructions are especially useful in related experiments when working with heterogeneous test rings to increase the impact of treatment evaluations. The Embedded Balanced Incomplete Block Design (NBIBD) is a concept with blocks of two systems, one of which is the original location and the other of which is the ultimate target blocks that neglect one of the two systems and leaving a balanced incomplete block design. For gathering a number of characteristics, a novel methodology for generating the structure of built-in balanced incomplete blocks is devised [14].

The nested block design is characterised as a floor plan with two systems of blocks, the second of which is positioned at the base (two squares of the second system are present in each square of the first), with the final goal being a balanced incomplete block on both frames. an image in which the squares correspond to the squares of another system. The list of these plans is only repeated fifteen times in each treatment. Yates' extension to retrieve block data in balanced incomplete block diagrams is the subject of this investigation [15]. The analysis is made for a balanced incomplete scheme that is typically developed. Sec-LEACH [17] combines TESLA and random key pre-distribution to produce a secure communication system

that is efficient and resists few attacks. Sec-LEACH is being used to compare with the proposed method.

3- Proposed Scheme

The suggested key management is divided into phases (shown in Figure.1).

- Node registration.
- Key generation and Key distribution.
- Key refreshment.

3-1- Node Registration

Initially, the base station registers all the nodes which have been deployed, before key distribution [6]. During this phase, the base station passes different registration messages and assign IDs to all the nodes in the network. Every node in the cluster transmits the identification parameters to all the other members in the cluster. A node in a cluster has three association parameters, the node ID (distinct for every node), the cluster ID (distinct for each cluster of sensor nodes), location of the node.

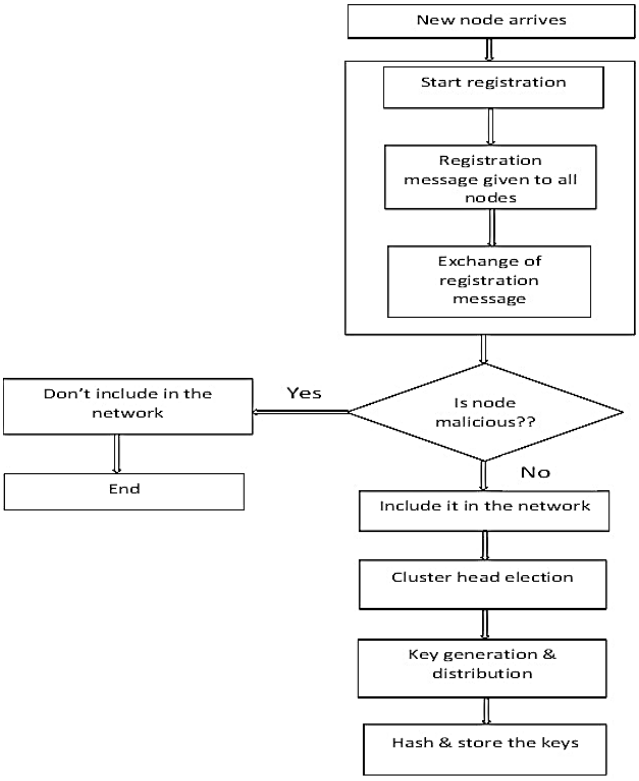

Fig. 1 Proposed scheme

The base station transmits information of every node to all the cluster members along with the registration messages transmitted to the nodes. The Neighbor Information Table holds the information of cluster members [5]. A node in the cluster first sends a message to its closest neighbors, followed by sending the message to all other members in the cluster. Every sensor node sends the message in addition to its ID. When a node is passing the message further, it sends its own ID along with the forwarded message. All the cluster nodes receive the registration messages from each cluster member.

A node in a cluster has one more table which stores registration messages and IDs of cluster members. When there is a mismatch in the information of registration message and ID received from any node, the node forwards the location of such a node to the base station. The malicious nature of a node is confirmed by the base station after analysing the data transmission and reception pattern of the suspicious node and if found to be malicious then the location is added to the blacklist. All other nodes which are tested not to be suspicious are registered with the base station. The nodes that have registered become a part of the network's subsequent processes.

Nodes are detected as malicious and removed based on their location information. Depending on its location, the node closest to the cluster's centre announces itself as the cluster head and sends a declaration message to the base station as well as all of the cluster's members. The acknowledgement received from the base station and cluster members is used to identify a node as the cluster head. When a node is marked as CH, a link is made between the BS, CH, and members nodes.

3-2- Key generation and distribution

Cluster head keys are created using the UBIBD (9,3,1) design by the base station, while member node IDs are generated using the SBIBD (45,12,3) design, where (9,3,1) and (45,12,3) are v (number of distinct values), b (number of blocks), and lambda (number of blocks in which two distinct values appear) respectively.

Following steps are used for key generation:

- Size of CH keys is 128 bit, which are generated randomly.
- Member node IDs are 128 bit random number.
- In order to generate authentication key and MN keys, the CH keys are used (Figure 2).
  1. Consider the following three keys that belong to a CH based on the UBIBD (9, 3,1) architecture (each block of the design contains three distinct keys, which are given to each cluster head).
  2. Each block in the UBIBD design has its own set of keys, pairs chosen at random are all unique and utilised to generate unique login keys for the member nodes of each cluster.
  3. Assume [a, b, c] are cluster head keys, Let [c, a] be a pair of 256-bit authentication keys created at random by the product of c and a.

$$\text{i.e., } AK = c * a$$

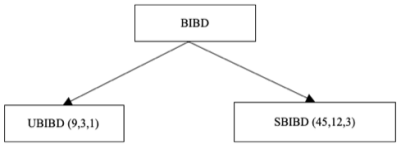

Fig. 2 Key Generation

The 256 bit AK and the data messages are used to generate the hashed output. Using HMAC SHA-256, this hashed data will be used for the authentication and integrity of the data.

4. MN keys are generated using one of the three CH keys, as well as the other two keys.
5. In the set [a, b, c] CH keys, the key which is not used to calculate AK is 'b' and the keys used in AK are c & a.
6. 'b' gets multiplied with both 'c' and 'a' separately to generate sub products and then the resulting products are multiplied to obtain a disntict multiple. i.e.,

   prod_1 = b * c
   prod_2 = b * a
   Umultiple=prod_1 * prod_2

   The secret key (SK) is extracted from the first half of unique multiple (256 bit MSB).

7. To generate MN keys, multiply the resultant multiple by the 128-bit MN IDs.
   MN keys = SK * ID (SBIBD)

As a result of all of the preceding calculations, the final result is:

- CH keys - 128 bit
- Authentication Key - 256 bit
- MN keys - 382 bit

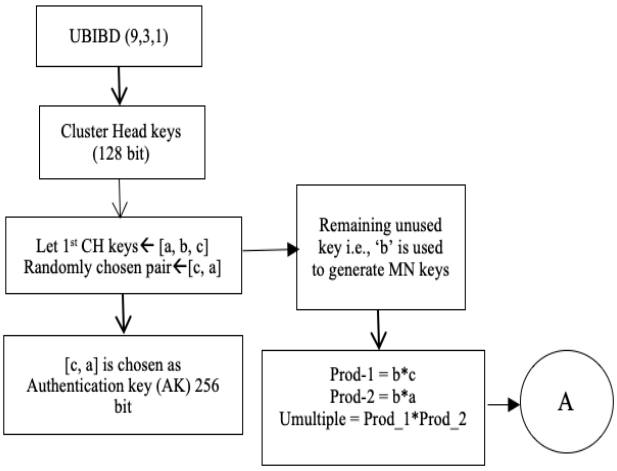

Fig. 2a Key Generation, UBIBD (9,3,1)

3-3- Key refreshment

XOR computations are used to refresh keys (Fig 3).

- • A total of thirteen 128-bit random numbers are generated. (Magic words: one WCH and twelve WMN).
  - W CH- used to refresh cluster head keys.
  - W MN - used for member node's key refreshment

Key refreshment for CH:

- Refreshment of CH key is carried by using one of the thirteen magic words (WCH). WCH is circularly shifted four times, and the resulting bit is R.

  R = circular shift (WCH , 4)
- XOR operation is carried out on the old key with the magic word WCH and the result obtained is defined as Kn.

  Kn = Old_Key CH ⊕ WCH
- A new refreshed key is generated by applying XOR on 'R' with the previous result Kn.

  K CH_refreshed = R ⊕ Kn

MN key refreshment:

- From the thirteen words, twelve magic words are used for key refreshment of member node.
- The XOR operation is conducted using corresponding W-MN in each cluster, and IDs are renewed. i.e.,

  R = circular shift (W-MN , 4)
  Kn = Old_key MN ⊕ W-MN
  MN refreshed ID = R ⊕ Kn
- MN refreshed key is generated by multiplying the Secret key (SK) generated by the new CH key with its corresponding MN refreshed ID.
- MN refreshed key = SK * MN refreshed ID

As a result, all Cluster head and member node keys are refreshed on a regular basis, ensuring that the UBIB and SBIB designs are met.

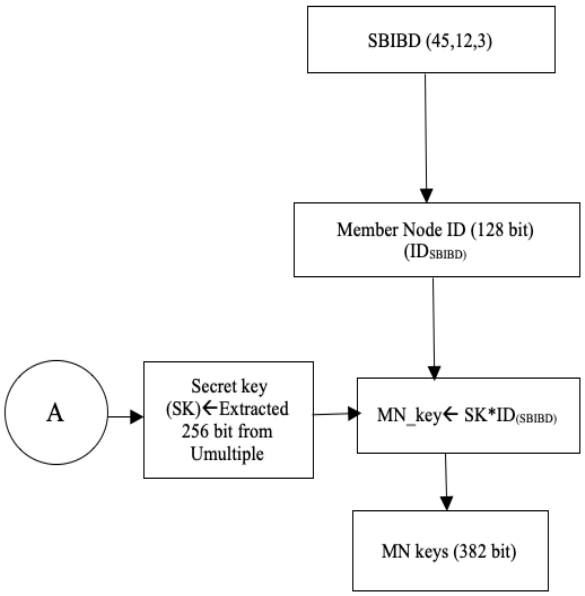

Fig. 2b Key Generation, SBIBD (45,12,3)

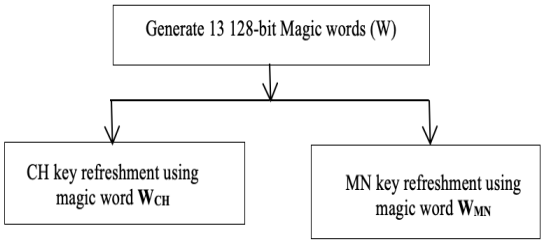

Fig. 3 Key Refreshment

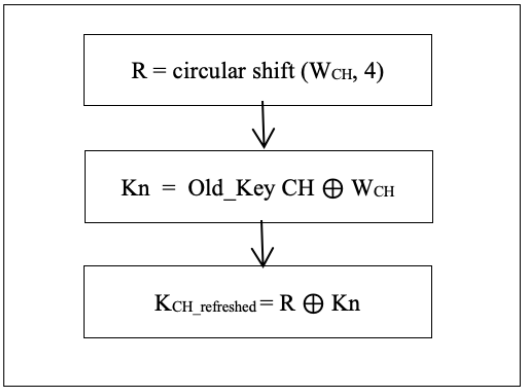

Fig. 3a CH key refreshment using magic word W_{CH}

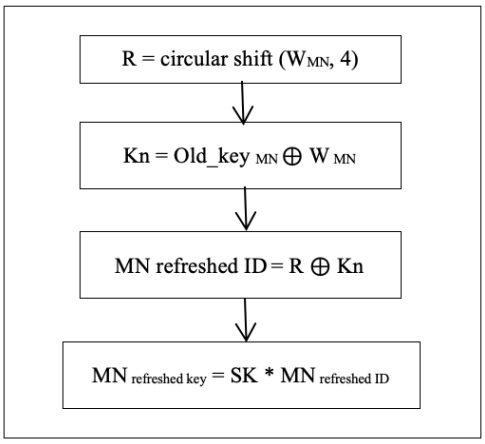

Fig. 3b MN key refreshment using magic word W_{MN}

4- Performance Evaluation

The performance of the proposed scheme presented in this paper is evaluated in this section. For implementation and performance analysis, we used the MATLAB tool for programming. A network with a size of 200m x 200m and a transmission range of 30m has been considered. The member nodes are located within the range of all the clusters, with the base station in the middle. For simulation, 500 rounds are taken and the nodes with initial energy of 0.5J is considered. Total number of nodes considered for simulation is 50 and two nodes are assumed to be compromised. Whichever node that is not registered is compromised. Randomly few nodes are considered to be compromised. The graphs show the average results. The modeling results are compared.

The metrics used for performance analysis are:

- Throughput - The total data sent over the network.
- Security – Security is given to the network by eliminating malicious nodes [5].
- Energy – Energy remaining in the network at the end of each round.

5- Simulation Results

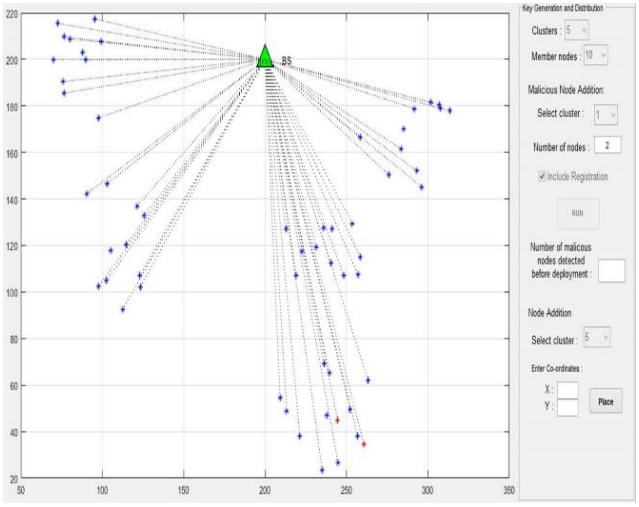

Fig. 4 Registration of nodes

Figure 4 depicts the registration process, which identifies and eliminates malicious nodes. As shown in the diagram, malicious items are highlighted in red and are not registered. Unregistered nodes will not be included in network since the base station issues keys to all registered nodes.

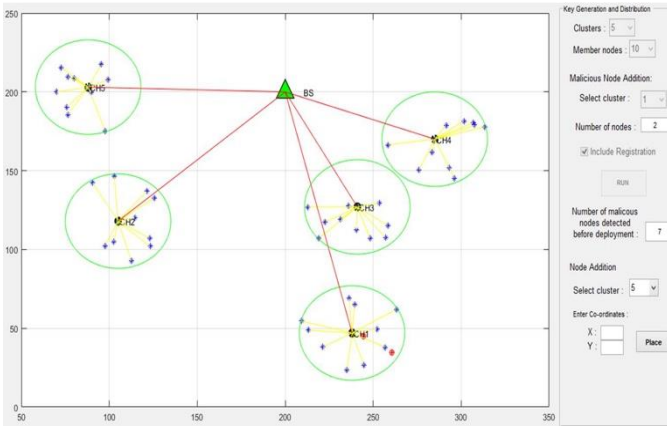

Fig 5. GUI Showing Base Station and Clusters

Figure 5 depicts the complete deployment of the network, including the base station, cluster heads, and member nodes. A malicious node can be added to any of the clusters by the user. The GUI displays a drop box where we can specify the number of member nodes, clusters, and suspicious nodes. The box which shows the number of malicious nodes which have been identified, here we can also insert new nodes.

Fig.6 shows the closer view of the cluster 1. The cluster consists of 10-member nodes and two malicious nodes which are red in color. This image shows the cluster after the identification and elimination of malicious nodes.

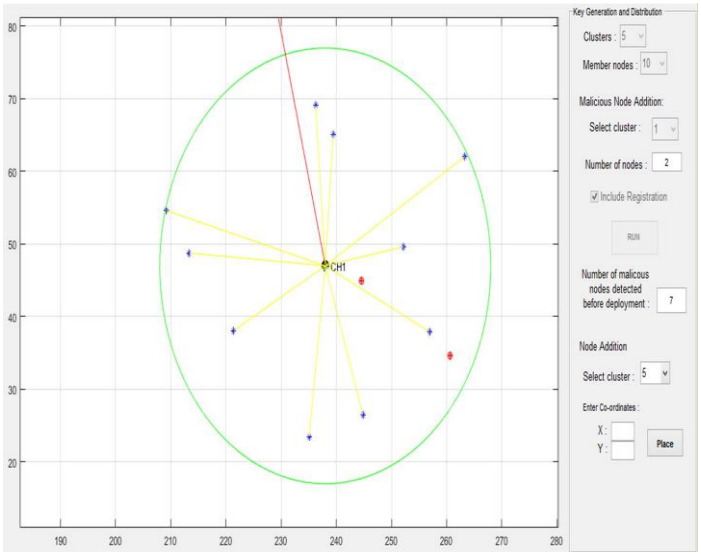

Fig. 6. Closer view of the cluster

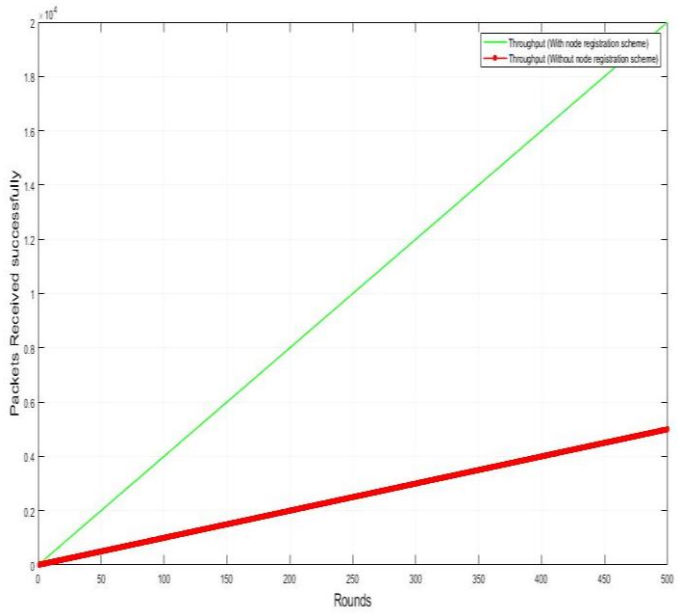

Fig 7. Throughput graph

The fig. 7 shows the throughput in the network for 500 rounds. In the above figure the red line represents the packets received successfully without the registration scheme that is 5000 for all the rounds, green line which is increasing linearly represents the packets received successfully with registration scheme which is 20000. Without employing node registration process the throughput of the network will be reduced as a result of the presence of malicious nodes.

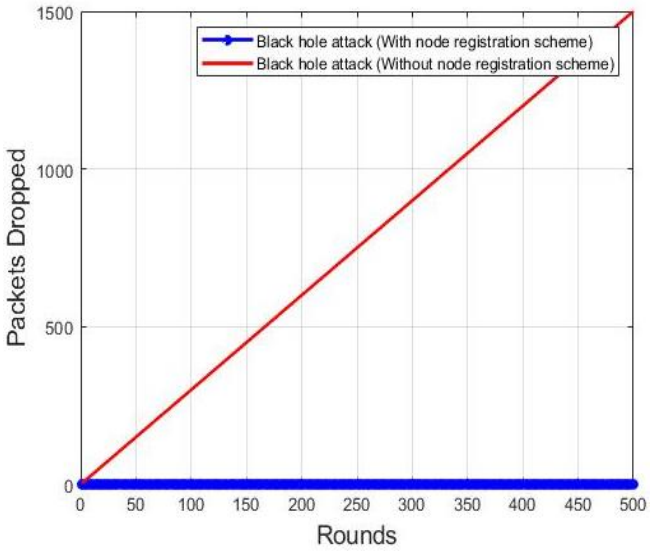

Fig 8. Packets dropped graph

The fig.8 shows the packets dropped in the network for 500 rounds. The red line in the graph is increasing linearly shows that 1500 packets are dropped without the registration scheme and the blue line in the graph shows the packets dropped with registration scheme is 0. Without including the node registration process, the number of packets dropped in every round of communication will be more than that when only registered nodes are involved in communication.

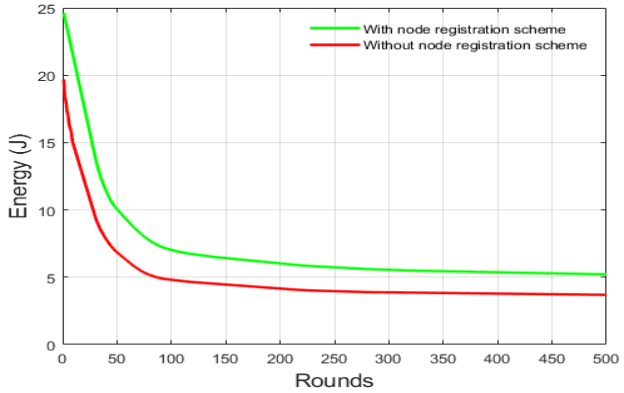

Fig 9. Energy graph

The fig.9 shows the energy graph of the network, which shows the amount of energy consumed as the number of rounds increase. Since there are 50 nodes, each node as 0.5 J of energy, so total energy is 25 J. From the graph, the total energy of the network at the end of 500 rounds of communication involving only registered i.e. non-malicious nodes was found to be around 5J whereas, that when involved the malicious nodes was found to be around 4.8J. The presence of malicious node negatively influences the overall energy consumption of the network.

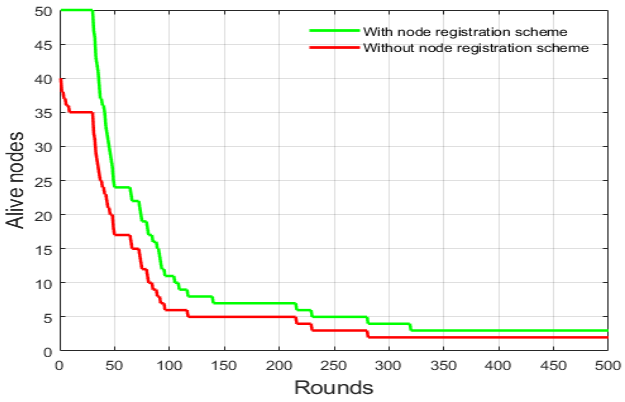

Figure 10. Alive nodes graph

The fig.10 shows the number the nodes alive in the network. Green line indicates there are 50 valid nodes, red line indicated there are 10 nodes out of 50 which are not registered. It was observed that with node registration scheme, since the influence of malicious nodes will be minimized, the number of alive nodes after 500 rounds of communication was found to be more (2 alive nodes) than that without node registration scheme (1 alive node). It is desired to have the nodes alive for longer duration which in turn would extend the lifetime of the network.

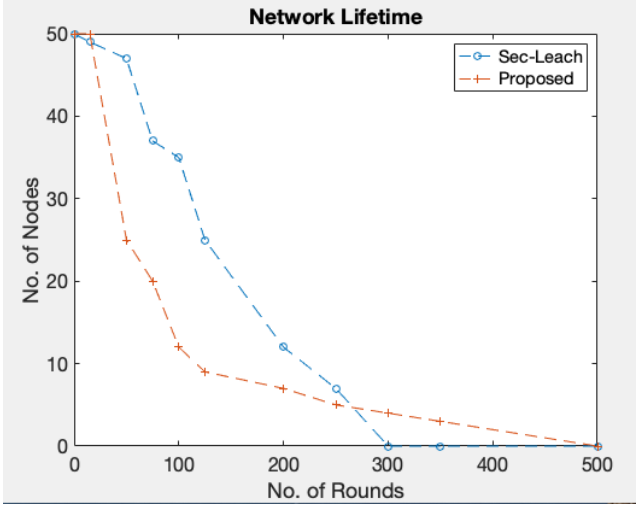

Figure 11. Network Lifetime Analysis

Figure 11 analyzes the results obtained in terms of network lifetime. Because there are various approaches to analysing the lifetime of a WSN, the number of alive nodes is taken into account here. From the figure, it is depicted that the presented method extends the network lifetime and extends the death of the nodes in WSN. Overall the network life time has been increased in the proposed method compared to existing method.

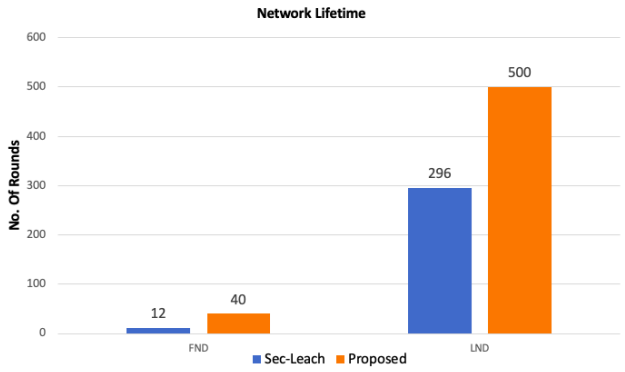

Figure 12. Network Lifetime

Another method for determining network lifetime analysis is to identify the first and last nodes to die (FND and LND) in a WSN. Figure 12 depicts the proposed and existing methods' network lifetime analysis in terms of FND and LND. Figure 12 shows that the presented technique, when compared to other methods, delayed the death of the first node . In Sec-LEACH FND occurs after 12 rounds, whereas the presented model's FND occurs after 40 rounds. Similarly, Sec-LEACH has an LND of 296 rounds, whereas the presented model has an LND of 500 rounds.

6- Conclusions

The proposed approach was implemented utilising static wireless nodes. When compared to existing schemes, this scheme provides better connectivity, coverage, key strength, and attack resistance. Because the primary benefit of sensor nodes is their mobility, they can be integrated into the system through the use of advanced routing protocols that boost system efficiency and support mobility. The impacts of increased node density must be investigated further. The proposed technique could be developed even more to provide higher resilience to a wider range of threats. Cluster heads are assumed to have high resistance against attacks, even if a cluster head is compromised the cells connected to that particular Cluster head are compromised. By making further advancements in this area, the system can be made more reliable.

References

[1] Alok Kumar, Neha Bansal, and Alwyn R. Pais. "New key pre-distribution scheme based on combinatorial design for wireless sensor networks." IET Communications 13, no. 7, 2019, pp. 892-897.

[2] Yang, Chin-Nung, Ting-Ju Lin, Song-Yu Wu, Shin-Shang Lin, and Wei Bi. "Cost Effective Hash Chain Based Key Pre-Distribution Scheme for Wireless Sensor Network." In 2018 IEEE 18th International Conference on Communication Technology (ICCT), pp. 518-522. IEEE, 2018.

[3] Khawla Naji Shnaikat and Ayman Ahmed Al Qudah, "Key management techniques in wireless sensor networks", International Journal of Network Security & Its Applications (IJNSA) Vol.6, No.6, November 2014, pp. 49-63.

[4] G. Choi and I. Lee, "A key distribution system for user authentication using pairing-based in a WSN," 2017 4th International Conference on Computer Applications and Information Processing Technology (CAIPT), 2017, pp. 1-4.

[5] Ruhul Amin, and G. P. Biswas. "Design and analysis of bilinear pairing based mutual authentication and key agreement protocol usable in multi-server environment." Wireless Personal Communications 84, no. 1, 2015, pp. 439-462.

[6] M. M. M. Fouad, M. M. Mostafa and A. R. Dawood, "A Pairwise Key Pre-distribution Scheme Based on Prior Deployment Knowledge," 2011 Third International Conference on Computational Intelligence, Communication Systems and Networks, 2011, pp. 184-189.

[7] Chonghuan Xu, and Weinan Liu, "Key Updating Methods for Combinatorial Design Based Key Management Schemes", Hindawi Publishing Corporation, Journal of Sensors, 2014.

[8] J.P. Morgana, D.A. Preeceb and D.H. Reesb, "Nested balanced incomplete block designs", Discrete Mathematics, vol. 231, 2001.

[9] Bart Preneel, "Cryptography for network security: failures, successes and challenges." In International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security, pp. 36-54. Springer, Berlin, Heidelberg, 2010.

[10] Mukund R. Joshi, and Renuka Avinash Karkade, "Network security with cryptography." International Journal of Computer Science and Mobile Computing 4, no. 1, 2015, pp. 201-204.

[11] P. M. Wightman and M. A. Labrador, "A3: A Topology Construction Algorithm for Wireless Sensor Networks," IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference, 2008, pp. 1-6.

[12] Shalini M S, Hemanth S R, "EKM-CI: Effectual key administration in dynamic wireless sensor network", International Research Journal of Engineering and Technology (IRJET), May 2018, pp.226-228.

[13] V. S. Janani and M. S. K. Manikandan, "Enhanced Security Using Cluster Based Certificate Management and ECC-CRT Key Agreement Schemes in Mobile Ad hoc Networks." Wireless Personal Communications 97, no. 4, 2017, pp. 6131-6150.

[14] A. Dey, U. S. Das, and A. K. Banerjee, "Construction of nested balanced incomplete block designs." Calcutta Statistical Association Bulletin 35, no. 3-4, 1986,pp. 161-168.

[15] D. A. Preece, "Nested balanced incomplete block designs." Biometrika 54, no. 3-4, 1967, pp. 479-486.

[16] Kazeem A. Osuolale and Oluwaseun A. Otekunrin. "An algorithm for constructing symmetric ((r+ 1) v, kr, kλ) BIBDs from affine resolvable (v, b, r, k, λ) BIBDs." Annals. Computer Science Series 12, no. 2, 2014.

[17] Sariga Arjunan, Sujatha Pothula and Dhavachelvan Ponnurangam, "F5N- based unequal clustering protocol (F5NUCP) for wireless sensor networks." International Journal of Communication Systems 31, no. 17, 2018, e3811.