ASSURED FULL COMMUNICATION BY MERGING BLOCKSRANDOMLY IN WIRELESS SENSOR NETWORKS USINGREED SOLOMON CODE FOR KEY PREDISTRIBUTION

Pinaki Sarkar[1], Aritra Dhar[2]

[1]Department of Mathematics, Jadavpur University, Kolkata-700032, INDIA
pinakisark@gmail.com
[2]CSE Department, Gurunanak Institute of Technology, Kolkata-700114, INDIA
aritra.dhar7@gmail.com

ABSTRACT

Limited resources available to the sensors (nodes) constituting a Wireless Sensor Network (WSN) is major constraint while dealing with security of such networks. This restrict us to use symmetric key cryptography instead of public-key techniques for transmission of message amongst the nodes. In any symmetric key system, both the sender and receiver must possess the unique enciphering and deciphering key prior to exchange of message. This leads to key distribution in the sensors which in itself is a major challenge. Though there are several methods of distributing the keys, due to resource constraints, Key Predistrubution (KPD) is preferred over other techniques. It requires predistribution of keys in nodes prior to deployment and establishing immediately once deployed. However there are certain weaknesses in various existing KPD schemes. For instance, often it is not guaranteed that any given pair of nodes communicate directly. This leads one to revert to multi-hop communication involving intermediate sensor nodes resulting in increased cost of communication. In this work a key predistribution technique using Reed-Solomon codes is considered which is faced with the above weakness. The authors suggests a technique of merging certain number of sensors into blocks ensuring that the blocks have full connectivity amongst themselves. The approach also improves both time and space complexity of the system while ensuring same scalability with similar resiliency.

KEYWORDS

Communication, Security, Reed-Solomon Codes, Randomized Merging blocks, Key Predistribution.

1 INTRODUCTION

Wireless Sensor Networks (WSN) are popular ad-hoc network where the constituting sensor nodes are distributed over a large geographical area. The sensor nodes constituting a WSN communicate with each other and with the base station by radio frequency. Each tiny sensor mainly consists of (i) a wireless transceiver, (ii) a small CPU and (iii) a small battery. All these resources available to be sensors have very limited capacities in terms of (battery) power, memory, transmission capabilities etc. In spite of the resource constraints in its basic building blocks, WSNs has several military and civilian applications like detecting and monitoring enemy movement, to detect and characterize chemical, biological, radiological, nuclear, explosive materials (CBRNE),monitor traffic movement in city roads and highways.

Due to the critical functionality of WSN, communication between the nodes must be encrypted to make it immune to unauthorized accesses. Considering the architectural design, WSN can be segmented to two classes, viz.:

DOI : 10.5121/ijnsa.2011.3516

1. **Hierarchal Wireless Sensor Network(HWSN):** In this HWSN, there is a predefined hierarchy in the participating sensor nodes. There are three levels in any HWSN model. They can be chiefly catagorised into:

   - The Base Station or the Key Distribution Server (KDS) -- as the name suggests this acts as the main server as well as operational head of the network. All connection to any other network passes through this base station (or KDS). Hence optimal security has to be ensured to it.
   - Cluster Heads (CHs) -- in a HWSN, sensors are subdivided into small clusters, each having its own operational head, viz. it Cluster Head. It is responsible for connecting all nodes under its cluster to various other parts of the network. So they must given extra security as compared to the nodes.

Normally messages meant for communication between two nodes of different clusters pass through their Cluster Heads unencrypted (as plain text). However through an novel trick proposed by Sarkar et al. in [12], it has been shown that it is not a necessity that Cluster Heads get to see messages meant for their sub-nodes and not concerning them. This unique achievement owes to their emphasis that connectivity and communication of any WSN and then introducing new set of keys, called `connectivity keys'. This enables double encryption of a message while sending. The CHs possasing one set of (connectivity) keys can't fully open the doubly encrypted massage. In fact their technique also ensures selective node attack (refer [11, 4]) is completely ruled out during key establishment phase.

   - Ordinary Sensor Nodes or Sensors or Notes or Motes -- These are the basic building blocks of the network and are deployed in adversary regions. As mentioned earlier, they are very limited in their resources. These nodes are primarily responsible for gathering (sensitory) informations and communicating them to rest of the network. Providing secure communication between the nodes is the central part of research in the area of `Security of Wireless Sensor Networks'.

Corresponding to the nature of nodes involved in communication within a HWSN, three types of communication possible in HWSN. They are:

   - **Unicast** -- sensor node to sensor node,
   - **Multicast** -- group wise communication which clearly involves the sensors and the CH of any cluster or the CHs and CH or Base Station above it &
   - **Broadcast** -- base station to sensor nodes which may or may not involve the CHs.

2. Distributed Wireless Sensor Network (DWSN): In case of DWSN there is no fixed type of architecture in the sensor nodes. The topology is unknown before the deployment. The mode of communication is mainly Unicast in this case, however Broadcasting may be invoked from time to time.

1.1 Related Works and our contributions

Key predistribution in sensor networks was first considered by Eschenaur and Gligor [5]. In their work, every key is associated with an unique *key identifier*. To form the *Key rings* of the sensors, keys are *randomly* drawn from the *key pool*. *Key establishment* is also random. Such method of key predistributuion is *probabilistic* in the sense that both key distribution and establishment is done randomly. Many such *probabilistic key predistribution* schemes have been well studied and presented in a survey report published in 2005 by Çampete and Yenner [2].

For the above probabilistic approach, *shared key establishment* and *path key discovery* can

become very difficult. Lee and Stinson proposed two schemes [6, 7, 8] where they have adopted combinatorial techniques for predistribuion and later establishment of keys. Their works also suggests that both *shared key establishment* and *path key discovery* can be better achieved by the suggested *deterministic approach*. Some other deterministic schemes have been proposed by Ruj and Roy using various combinatorial designs like PBIBD and Transversal Designs in their works [9, 10] respectively. Very recently unique factorization of polynomials over Finite Fields has been invoked by Sarkar and Chowdhury [16] to give a KPD scheme while Bag and Ruj [1] utilizes Affine plane geometry over Finite field for similar purpose.

Hybrid key predistribution scheme by Merging block technique in WSN was first proposed by Chakarabarti et. al. [3]. Their merging block technique was based on transversal design proposed by Lee and Stinson [6, 7]. Transversal design proposed by Lee and Stinson [6, 7] has a major drawback i.e., the absence of full communication hence intermediate nodes were incorporated which increase overall system overhead. Here nodes were merged in random fashion to get new nodes. The objective was to increase number of common keys between any two given new (merged) nodes and achieve full communication within the system.

There are other schemes with similar drawback. For instance a key pre-distribution scheme using Reed-Solomon code with parameters (n, q^k, d, q) was proposed by Ruj and Roy[11]. The authors of [11] has established the number of common key between any two given nodes are at most $k-1$. Thus in their scheme there is a high probability that there may not exits any common keys between any given pair of nodes hence no direct communication.

In this paper we apply the merging block scheme on the nodes in WSN where key-predistribuion is done by Reed-Solomon code. Using this merging block technique one can observe the increment of common keys between any pair of given merged block hence increase the probability of direct communication. This enhances system efficiency in greater extent.

Of course inability of direct communication is not the only deficiency of Ruj & Roy's scheme in [11]. These schemes like most KPD schemes are faced with a serious problem of *selective node attack* during key establishment phase. In their recently published pioneering work, Sarkar et al. [12] has develop a novel black box technique which ensure security against this form of attack. They have theoretically established that their method enhances security of the overall messaging immensely. Of late this technique have been used to by Sarkar & Saha in [13], Bag, Saha & Sarkar in [14, 15] to improved schemes proposed in [9, 1, 6, 7, 8] respectively.

2 PRELIMINARIES

This section is devoted to describing various preliminary aspects that we shall use while designing our key predistribution scheme. As described earlier, we shall develop our merging block design based on a KPD scheme proposed by Ruj and Roy [11] that uses Reed--Solomon codes. Hence after briefly stating the basics of BIBD(Balanced Incomplete block diagram) designs in section 2.1, we move on to describing their scheme in section 2.2 and then point out a potential weakness in their proposed scheme in section 2.3.

2.1 Combinatorial design: BIBD

Detailed explanation of Balanced Incomplete Block Designs (BIBD can be found in combinatorics books like the one written by Stinson [17]. Such designs are useful for constructing key predistribution (KPD) schemes in Wireless Sensor Networks (WSN). A brief outline is presented here. Let X be a set and A be the finite set of subsets (also known as block) of X. The pair (X,A) is known as s *set system* or *design*. The degree of a point $x \in \mathsf{X}$ is the number of subsets containing the point x. (X,A) is said to be uniform of rank k if all its

subsets i.e. the blocks has same size k. If all points have same degree r then (X,A) is said to be regular of degree r. A regular and uniform set system is known as a $(v,b,r,k)-1$ design where $|X|=v,|A|=b$, r is the degree and k is the rank. The condition $bk = vr$ (refer [6, 7] is necessary and sufficient for existence of such a system. If any two distinct block intersect in *zero* or *one* point then $(v,b,r,k)-1$ is known as a (v,b,r,k) design.

2.2 Key Predistribution using Reed Solomon code

Consider a (n,q^k,d,q) Reed Solomon code having alphabet in the finite field F_q for $q>2$. The length of the code is $n=q-1$, distance is $d = n-k+1$ and dimension is k. The number of codeword is $M=q^k$. When this code is mapped to a Wireless Sensor Network, number of node in the network is q^k each having $q-1$ number of keys. The number of common keys between any two nodes is $n-d = k-1$. For any codeword $x=(a_1,a_2,\ldots,a_n)$, the keys assigned to the node x are $(a_1,1),(a_2,2),\ldots,(a_n,n)$. The key pool consists of qn number of keys $\{(a_i,i): a_i \in F_q, i=1,2,\ldots,n\}$. Let F_q be a finite field of $q>2$ elements. Let ρ be the set of polynomials over F_q of degree at most $k-1$. Thus $|\rho|=q^k$ Let $F_q^* = \{\alpha_1,\alpha_2,\ldots,\alpha_{q-1}\}$ be the set of non-zero elements of F_q. For each polynomial $\rho_i(x)\in\rho$, it is defined $c_{p_i} = (\rho_i(\alpha_1),\rho_i(\alpha_2),\ldots,\rho_i(\alpha_{q-1}))$ to be the i-th codeword of length $q-1$. It is defined that $C=\{c_{p_i} : p_i(x)\in\rho\}$. so, C is a Reed Solomon code.

A sample network having 16 nodes is present in figure 1.

2.3 Weakness : Motivation of our work

Among several other weaknesses we figure out a potential weak point i.e. lack of full communication within the above mentioned scheme using Reed Solomon codes for key predistribution. This clearly indicates that there are several possibilities that direct communication between any pair of given node is not possible, which may lead to hopping node incident causing increased system overhead.

3 REMEDIAL STRATEGY : MERGING BLOCK IN COMBINATORIAL DESIGN

Merging block technique is a novel trick to overcome the drawbacks imposed by the KPD using Reed Solomon code. In this merging block technique several blocks are merged together randomly to form a new node. Here the model is flexible enough that one can mention the number of blocks to merged together randomly (here denoted by z). This technique causes increment of keys in newly formed node, which ensures increment of probability that any given pair nodes can communicate directly. Details of the technical results are discussed bellow.

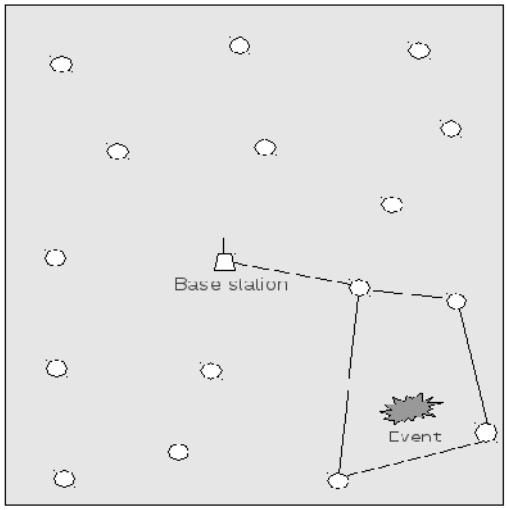


Figure 1: WSN based on KPD using Reed Solomon codes where $q = 4, k = 2$

This KPD design prosed in [11] can be easily checked to be a $(v, b, r, k)-1$ BIBD having configuration $v = rk$ and $b = q^2$. Here z blocks are merged together to form a node where

1. Number of sensor nodes $N = \lfloor \frac{b}{z} \rfloor$
2. Probability that any two nodes share no common key is $(1-p_1)^{z^2}$ where

$$p_1 = \frac{\sum_{i=1}^{k-1} (-1)^{i-1} q^i \binom{q^{k-1}}{2} \binom{q-1}{i}}{\binom{q^k}{2}}$$

3. The expected number of keys shared between two nodes is $z^2 p_1$
4. Each node will contain M many distinct keys, where $zk - \binom{z}{2} \le M \le zk$.

Theaverage value of M is $\hat{M} = zk - \binom{z}{2} p_1$.

5. The expected number of links in the merged system is

$$\hat{L} = \left(\binom{b}{2} - \lfloor \frac{b}{z} \rfloor \binom{z}{2} \right) p_1 - bk(mod z)$$

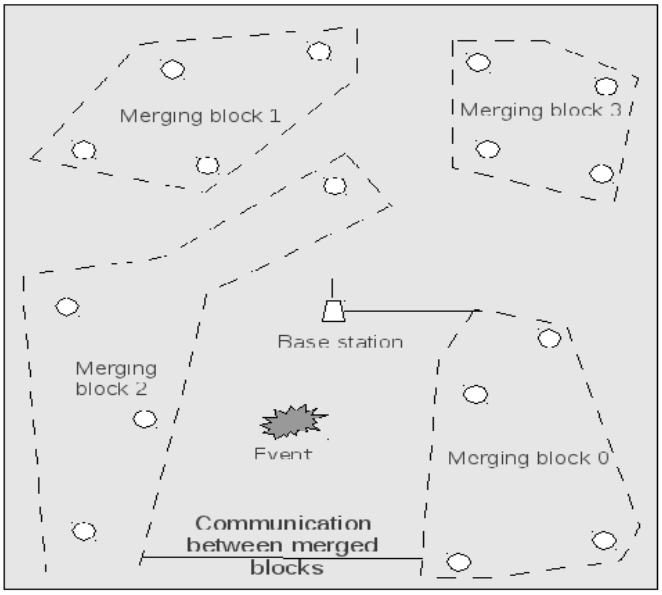
6. The key will be present in Q many nodes where $\lceil \frac{r}{z} \rceil \le Q \le r$. The average value of Q is $\hat{Q} = \frac{1}{kr} (\lfloor \frac{b}{z} \rfloor)(zk - \binom{z}{2} p_1$

In this merging block technique we use a probabilistic i.e. a random merging technique. This approach mainly involves two steps

1. First the key predistribution is done using Reed-Solomon code
2. Then randomly some number of nodes (usually this number is denoted with z) are merged together which makes new merged nodes.

After the blocks are merged randomly, in a newly merged node there may be several repeating keys. This repeating keys indicate the possibility that intra-node communication may happen. In this random merging approach, one can't ensure that the nodes that are being merged will not share a common key. To eliminate the possibility of intra-node communication, one trick is to take all the keys once while merging. As such we can make use of a heuristic suggested by Chakraborti et. al [3, section 4] to address this issue.

A typical picture of merging block design corresponding to network of figure 1 on 16 nodes with $z = 4$ is given in figure 2.



Wireless Sensor Network KPD using merging block on Reed Solomon code KPD where q=4. k=2 and z=4.

Figure 2: Same WSN using merging block over Reed Solomon code where $q = 4, k = 2, z = 4$.

4 KEY ESTABLISHMENT : MERGED BLOCK FORMATION

Here blocks are merged in randomized manner. Let $array\_node[0: q^k - 1]$ denotes the array of node id and $array\_key[0: q^k - 1][0: q - 1]$ be the array containing all keys of every node. Then key establishment is achieved according to the following algorithm 1:

Start of algorithm 1.

Take a value z(the no of nodes to merge together to form a new node).
Calculate $N := \left\lfloor \frac{q^k}{z} \right\rfloor$.
Take a new array $array\_node\_random[0: q^k - 1]$.
Randomize elements of $\_node[0: q^k - 1], array\_node\_random[0: q^k - 1] := array\_node[0: q^k - 1]$.
Take a new array $array_node_merge[0: N][0: z]$take z number of elements from$array_node_random[0: q^k - 1]$and store it to $array\_node\_merhe[0: N][0: z]$.
ifKey from the array $array_key[0: q^k - 1][0: q - 1]$is not present in $array\_key\_merge[0: N][0: z][0: q - 1]$**then**
Find the keys of the nodes in $array_node_merge[0: N][0: z]$from the array$array_key[0: q^k - 1][0: q - 1]$and store it to to a new array $array\_key\_merge[0: N][0: (q - 1) * z]$.
else
Skip it and move to next key.
end if
The array $array_key_merge[0: N][0: (q - 1) * z]$contain keys of all merged block where no intra-node common key is present.
End of algorithm.

5 COMMUNICATION

After the blocks are merged, communication between new nodes takes place. Here the Reed-Solomon code is taken as (n, q^k, d, q). The number of common key between any two given nodes at most $k-1$, i.e., varies from 0 to $k-1$. When the sensors are merged to form big blocks of z many sensors each, number of common keys between any pair of given nodes increases which greatly increases probability of direct communication between any pair of given nodes. Communication testing algorithm between any given pair of nodes is discussed bellow.

Start of algorithm 2.

Take input id_1 and id_2which denotes $node\ id$ of a pair of nodes.
Initialize flag as flag = 0.
fori=0:(q-1)*z **do**
  forj=0:(q-1)*z **do**
  if$array\_key\_merge[id_1][i]$ = $array\_key\_merge[id_2][i]$**then**
  flag := flag + 1
  end if
  end for
end for
ifflag! = 0 **then**
Direct communication is possible.
else
Direct communication is not possible
end if
End of algorithm.

6 COMMUNICATION PROBABILITY

The term *Communication probability* denoted by $\rho_c (also\ denoted\ by\ \rho_1)$ of the network is the probability that two nodes are connected i.e. the share at least one common key. As in our case, in the Reed-Solomon code $k = 2$ is considered. So, if two nodes are connected then maximum number of common key between them is one. We know that total number of nodes in the network is q^k and each node contains $q-1$ no of keys. So, in the network total $\binom{q^k}{2}$ links are possible. So, we can state that:

$$\rho_c = \frac{Number\ of\ links\ present\ in\ the\ network}{Total\ number\ of\ links\ in\ the\ network}.$$

From Ruj and Roy we get the value of ρ_c in their system as:

$$\rho_c = \frac{\sum_{i=1}^{k-1}(-1)^{i-1} q^i \binom{q^{k-1}}{2}\binom{q-1}{i}}{\binom{q^k}{2}}.$$

However, the following *Theorem 1* established that in our system $\rho_{cmb} = 1$ for $z \geq 4$. Recall we are merging z blocks of their model randomly. Since we are talking about communication probability of two different systems, we are making use of the notation ρ_c for Ruj & Roy's [11] (old) system while ρ_{cmb} denotes the communication probability of our (new) system based on their's [11].

Theorem1: When z random blocks (nodes) of a WSN designed on basis of Reed Solomon codes having $k = 2$ as suggested in [11] are merged to get bigger nodes (refer section 3), then the new system has communication probability $\rho_{cmb} = 1 <=> z \geq 2$.

Proof: Here, we consider $k = 2$ which ensures that number of common keys between any given pair of nodes are either zero or one. Such a network consists of a maximum of q^2 nodes with each of them containing $q-1$ number of keys. When z numbers of nodes are merged together to form a bigger merged node, then each merged node is *expected* to contain $z(q-1) - \binom{z}{2} p_c$ number of distinct keys.

Now let us consider$z = 2$. We shall show full communication among the merged node is assured. Our argument will also show that the converse is true.

Recall that one node in the network has $q-1$ number of keys and when $k = 2$, each of the keys can be found in q number of nodes. So, each node can communicate with a total of $(q-1)^2$ number of distinct node in the network. As, the network contains a total of q^2 nodes, clearly there are $q^2 - (q-1)^2 = 2q-1$ nodes having no communication link(s) with this (first) node. For the second node in the merged block, it also has $q-1$ number of keys. 2 cases arises:

• The second node has a common key with the first node. Thus the minimum number of distinct new keys contributed by the second node is $(q-1)-1 = q-2$.

• Since each of these new keys is shared between '*q*' distinct nodes, we are assured that each of them will link these two nodes with '*q – 1*' distinct new nodes not previously connected.

• Now consider any '3' new keys of the 2nd node (not in old node). They link to '*3(q – 1)*' many new nodes.

Since the 1st node was not connected to only *2q – 1* nodes, this means the new keys of the second not only covers up for all these nodes not connected to 1st node but also connects the 2nd node to many other nodes which were connected to 1st node. Thus the combination of any two randomly chosen nodes is sufficient for full connectivity. Hence the theorem and title of the paper. Q.E.D.

7 Resilience

Under adversarial situation, one or more numbers of sensor nodes may get compromised. In that case, all the keys in the node(s) get exposed. They can't be used in the secret communication any longer. Links which are connected by those exposed keys will be broken. When communication links are broken, communication may still exists using alternative paths. Now, another situation may takes place. Let there is node which have all keys compromised. Then the node will get *disconnected*. Node disconnection is a fatal situation as there is no way to communicate with the disconnected node. After the nodes get compromised, one has to calculate the proportion of links broken i.e. the links can not be used any further. This proportion is denoted by $E(s)$. Thus,

$$E(s) = \frac{Numbers\ of\ links\ disconnected\ when\ s\ nodes\ are\ compromised}{Total\ number\ of\ links\ before\ compromise}$$

From the paper of D. Chakrabarti et al., in merging block technique the calculation of $E(s)$ is as bellow.

$$E(s) = \frac{\sum_{i=1}^{z^2} \frac{\binom{\gamma}{i}}{\binom{q(q-1)}{i}} \binom{z^2}{i} p_1^i (1-p_1)^{z^2-i}}{1-(1-p_1)^{z^2-i}} \quad where\ \gamma = sz(k - \frac{(sz-1)p_1}{2})$$

7.1 Calculation of $E(s)$

Let $N_1 \& N_2$ be any 2 given merged nodes. Consider two events $E \& F$ as follows:

1. E : N_1 and N_2 are disconnected after the failure of s number of nodes,
2. F : N_1 and N_2 were connected before the failure of those s nodes.

Then we can clearly see that

$$E(s) = P(E \mid F) = \frac{P(E \cap F)}{P(F)}.$$

Now let X be a random variable denoting the number of common keys between N_1 and N_2. Thus we may assume that X follows $B(z^2 p_1)$, i.e. it follows Binomial distribution in accordance to the assumption made in Chakrabarti et. al. [3, section 3]. Thus,

$$P(F) = P(X > 1) = 1 - P(X = 0) = 1 - (1-p_1)^2.$$

Next we can consider two events:

1. E_{1i} : i number of keys (shared between N_1 and N_2) areÂ revealed consequent upon the failure of s nodes,
2. E_{2i} : i number of keys are shared between N_1 and N_2.

Let $E_i = E_{1i} \cap E_{2i}$ for $i = 1,2,\ldots,z^2$. So, $E_i \cap E_j = \phi$ for $0 \le i \ne j \le z^2$. As $E \cap F = \bigcup_{i=1}^{z^2} z^2$, we have $P(E \cap F) = P(\bigcup_{i=1}^{z^2} E_i) = \sum_{i=1}^{z^2} p(E_i) = \sum_{i=1}^{z^2} P(E_{1i} \mid E_{2i}) P(E_{2i})$ and $P(E_{2i}) = \binom{z^2}{i} p_1^i (1-p_1)^{z^2-i}$.

As in Chakrabarti et. al. [3, section 3.2] we estimate $P(E_{1i} \mid E_{2i})$ by hypergeometric distribution. In this merging block technique the size of the key pool is $q(q-1)$. Let γ denotes the number of revealed distinct keys in a node then

$$\gamma = sz(k - \frac{(sz-1)p_1}{2}).$$

So, $P(E \mid F) = \dfrac{\binom{\gamma}{i}}{\binom{q(q-1)}{i}}$ Finally we can get the value of $E(s)$.

8 EXPERIMENTAL RESULTS

Simulation results for $E(s)$ are presented in table 1 and table 2 compares our results with Ruj & Roy [11] where $k = 2$ is assumed for a network with $N = 2401$ nodes. Thus in our case we take $N = 2550$ nodes. In both cases we have assumed $s = 10$ nodes have been captured. Their communication probability $\rho_c = p_1 =$ the expected number of keys for a given pair of nodes. In the tables `RS' means Reed-Solomon scheme that has been presented in [11] while `MB' means the present scheme. In the experiment we considered $q = 49$. So, total number of nodes in the network is $49^2 = 2401$. Now $z = 4$ i.e 4 nodes are merged together to form a new merged node. This renders the following scheme.

1. In this case number of merged sensor nodes in the network is $\lfloor \frac{2401}{4} \rfloor = 600$.
2. Probability that two nodes do not shares a common key is 0 $(recap\, theorem 1)$.
3. Number of keys shared between two nodes are either 0 or 1 (as here $k = 2$ is considered).
4. Each node will contain $\hat{M} = 4 \times 48 - \binom{4}{2}\frac{48}{50} = 186$ number of keys.
5. $E(10) = 0.3164$ and $E(5) = 0.2109$.

Table 1: Simulation results for $E(s)$ for $N = 600, 2550$ as *s=number of nodes captured* varies.

N	$s = 3$	$s = 4$	$s = 5$	$s = 7$	$s = 8$	$s = 9$	$s = 10$
600	$s = 0.1697$	0.1932	0.2109	0.2527	0.2715	0.2931	0.3164
2550	0.0657	.08374	0.1024	0.140	0.1592	0.1808	0.1997

Table 2: Comparative results for $E(s)$ between our scheme and RS scheme in [11] having about $N = 2400$ nodes with $z = 2$when $s = 10$ node capture.

Comparison of our design with RS design of [11]	Merging Block (MB) design over RS design of [11]	Reed – Solomoncode (RS) design as in [11]
Number of nodes in the network (N)	2550	2401
Number of keys per node	394	48
Communication probability	1 (by *theorem 1*)	0.52
$E(s)$ for $s = 10$	0.19996	0.18656

9 CONCLUSION

In this paper a block merging technique is presented which is applied on key predistribution strategy using Reed Solomon code. Key predistribution using reed Solomon code has several drawbacks. In several situations direct communication is not possible when there are no common keys between given pair of nodes. This causes indirect connection or hopping using a intermediate node which increases system overhead. System overhead is pretty costly in such systems where the system components (here the wireless sensor nodes) are constraint to a certain limit of resources(processing power, memory and power supply). The merging block scheme in this paper resolves this performance overhead greatly by increasing number of common keys between any two given merged nodes while eliminating intra-node communication by reducing intra-node common keys. Here in this paper the main objective is to achieve full communications within the network keeping security intact if in some case some nodes get compromised. This block merging strategy provides a very robust network ensuring full communication.

10 FUTURE WORK

In this merging block technique on *Reed Solomon* code is purely randomized. Whenever the blocks are merged (where z is the number of nodes to merged together to form a new node) it is impossible to determine participating blocks in a particular merged node (or block). So, the control over this kind of model is pretty low. Moreover this whole merging is done during key establishment. So, from the system administrator's point of view, this can be a fatal situation. This only can be resolved by a *deterministic* merging block technique. Because in deterministic approach only those nodes are included in the newly formed node such that it generates no intra-node common key. So, it can be clearly find out that deterministic approach is only the way to tackle this minimized controlling factor.

Another future aspect of this randomized merging block technique is to tackle the primitive requirement of Wireless Sensor Network (WSN) i.e. to reduced $E(s)$ (mentioned earlier in this paper) which improves resilient factor. In this probabilities merging block technique as the number of common keys between any given pair of nodes is increased so there is a high chance of intra node communication. So a better took is desirable to decrease the $E(s)$ factor. As such if the blocks can be merge deterministically prior to deployment, then we can think of applying a

novel black box scheme suggested recently by Sarkar et. al. [12] to the merged block design and obtain much better resiliency.

Other than this, its is important to look KPD having full connectivity, high resiliency and is equally scalable. In regards some Algebraic, Combinatorial or other Mathematical techniques may be useful as has been proposed by Sarkar and Chowdhury in [16] and Bag and Ruj in [1].

ACKNOWLEDGEMENT

We would like to heartily thank Ms. Amrita Saha, IIT, Bomay for discussion various aspects of the paper. A special word of appreciation goes to Prof. Subhamoy Maitra, ISI, Kolkata for motivating us to use their random block idea in the present case.

We want to also express our gratitude to University Grants Commission of India forfinancially supporting the doctoral program of Mr. PinakiSarkar. This work is meant tobe a part of the doctoral thesis of Mr. PinakiSarkar.

This work is throughly extended and revised work of our paper entitled " Full Communication in a Wireless Sensor Network by Merging Blocks of a Key Predistribution using Reed-Solomon Code" published at CCSEA 2011. ***Theorem 1***of **section6** (Communication Probability) makes this extension very special as it assures full connectivity among the newly constructed nodes formed out of merging. Thus finally we thank organizors, PC commetee and editorial body of CCSEA 2011for giving us this oppurtunity to present a more detailed work.

REFERENCES

[1] S. Bag and S. Ruj. Key Distribution in Wireless Sensor Networks using Finite Affine Plane. *IEEE Computer Society AINA-2011*, pp. 436-442, 2011.

[2] S. A. Çamtepe and B. Yener, Key distribution mechanisms for wireless sensor networks:A survey 2005. Technical Report, TR-05-07 Rensselaer Polytechnic Institute, Computer Science Department, March 2005.

[3] D. Chakrabarti, S. Maitra and B. Roy, A key pre-distribution scheme for wireless sensor networks: merging blocks in combinatorial design, *International Journal of Information Security*, vol. 5, no. 2, pp. 105--114, 2006.

[4] R. Di Pietro, L. V. Mancini, A. Mei, Energy efficient node-to-node authentication and communication confidentiality in wireless sensor networks, *Wireless Networks 12(6)*, pp. 709-721, 2006.

[5] L. Eschenauer and V. D. Gligor,A key-management scheme for distributed sensor networks, *ACM Conference on Computer and Communications Security*, pp. 41--47., 2002.

[6] J. Y. Lee and D. R. Stinson, Deterministic key predistribution schemes for distributed sensor networks, *Selected Areas in Cryptography*, ser. Lecture Notes in Computer Science, pp. 294--307, Springer, 2004.

[7] J. Y. Lee and D. R. Stinson, A combinatorial approach to key predistribution for distributed sensor networks, *IEEE Wireless Communications and Networking Conference, WCNC 2005*, New Orleans, LA, USA, 2005.

[8] J. Y. Lee and D. R. Stinson,On the construction of practical key predistribution schemes for distributed sensor networks using combinatorial designs, *ACM Trans. Inf. Syst. Secur.*, 11(2), 2008.

[9] S. Ruj and B. Roy, Key predistribution using partially balanced designs in wireless sensor

networks, *ISPA 2007*, ser. Lecture Notes in Computer Science, Springer, Heidelberg, pp. 431--445, 2007.

[10] S. Ruj and B. Roy, Revisiting key predistribution using transversal designs for a grid-based deployment scheme, *International Journal of Distributed Sensor Networks*, IJDSN 5(6), pp:660--674, 2009.

[11] S. Ruj and B. Roy, Key Predistribution Schemes Using Codes in Wireless Sensor Networks, *Inscrypt 2008, LNCS 5487, Springer-Verlag Berlin Heidelberg.*, pp. 275--288, 2009.

[12] P. Sarkar, A. Saha and M. U. Chowdhury. Secure Connectivity Model in Wireless Sensor Networks Using First Order Reed-Muller Codes, *MASS 2010*, pp. 507--512, 2010.

[13] P. Sarkar and A. Saha. Secure Connectivity Model in Wireless Sensor Networks Using First Order Reed-Muller Codes, *MASS 2010*, pp. 507--512, 2010.

[14] S. Bag, A. Saha and P. Sarkar. Highly Resilient Communication Using Affine Planes For Key Predistribution And Reed Muller Codes For Connectivity InWireless Sensor Network, *The Third International Conference on Wireless & Mobile Networks (WiMo-2011)*, to be published, 2011.

[15] S. Bag, A. Saha and P. Sarkar. Highly Resilient Key Predistribution Scheme Using Transversal Designs And Reed Muller Codes For Wireless Sensor Network, *The Fourth International Conference on Network Security & Applications (CNSA-2011)*, to be published, 2011.

[16] P. Sarkar and M. U. Chowdhury. Key Predistribution Scheme Using Finite Fields And Reed Muller Codes, *Accepted in SNPD 2011. Recommended for publication in 'Springerâ€™s Studies in Computational Science'*, Springer, 2011.

[17] D. R. Stinson Combinatorial Designs: Constructions and Analysis, *Springer-Verlag, New York*, 2003.

Authors

PinakiSarkar* is currently doing his Ph.D. from Jadavpur University in collaboration with cryptology group of ASU,ISI, Kolkata. Earlier he graduated with Mathematics honours from St. Xaviers' College, Kolkata and did his masters with Mathematics from CMI, Chennai. His subjects of interests are Algebra, Number Theory, Coding Theory, Combinatory, Cryptology and Wireless Sensor Network Security.

AritraDhar is pursuing his B.Tech. in Computer Science and Engineering enrolled in Gurunanak Institute of Technology, Kolkata which is affiliated to West Bengal University of Technology.