

# A Novel Dynamic Key Management Scheme Based On Hamming Distance for Wireless Sensor Networks

R.Divya , T.Thirumurugan

**Abstract** - Numerous key management schemes have been proposed for sensor networks. The objective of key management is to dynamically establish and maintain secure channels among communicating nodes. Many schemes, referred to as static schemes, have adopted the principle of key predistribution with the underlying assumption of a relatively static short-lived network (node replenishments are rare, and keys outlive the network). An emerging class of schemes, dynamic key management schemes, assumes long-lived networks with more frequent addition of new nodes, thus requiring network rekeying for sustained security and survivability. This paper proposes a dynamic key management scheme by combining the advantages of simple cryptography and random key distribution schemes. When the hamming distance between the two nodes is found high, the unique key is changed instead of changing the set of keys and the communication takes place by using any one of the set of key x-oring with the new unique key. The security and performance of the proposed algorithm is compared with the existing dynamic key management scheme based on Exclusion Basis System and prove that the proposed scheme performs better when compared to existing scheme by considering the number of nodes colluded with time. The result obtained by simulation also shows that the proposed scheme provides security solution and performs better than the existing scheme.

**Index Terms** - WSN's, dynamic key management, collusion, hamming distance, security.

## 1. INTRODUCTION

THE envisioned growth in utilizing sensor networks in a wide variety of sensitive applications ranging from healthcare to warfare is stimulating numerous efforts to secure these networks. Sensor networks comprise a large number of tiny sensor nodes that collect and (partially) process data from the surrounding environment. The data is then communicated, using wireless links, to aggregation and forwarding nodes (or gateways) that may further process the data and communicate it to the outside world through one or more base stations (or command nodes). Base stations are the entry points to the network where user requests begin and network responses are received. Typically, gateways and base stations are higher-end nodes. It is to be noted, however, that various sensor, gateway, and base station functions can be performed by the same or different nodes. The sensitivity of collected data makes encryption keys essential to secure sensor networks.

### 1.1 Key Management

The term key may refer to a simple key (e.g., 128-bit string) or a more complex key construct (e.g., a symmetric bivariate key polynomial). A large number of keys need to be managed in order to encrypt and authenticate sensitive data exchanged. The objective of key management is to dynamically

establish and maintain secure channels among communicating parties.

Typically, key management schemes use administrative keys (key encryption keys) for the secure and efficient (re-)distribution and, at times, generation of the secure channel communication keys (data encryption keys) to the communicating parties. Communication keys may be pair-wise keys used to secure a communication channel between two nodes that are in direct or indirect communications, or they may be group keys shared by multiple nodes. Network keys (both administrative and communication keys) may need to be changed (re-keyed) to maintain secrecy and resilience to attacks, failures, or network topology changes. Numerous key management schemes have been proposed for sensor networks. Most existing schemes build on the seminal random key pre-distribution scheme introduced by Eschenauer and Gligor [1]. Subsequent extensions to that scheme include using deployment knowledge [2] and key polynomials [3] to enhance scalability and resilience to attacks. These set of schemes is referred as static key management schemes since they do not update the administrative keys post network deployment.

An example of dynamic keying schemes is proposed by Jolly et al. [4] in which a key management scheme based on identity based symmetric keying is given. This scheme requires very few keys (typically two) to be stored at each sensor node and shared with the base station as well as the cluster gateways. Rekeying involves reestablishment of clusters and redistribution of keys. Although the storage requirement is very affordable, the rekeying procedure is inefficient due to the large number of messages exchanged for key renewals. Another emerging

*R.Divya is currently working in Dr.Pauls Engineering College in Electronics & Communication Engineering, Anna University, India. E-mail: [div.1484@gmail.com](mailto:div.1484@gmail.com)*

*T.Thirumurugan is currently pursuing doctorate degree from Anna University, Coimbatore, India. E-mail: [thiru0809@gmail.com](mailto:thiru0809@gmail.com)*

category of schemes employ a combinatorial formulation of the group key management problem to affect efficient rekeying [5, 6]. These are examples of dynamic key management schemes. While static schemes primarily assume that administrative keys will outlive the network and emphasize pair wise communication keys.

Dynamic schemes advocate rekeying to achieve resilience to attack in long-lived networks and primarily emphasize group communication keys. Since the dynamic scheme has the advantage of long lived network and rekeying when compared to the static schemes, the dynamic key management is chosen as a security scheme for the WSN's.

## 2. KEY MANAGEMENT SCHEMES IN SENSOR NETWORKS

The success of a key management scheme is determined in part by its ability to efficiently survive attacks on highly vulnerable and resource challenged sensor networks. Key management schemes in sensor networks can be classified broadly into dynamic or static solutions based on whether rekeying (update) of administrative keys is enabled post network deployment.

### 2.1 Static Key Management Schemes

The static schemes assume that once administrative keys are predeployed in the nodes, they will not be changed. Administrative keys are generated prior to deployment, assigned to nodes either randomly or based on some deployment information, and then distributed to nodes. For communication key management, most static schemes use the overlapping of administrative keys to determine the eligibility of neighboring nodes to generate a direct pair-wise communication key. Communication keys are assigned to links rather than nodes. In order to establish and distribute a communication key between two non neighboring nodes and/or a group of nodes, that key is propagated one link at a time using previously established direct communication keys.

### 2.2 Dynamic Key Management Schemes

Dynamic key management schemes may change administrative keys periodically, on demand or on detection of node capture. The major advantage of dynamic keying is enhanced network survivability, since any captured key(s) is replaced in a timely manner in a process known as rekeying. Another advantage of dynamic keying is providing better support for network expansion; upon adding new nodes, unlike static keying, which uses a fixed pool of keys, the probability of network capture increase is prevented. The major challenge in dynamic keying is to design a secure yet efficient rekeying

mechanism. A proposed solution to this problem is using exclusion-based systems (EBSs); a combinatorial formulation of the group key management problem

## 3. SENSOR NETWORK MODEL

Both the proposed and the existing security algorithm are based on a wireless sensor network consisting of a command node and numerous sensor nodes which are grouped into clusters. The clusters of sensors can be formed based on various criteria such as capabilities, location, communication range, etc. Each cluster is controlled by a cluster head, also known as gateway, which can broadcast messages to all sensors in the cluster. We assume that the sensor and gateway nodes are stationary and the physical location and communication range of all nodes in the network are known. Each gateway is assumed to be reachable to all sensors in its cluster, either directly or in multihop. Sensors perform two main functions: sensing and relaying. The sensing component is responsible for probing their environment to track a target/ event. The collected data are then relayed to the gateway. Nodes that are more than one hop away from the gateway send their data through relaying nodes. Sensors communicate only via short-haul radio communication. The gateway fuses reports from different sensors, processes the data to extract relevant information and transmits it to the command node via long-haul transmission.

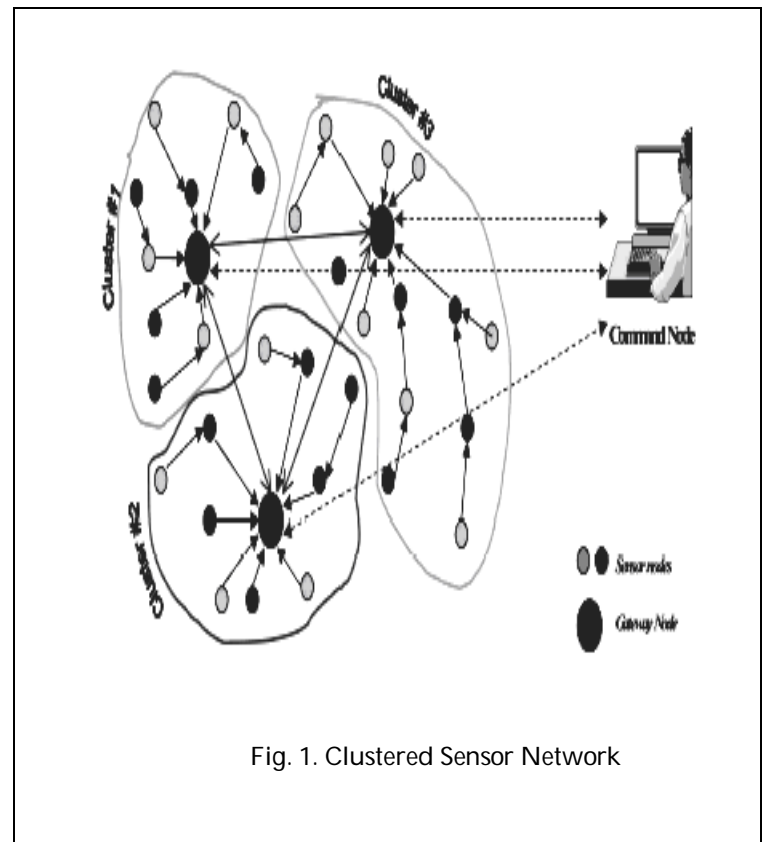


Fig. 1. Clustered Sensor Network

The network architecture is depicted in Fig. 1. Each tier of the network possesses different capabilities. The command node is resource-rich. However, the amount of traffic flowing between the command node and gateways causes the communication channel between the command node and gateways to be restrained. Most often, the command node is situated at a considerable distance from the deployment region, and might only be reachable through slow satellite links. Larger communication distances also incur increased security vulnerability and packet loss during long haul transmissions.

#### 4. COLLUSION PROBLEM

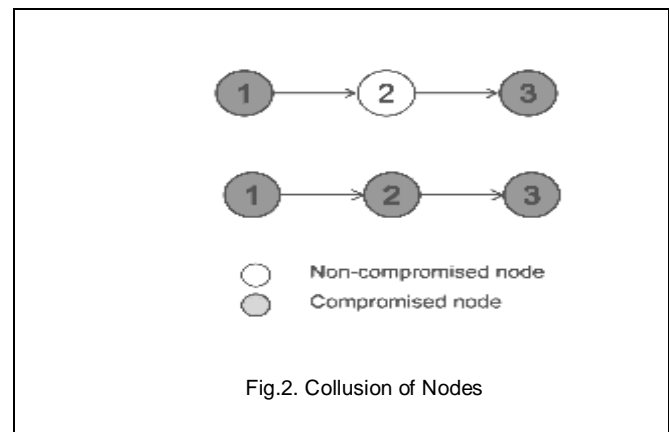
The security scheme proposed in [6] is based on the Exclusion Basis System (EBS) to address the collusion problem in EBS that performs location based key assignment to minimize the number of keys revealed by capturing collocated nodes. The network model is similar to the model developed in [6] with clusters and gateways. It uses the EBS framework to perform rekeying within each cluster. Keys are distributed to nodes by the gateways. SHELL uses post-deployment location information in key assignment; collocated nodes share more keys than nodes that are not collocated.

##### 4.1 Exclusion Basis System (EBS)

EBS is a combinatorial formulation of the group key management problem in which each node is assigned  $k$  keys out of a pool of size  $P = k + m$  keys. Rekeying takes place either periodically or when one or more nodes are captured (or suspected of being captured). Replacement keys are generated, then encrypted with all the  $m$  keys unknown to the captured nodes, and finally distributed to other nodes that collectively know the  $m$  keys. A drawback of the basic EBS-based solution is that a small number of nodes may collude and collectively reveal all the network keys. This is particularly true when the value of  $m$  is selected to be relatively small (to make rekeying feasible in terms of number of messages). EBS-based key management can be prone to collusion attacks. Two nodes collude when they share their keys with each other. In other words, colluding nodes would grow their knowledge about the network security measures. In SHELL [6], keys are reused in multiple nodes and only key combinations are unique. Therefore, it is conceivable that few compromised nodes can collude and reveal all the keys employed in the network to an adversary. Such a scenario is considered as capturing the entire network since the adversary would be capable of revealing all encrypted communications in the network. SHELL exploits the physical proximity of nodes so that a node would share most keys with reachable nodes. To

avoid the assignment of same key combinations, swapping of keys is employed. If  $S_1$  is a neighbor of  $S_2$ ,  $S_2$  is a neighbor of  $S_3$ , and  $S_1$  colludes with  $S_2$ , the resultant keys known to both of them would be [Keys of  $S_1$ ]  $\cup$  [Keys of  $S_2$ ]. Thereafter if  $S_2$  and  $S_3$  collude, the keys known to  $S_2$  and  $S_3$  would be [Keys of  $S_1$ ]  $\cup$  [Keys of  $S_2$ ]  $\cup$  [Keys of  $S_3$ ].

Thus, it can be seen that if multiple nodes collude, it is likely to uncover all employed keys. The collusion of nodes is illustrated in Fig 2. Due to swapping of keys, the key combination gets repeated at a particular time instant such that the number of nodes that share the same key combination increases.



As a result the neighboring node gets colluded if it shares the same key combinations. When the number of nodes getting colluded increases, capturing few node will reveal all the keys that has been employed in the network which results in capturing the entire network. In order to address the collusion problem in [6], an efficient dynamic key management scheme is proposed.

#### 5. DYNAMIC KEY MANAGEMENT

Due to swapping of keys the number of nodes getting colluded with the neighboring nodes is increased so capturing lesser nodes will reveal most of the keys and thus the whole network can be captured by the attacker. In order to reduce the number of colluding nodes a dynamic key assignment was chosen to employ the simple cryptography and random key distribution. Both the simple cryptography and random key distribution has its own advantages and limitations. Thus the dynamic key management scheme with the advantages of both the schemes and by taking the hamming distance into consideration is proposed as a security solution. Three methods are proposed for the dynamic key management namely

1. Simple cryptography
2. Random key distribution
3. Dynamic key assignment

### 5.1 Simple Cryptography

A simple cryptography of x-oring two keys is first tried as a dynamic key management. In this simple cryptographic scheme, each sensor node is assigned a set of keys and a unique key. Communication takes place through any one of the set of keys x-oring with the unique key. Once the encryption is over, the decryption takes place through the unique key that is known to the gateway node. The major drawback in this scheme is that the security level is low i.e. when any two key is known the other key may be revealed which results in revealing the keys of that node.

### 5.2 Random Key Distribution

Since the security level is low in x-oring of two keys, random distribution of keys is tried to enhance the security of the proposed method. In this random key distribution scheme, a set of keys is assigned to each sensor node. The communication takes place through any one of the set of keys. Once the hamming distance between any two nodes is found high the set of keys are randomly replaced and the new set of keys will be generated. Since all the keys are newly generated whenever the hamming distance is high the power consumption will be higher in this scheme and the security level is also enhanced since keys cannot be revealed by the reversing of x-or operation.

### 5.3 Dynamic Key Assignment

The proposed dynamic key assignment takes the advantage of both the simple cryptography and random key distribution scheme to reduce the collusion of nodes. In this dynamic key management algorithm, each key combination can be represented in the form of bit strings of k 1's and m 0's, where k is the number of keys stored at each node and m is the number of rekey messages required. The Hamming distance between any two combinations is defined as the number of bits that the two key combinations differ in. Let d be the Hamming distance between a pair of key combinations.

The value of d is bounded by:

$$\begin{aligned} 2 \leq d \leq 2k & \quad k < m \\ 2 \leq d \leq 2m & \quad m < k \\ 2 \leq d \leq k+m & \quad k = m \end{aligned}$$

When two nodes collude, they both will know at least d keys, since d is the number of keys that they differ in. In addition, they will also know all the keys that are common to

both nodes. The common keys are equal to  $k - d/2$ . Thus, the number of keys known to the two colluding nodes as  $k + d/2$ . This leads to the conclusion that the lower the Hamming distance (the value of d) fewer the total number of potentially revealed keys.

In this proposed dynamic key management, each sensor node is assigned a set of keys and a unique key as in the simple cryptography case. When the hamming distance between the two nodes is found high by the boundary condition, the unique key alone is changed instead of changing the set of keys and the communication takes place by using any one of the set of key x-oring with the new unique key. This method provides enhanced security with less power consumption when compared to the other two schemes.

## 6. SIMULATION

The simulations were carried out in MATLAB 6.5. The number of nodes getting colluded with time has been analyzed for both the SHELL and the dynamic key assignment and it is observed that the proposed dynamic key assignment performs better when compared to the existing method SHELL.

### 6.1 Simulation Parameters

The simulation parameters that are considered for simulation of both dynamic key assignment and SHELL are listed below:

- (i) Transmission range - 55m
- (ii) Deployment region - 850x500m
- (iii) Frequency of key renewals -  $5 \times 10^2$  sec
- (iv) Key size - 128 bits

By taking the above parameters, the simulation were performed on a network of 2000 nodes deployed in an area of (850x500) meters size setting a transmission range of 55 meters. Each of the sensor nodes is assigned unique key and a set of keys. Once the keys are assigned, the keys of the neighboring nodes are verified whether the hamming distance is low based on the boundary condition given in section 5.3. If the hamming distance of any node is found to be high, the unique key of that node is dynamically changed and the communication takes place by x-oring the new unique key with any one of the set of keys. The operation continues for the other nodes also. The simulation were carried out and the results are obtained for the following cases

Case (i) : Number of nodes colluded with time

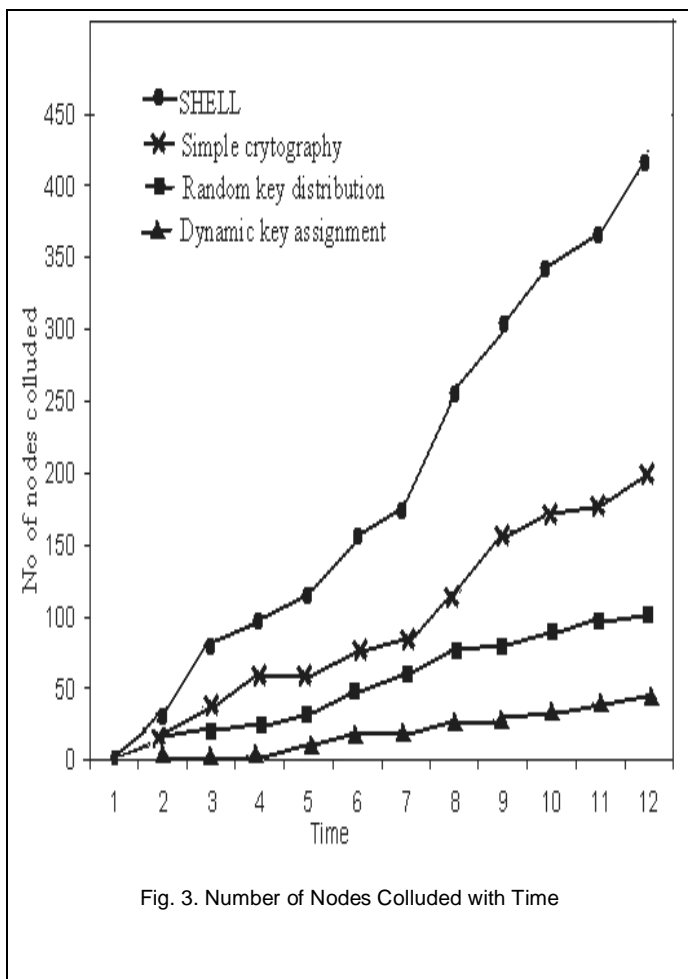
Case (ii) : Comparison of methods with static and

mobile nodes

## 6.2 Results and Discussion

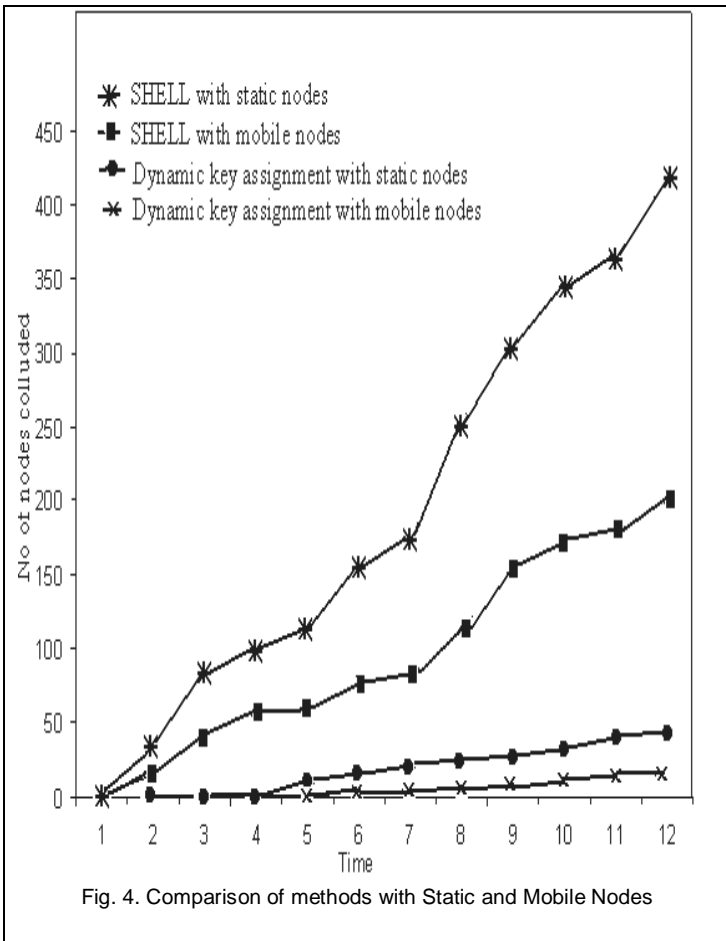
### 6.2.1 Number of nodes colluded with time

From the simulation results it is found that the number of nodes getting colluded by the dynamic key assignment scheme is reduced to a greater extent. The average number of nodes colluded is found out by varying the time and the results are plotted in Fig 3. From Fig 3 it is observed that as the number of nodes colluding with each other in the dynamic key assignment is reduced when compared to the other methods like simple cryptography, random key distribution and SHELL. The dynamic key assignment out performs the simple cryptography and random key distribution scheme as expected since it is a combination of both the schemes. The random key distribution performs better than the simple cryptography which in turn performs better when compared to SHELL.



### 6.2.2 Comparison of methods with static and mobile nodes

In this case, both the dynamic key assignment and the SHELL are compared with static and mobile nodes. The simulation setup is the same for static nodes and the mobile nodes are given mobility with a speed of 20 meter per sec with the same simulation set up. Again the number of nodes colluding with each other gets reduced in the dynamic key assignment. The number of nodes colluded when the nodes are static and mobile for both conventional and proposed scheme is obtained by simulation and plotted in fig 4. It is observed that when both the schemes are compared for static and mobile nodes, the number of colluded nodes for the mobile nodes is lesser and approaches nearly zero preventing the collusion of nodes when compared to the static nodes because the hamming distance remains the same when the nodes are static and it differs when the nodes are given mobility. Thus by preventing the collusion of nodes the dynamic key assignment provides enhanced security when compared to other existing dynamic key management schemes.



## 7 CONCLUSION

The number of nodes getting colluded with each other in the dynamic key assignment scheme is greatly reduced when compared to the other dynamic key management schemes. From Figure 3, it is observed that the proposed dynamic key management performs far better than the SHELL and from Figure 4, it is observed that by providing mobility to the nodes the collusion can be prevented. Thus the proposed dynamic key assignment prevents the collusion of nodes and provides enhanced security to the cluster based sensor network.

## REFERENCES

- [1] L. Eschenauer and V. Gligor, "A Key Management Scheme for Distributed Sensor Networks," Proc. 9th ACM Conf. Comp. and Commun. Sec., Nov. 2002, pp. 41-47.
- [2] W. Du et al., "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge," Proc. IEEE INFOCOM '04, Mar. 2004.
- [3] D. Liu and P. Ning, "Improving Key Pre-Distribution with Deployment Knowledge in Static Sensor Networks," ACM Trans. Sensor Networks, 2005, pp 204–39.
- [4] G. Jolly et al., "A Low-Energy Key Management Protocol for Wireless Sensor Networks," Proc. IEEE Symp. Comp. and Commun., June 2003, p. 335
- [5] M. Eltoweissy et al., "Group Key Management Scheme for Large-Scale Wireless Sensor Network," J. Ad Hoc Networks, Sept. 2005, pp. 796–802.
- [6] M. Younis, K. Ghumman, and M. Eltoweissy, "Location aware Combinatorial Key Management Scheme for Clustered Sensor Networks," to appear, IEEE Trans. Parallel and Distrib. Sys., 2006.