Secure Communication of Wireless Sensor Network (WSN) using Coding Theory

Kunal Hui
Department of Computer Science and Engineering
Haldia Institute of Technology
Haldia-721657, India.
Email:kunalhui@gmail.com

Abstract-*Several schemes to establish secret key between sensor nodes in wireless sensor network for secure communication has been proposed. But due to lack of priori deployment knowledge and limited resources of sensor nodes and security threats, it is always challenging to propose a better scheme apart from existing. In this paper I have suggested a novel idea for secure communication in wireless sensor network* (WSN) *using BCH code. Here communication key and connectivity key are used separately for more secure communication. I have created connectivity key using BCH code. In this scheme a new approach has been proposed to establish a secure communication between sensor nodes with desired scalability and resiliency. This scheme also support the mobility of sensor nodes.*

Keywords: *Key predistribution, Connectivity Key, Communication Key, BCH Code, Reed Solomon Code, Mobility.*

I. Introduction

Usage of sensor nodes in several fields is growing rapidly as it is very economical. Wireless Sensor Network (WSN) has several applications like environmental monitoring(i.e. Air pollution monitoring, Forest fire detection, Greenhouse monitoring, Land slide detection), Industrial monitoring. WSN also has several applications in military field like explosive detection, collecting information about enemy movement etc. It has many civil application like traffic control.

In wireless sensor network thousands of such sensor nodes are distributed over an area and nodes sense surrounding data of that area. These data needs to transmit to other neighbor nodes or to base station via several intermediate nodes securely. That means all data should be encrypted before transmitting. As sensor nodes are resource constrained, heavy computation is not desirable. In general sensor nodes consists of four basic units like processing unit, sensing unit, transceiver unit and power unit. Though WSN has several applications it faces some challenges like wireless nature of communication, resource limitation, very large and dense WSN, lack of fixed infrastructure, unknown network topology prior to deployment, high risk of physical attack.

When secure communication between sensor nodes is required it uses private key cryptosystem instead of public key cryptosystem. Because public key cryptosystem requires huge computing resource which a sensor node cannot afford. That is why we need to predistribute the keys (i.e keys are uploaded into the nodes before deployment) into the sensor nodes. In this scheme have used such a scheme suggested by Ruj and Roy[1] for communication key and BCH code for connectivity key.

This scheme is applicable for hierarchical network as well for distributed WSN. Here every sensor node is given a connectivity key using BCH code for more secure communication. By doing so I have maintained resiliency and scalability of WSN. And increased the connectivity with more security.

A. Related Works

In Eschemauer and Gligor's[2] scheme the keys are drawn randomly from key pool. A code based key management system was proposed by Al-Shurman and Yoo [3]. Where a matrix along with a vector is used for generating codeword. Camtepe and Yener [4] proposed a deterministic key predistribution scheme using combinatorial designs. [5]shows a survey of different pre key-distribution schemes for Distributed Sensor Networks.

Ruj and Roy[1] proposed a scheme where Reed Solomon code is used for key pre-distribution and each node is given a unique polynomial. Key Pre-distribution Schemes in Distributed Wireless Sensor Network using Combinatorial Designs ware revisited by Pattanayak and Majhi[6].

Sarkar , Saha and Chowdhury[7] proposed a scheme where communication and connectivity model is introduced. Where if two node want to communicate with each other then it has to match both of the connectivity key and communication key. But here node movement from one sub network to another sub network or to any location of the network is not possible. In case of node movement results disconnected node. For making the nodes connected automatically with new sub network or with the existing network need redistribution of keys. We have to redistribute the keys if an existing node need to be moved from one sub network to another sub network . This is not desirable because sensor node are moveable. In this scheme cluster head decides using which path the nodes will communicate with each other. Cluster head may be kept more secure but if the cluster head became compromised then all the nodes under cluster head will be compromised.

B. My Contributions

I have tried to remove above problems by giving a connectivity key to each node which denotes which nodes are within the communication region of a certain sensor node. The connectivity key is generated using BCH code. Sensor nodes within same communication region will be connected to each other with connectivity key. Here is no concept of cluster head. So there is no possibility of compromised hole network under cluster head. Every node is responsible for its own communication. The existing sensor nodes can be moved different location of the network without redistributing the keys has shown later. And communication of sensor nodes is made more secure using this connectivity key.

C. Organization of the paper

The rest of the paper is organized as follows. In Section II I have discussed about communication process of sensor nodes and how communication key and connectivity key works. In section III I have defined how the codewords are generated from BCH code and how new nodes are inserted to the existing WSN. Also discussed about the mobility of the sensor nodes. Scalability and Resiliency are discussed according to my proposal in Section IV and V. And in Section VI I concluded with some future work.

II. Communication Process of Sensor Nodes

Every sensor node has a communication region. Sensor node can communicate within their communication region. In case they are not in the same communication region then multi-hop communication is done. If two node want to communicate with each other then first it will match the connectivity key and then communication key. If both key matches then nodes can communicate with each other.

A. Connectivity Key

Connectivity key describes the connection of sensor nodes. Connectivity key is introduced by Sarkar , Saha and Chowdhury[7]. The details will be explained in section III.

B. Communication Key

In this scheme it preferable to use scheme proposed by Ruj and Roy[1] based on Reed Solomon Code for key predistribution for communication key. One can use any other scheme. According to Ruj and Roy[1] the communicating nodes must be in the same communication region and share at least one secret key. In case they are not in the same communication region then multi-hop communication is done.

Each sensor node within same network is given a unique connectivity polynomial and some communication key generated by Reed Solomon Code (scheme proposed by Ruj and Roy [1]). If two nodes want to communicate with each other than first it has to match the connectivity polynomial. If minimum one bit position matches between the connectivity polynomials that means they are connected and then it has to match the communication key. Only if both connectivity key and communication key matches then nodes can communicate with other.

Figure 1 shows how sensor nodes communicate with each other using our proposed scheme. This network consists of nodes A, B, C, D, E and F. Suppose these nodes are within their communication region. Each node is given some connectivity key and some communication key. For example node A is given the connectivity polynomial (Bit representation) 1001 and a set of communication key $(1, 1), (1, 2), (1, 3)$, node B is given the connectivity polynomial 1110 and set of communication key $(3, 1), (3, 2), (3, 3)$ and so on. Lines shows the connection (i.e. at least one bit position matches between the connectivity polynomials.) between the nodes. Arrow line shows direct communication of nodes. Node B and E can communicate with each other. Because they have same bit value in the 1st and 3rd bit position and they share same communication key $(3, 2)$. Though node A and D have same communication key they cannot communicate with each other. Because the connectivity polynomial does not matches. Again node C and D cannot communicate with other because they are not sharing same communication key.

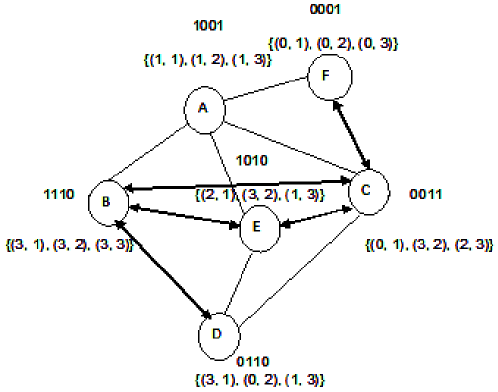


Fig. 1. Communication of nodes

III. Connectivity Key Using BCH Code

In coding theory[8] the BCH codes[9,10] form a class of parameterized error-correcting codes. This code is a powerful random error-correcting cyclic code. BCH codes were invented in 1959 by Hocquenghem, and independently in 1960 by Bose and Ray-Chaudhuri . The acronym BCH comprises the initials of these inventors' names.

Fix a finite field GF(q), where q is a prime power. Also fix positive integers m, n, and d such that $n = 2^m - 1$ and construct a polynomial code over GF(q) with code length n, whose minimum Hamming distance is at least d. The BCH(n,k) is defined as any positive integer $m(m \geq 3)$ and $t(t < 2^{m-1})$ and the following parameters

Block Length : $n = 2^m - 1$

Number of parity$-$check digits : $n - k \leq$ mt

Minimum distance : $d_{min} \geq 2t + 1$

Block id	Polynomial	Connection Bit Representation
0	0	0 0 0 0
1	1	1 0 0 0
2	α	0 1 0 0
3	α^2	0 0 1 0
4	α^3	0 0 0 1
5	$1+\alpha$	1 1 0 0
6	$\alpha+\alpha^2$	0 1 1 0
7	$\alpha^2+\alpha^3$	0 0 1 1
8	$1+\alpha+\alpha^3$	1 1 0 1
9	$1+\alpha^2$	1 0 1 0
10	$\alpha+\alpha^3$	0 1 0 1
11	$1+\alpha+\alpha^2$	1 1 1 0
12	$\alpha+\alpha^2+\alpha^3$	0 1 1 1
13	$1+\alpha+\alpha^2+\alpha^3$	1 1 1 1
14	$1+\alpha^2+\alpha^3$	1 0 1 1
15	1+α^3	1 0 0 1

TABLE I
CONNECTIVITY POLYNOMIAL AND BIT REPRESENTATION

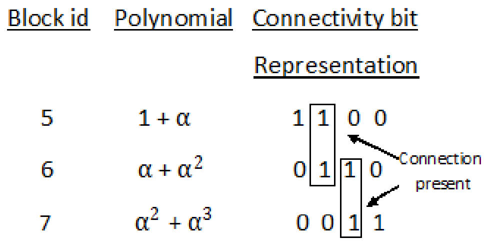
Baring a few minor changes I shall use the BCH code to develop connectivity key. Each sensor node is given a unique connectivity polynomial. Sensor nodes will match the bit positions of the polynomials first. If in both polynomial same bit position matches then are connected. Depending on maximum connection possible with a sensor node take the value of m. Based on the primitive polynomial construct the codeword of $GF(2^m)$. Let α be the primitive element in $GF(2^m)$. One can choose any other primitive polynomial for generating codeword. Each sensor node is given a unique connectivity polynomial. Sensor nodes will match the bit polynomials first. If in both polynomial same bit position matches then are connected. Here I have used 0,s and 1's for better understanding. If there are all 1's in the same position then the nodes are connected.

Example 1.
Let $m = 4$ then $n = 15$ from $n = 2^m - 1$. α be the primitive element of $GF(2^4)$. Here we took primitive polynomial $1+\alpha+\alpha^4 = 0$. Using the primitive polynomial we compute the power of α. For example $\alpha^6 = \alpha^2\alpha^4 = \alpha^2(1+\alpha) = \alpha^2+\alpha^3$. Similarly the connectivity polynomials are shown in Table 1.

Each sensor nodes are given this polynomials. If at least one bit between the polynomials of the sensor nodes matches(i.e. 1 present in the same bit position) that means the sensor nodes are connected(i.e. they are in the same communication region). Figure 2 shows the connection representation of sensor nodes. In this example node 5 is connected with node 6 and node 6 is connected with node 7. Node 5 and node 6 can communicate with each other and node 6 and node 7 can communicate with each other if they have at least one communication key common.

A. Incretion of new node into WSN

Incretion of new node is done by increasing the value of m depending on number of nodes to be inserted. This new nodes can create another sub network or can be inserted into an existing sub network without changing the existing keys.

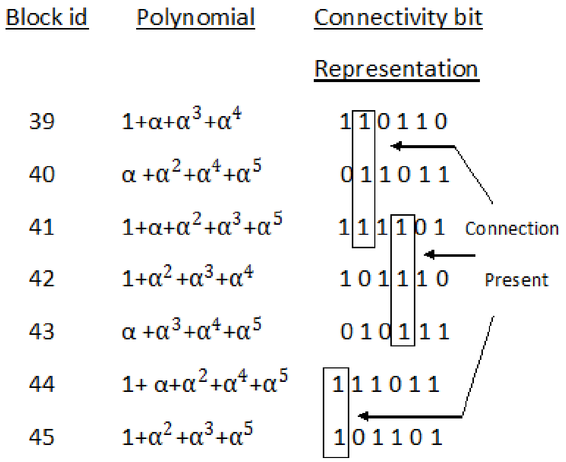


Block id	Polynomial	Connectivity bit Representation
5	1 + α	1 1 0 0
6	α + α^2	0 1 1 0
7	α^2 + α^3	0 0 1 1

Fig. 2. Connection of nodes

Suppose in the existing network if the value of m is kept u_1. Then took any value of $m > u_1$ depending on number of nodes to be inserted. It is preferable to choose small differences value of m from the existing value of m. This makes our scheme scalable.

Using previous example I showed this fact. If some new nodes need to inserted into the network which is shown in the previous example then just take the value of $m > 4$ and generate new connectivity polynomial for the new sub network or inert the nodes into the existing network. Let we take m=6. Let α be the primitive element of $GF(2^6)$. Here primitive polynomial is $1+\alpha+\alpha^6 = 0$. Node 39 to 45(Node id) create new sub network. This sub network is added to the existing network. Figure 3 shows how the new sub network is created using BCH code and how the new nodes are connected with each other and Figure 4 shows how new nodes can be inserted into the existing network.



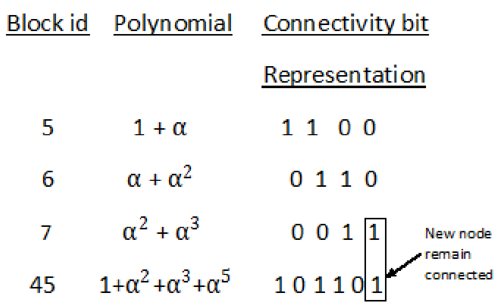
Block id	Polynomial	Connectivity bit Representation
39	1+α+α^3+α^4	1 1 0 1 1 0
40	α +α^2+α^4+α^5	0 1 1 0 1 1
41	1+α+α^2+α^3+α^5	1 1 1 1 0 1
42	1+α^2+α^3+α^4	1 0 1 1 1 0
43	α +α^3+α^4+α^5	0 1 0 1 1 1
44	1+ α+α^2+α^4+α^5	1 1 1 0 1 1
45	1+α^2+α^3+α^5	1 0 1 1 0 1

Fig. 3. Creation of new sub network

B. Mobility of Sensor Nodes

Mobility is a basic characteristic of sensor nodes. Sensor node can be moved to one sensor nodes communication region to another sensor nodes communication region. When a sensor node is moved to one sensor nodes communication region to another sensor nodes communication region it then must be remain connected with new region otherwise the sensor node will be unreachable.

In this scheme BCH code is used for connectivity key. That is why if a sensor node is moved to one sub network to another



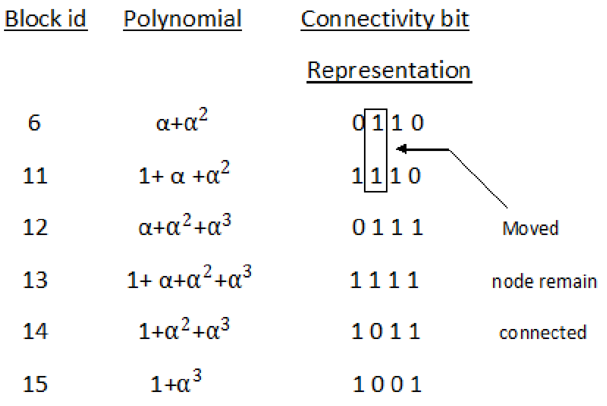
Block id	Polynomial	Connectivity bit Representation
5	$1 + \alpha$	1 1 0 0
6	$\alpha + \alpha^2$	0 1 1 0
7	$\alpha^2 + \alpha^3$	0 0 1 1
45	$1+\alpha^2+\alpha^3+\alpha^5$	1 0 1 1 0 1

New node
remain
connected

Fig. 4. Insertion of new node to existing network

sub network or from any place of the network to other place of the network then there remain a probability of being connected with new sub network or with the existing network. That is why sensor nodes cannot be disconnected. If one has to move a sensor node from a sub network to another sub network or any place of the network then there is no need of redistributing the keys to the sensor node. Sensor nodes remain connected when sensor nodes is moved to any location of the network.

With continuation from previous example I have illustrated the mobility of sensor nodes. Let there are two sub network $subnet_1$ and $subnet_2$. $Subnet_1$ consists of nodes 5, 6 and 7(Block id) and $subnet_2$ consists of nodes 11 ,12 ,13 , 14 and 15(Block id). If node 6 is moved into $subnet_2$ from $subnet_1$. Then the node 6 will remain connected with $subnet_2$. There is no need of redistributing the key to the node 6. Because 2nd and 3rd bit position of node 6 matches with node 11's bit in same position. So node 6 remain connected with $subnet_2$. So there is a probability of remain connected with the new sub network. Figure 5 shows how moved node 6 remain connected with the network.



Block id	Polynomial	Connectivity bit Representation
6	$\alpha+\alpha^2$	0 1 1 0
11	$1+ \alpha +\alpha^2$	1 1 1 0
12	$\alpha+\alpha^2+\alpha^3$	0 1 1 1
13	$1+ \alpha+\alpha^2+\alpha^3$	1 1 1 1
14	$1+\alpha^2+\alpha^3$	1 0 1 1
15	$1+\alpha^3$	1 0 0 1

Moved
node remain
connected

Fig. 5. Moved node remain connected

IV. Communication probability of the network

Communication probability or Connectivity(p_c) defines the probability that there is a connection between the nodes. If the nodes have at least one same bit position then we say that a link is present between two nodes. Mathematically, we have

$$p_c = \frac{\text{Number of connection present in the network}}{\text{Total Number of possible connection}}$$

Here communication not only dependents on connection. Communication depends on connection as well as on communication key. So if two nodes are in their communication region and if they have same connection key and communication key then direct communication is possible. Otherwise multi hop communication is done. In this case there exist one or more nodes which connect these two nodes. Connection probability may vary from one network to another network because sensor nodes are moveable.

V. Scalability of the network

This model is scalable because any number of node can be entered into the network without changing keys in the already existing nodes. Same Key means keys are drawn from same key pool and key-chain length is same.One can easily create a sub network using new nodes without changing the existing keys or insert new node into the existing network in any place. By increasing the value of m, depending on number of new node to be added to the existing network on can easily create a new sub network without changing the existing keys. Suppose there are $n = (2^m - 1)$ number of nodes nodes in the network. If we increase the value of m by 1, then there can be $2^{m+1} - 1$ number of new connectivity polynomial. So we can introduce a maximum of $(2^{m+1} - 1) - (2^m - 1) = 2^m$ number of new nodes without redistribution of keys. One can increase the value of m more than 1 for introducing more number of nodes. Here key pool remain unchanged and key-chain length remain un changed for the existing nodes.

For example let $m = 4$. Then number of node will be 15. If we increase the value of m by 1. Then number of new node can be introduced is $2^4 = 16$.

VI. Resiliency

Resiliency refers to the sustainability of the sensor network when one or more of its nodes have compromised by the attacker. A selective node capture attack or random node capture attack can occur in WSN. In both type of attack if a node became compromised then all the connectivity and communication key will be exposed. The links will break where these keys are used. But when two nodes want to communicate with other they broadcast their node ids not the key identifier. So attacker cannot able to understand which nodes are being remained compromised. So selective node capture attack is not possible with this scheme. There are two kinds of resiliency E(s) and V(s).

$$E(s) = \frac{\text{Number of communication links broken}}{\text{Total number of communication links}}$$

$$V(s) = \frac{\text{Number of captured node}}{\text{Total number node}}$$

Number of communication link is depends on both communication and connectivity key. One more key is used for communication of sensor nodes. That is why the attacker will not able to understand in which way the nodes are connected. And resiliency depending on key management scheme for connectivity and communication key. It is preferable to use keys with high distance for more security. So it is obvious in this scheme the desired resiliency is maintained with more secure communication.

VII. CONCLUSION AND FUTURE WORK

Here a novel technique is proposed for making communication more secure between sensor nodes using codes. In this scheme different keys are used for communication and connection. The connectivity key is generated using BCH code. The important advantages of this scheme are moved nodes remain connected with the network and new node can be added to the network without redistributing the keys in the existing sensor nodes. So moved sensor nodes will not be disconnected and network will remain scalable. In our scheme each node is responsible for its own communication. We use different connectivity key for more secure communication. As two different keys are being used in this scheme, the communication of sensor nodes became more secure.

However there is a scope for further developments on this area. One can use other code for connectivity key so that more secure communication can be done and achieve better resiliency. It will be a nice work if one can increase the connectivity maintaining secure communication. One can use any other code for communication for better result. Further development can be done on mobility of sensor nodes maintaining desired scalability and resiliency.

REFERENCES

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

[2] L. Eschenauer, V. D. Gligor, A key-management scheme for distributed sensor networks In: Atluri, V. (ed.) ACM Conference on Computer and Communications Security, pp. 41-47. ACM, New York, 2002.

[3] M. Al-Shurman and S. M. Yoo, Key pre-distribution using mds codes in mobile ad hoc networks, In: ITNG, pp. 566-567. IEEE Computer Society Press, Los Alamitos, 2006.

[4] Seyit Ahmet Camtepe and Bulent Yener. Combinatorial design of key distribution mechanisms for wireless sensor networks. In Pierangela Samarati, Peter Y. A. Ryan, Dieter Gollmann, and Refik Molva, editors, ESORICS, volume 3193 of Lecture Notes in Computer Science, pages 293308. Springer, 2004.

[5] S. A. Camtepe, 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.

[6] Anupam Pattanayak, B. Majhi Key Predistribution Schemes in Distributed Wireless Sensor Network using Combinatorial Designs Revisited National conference on modern trends of Operating Systems MTOS -2009 Bhubanaswaer,2009 pp 43-50.

[7] Pinaki Sarkar, Amrita Saha and Morshed Udan Chowdhury, Secure Connectivity model inWireless Sensor Networks (WSN) using 1st order Reed- Muller codes, 2010.

[8] San Ling, Chaoping Xing, Coding Theory A First Course, Cambridge University Press, 2004.

[9] Madhu Sudan, Algorithmic introduction to coding theory lecture on BCH code September 19, 2001.

[10] Dr.Thamer, Information Theory 4th Class in Communications, BCH Code. http://www.uotechnology.edu.iq/dep-eee/lectures/4th/Communication/Information20theory/2.pdf.

[11] Ruj. S, Application of Combinatorial Structures in Key Predistribution to Sensor Networks using Combinatorial Designs, Ph.D. thesis, Indian Statistical Institute, 2009.

[12] B. Cooke, Reed Muller Error Correcting Codes, MIT Undergraduate Journal of Mathematics, 1999.