ACSIJ
WWW.ACSIJ.ORG

Key management in heterogeneous wireless sensor networks using Voronoi diagrams

Ronak Tahmasbi[1], H.Haj seyed javadi[2], M.E.Shiri[3], Ahmad Allahyari[4]

[1] Software Eng. Department of computer, Kish International Branch, Islamic Azad University, Kish Island, Iran
[2] Department of Mathematics and Computer Science, Shahed University, Tehran, Iran
[3]Industrial Eng. Faculty of Amirkabir University of technology
[4] Industrial Eng. Faculty of Industrial Engineering, Alghadir University, Iran

Abstract

Wireless sensor networks' nodes are divided to H-sensors and L-sensors. H-sensor nodes are more powerful than L-sensor nodes in term of processing and memory. Because of that H-sensors are considered as the cluster head and L-sensor as the cluster member. H-sensor is responsible for the security of this communication. Each node can communicate with neighbor nodes. Problems of the network are security and battery lifetime for each node. Proposed algorithm presents a scheme to keep the security, reduce energy consuming and the length of message in wireless sensor network. In this scheme each node select as a cluster head based on minimum distance of neighbor nodes. This algorithm, uses Voronoi algorithm, did the most optimal clustering and divides the operational environment to Voronoi spaces and allocate a key to each Voronoi space for secure connection with neighbor spaces. Each Voronoi spaces have special key, so that energy consuming decreases and security increases. This scheme is evaluated by MATLAB, simulation software, and compared with previous algorithms. Results of this simulation show that this scheme operates better than similar schemes because it decreases the length of message and energy consuming.

Keywords: *key management, wireless sensor network, Voronoi diagram*

1. Introduction

Wireless sensor network consist of some H-sensors and L-sensors that H-sensor apply as the cluster head because of their power in processing and memory and L-sensors are node's cluster members. Communication between nodes must be secure so H-sensors are reasonable for authentication and security. The number of H-sensor are not too much But they are more powerful that L-sensors so using H-sensors in key management and reduce money and consume memory. First H-sensors pre-distribute with keys and H-sensors pre-load L-sensor with key similar BS works in front of H-sensors. In this scheme key pre-distributed scheme based on random key pre-distributed for Heterogonous sensor network Proposed with Voronoi algorithm. This scheme based on [2] using key management.

Note is that the keys pre-loaded in nodes with cluster head and is not need to preload with BS also derived keys.

For comparison of the proposed scheme with scheme presented in [1], some factors are compared and studied like: communication captured node and consumed energy. Communication is assumed to be secured.

Communication between two L-sensors in one cluster or two different clusters is possible. Voronoi spaces will be explained in section C, in node capturing section will study communication networks that one of their keys is discovered. The only defect is memory consuming and it leads to increscent of security and this consuming is not too much so it can be ignored. In the other hand security and energy consuming in proposed scheme had been optimized.

This article is organized as below: Section 2 explains related works, section 3 and 4 is about network model and proposed scheme. Section 5 presents results and analyzes the efficiency of the

proposed scheme and finally section 6 will state the conclusion of this study.

2. Related works

Banihashemian and Ghaemi Bafghi [1] proposed an efficient key management in wireless sensor networks. Resiliency and connectivity are two important factors in proposed scheme. This scheme contains four stages. These stages are key pre-distribution and localization, seeds assignment, deriving new keys and shared discovered keys.

Du et al. [2], based on symmetric pre- distribution key management and proposed scheme called AP. Main idea in AP is asymmetric pre-distribution key management, preload many keys in little number of H-sensors, H-sensors are powerful and they are not too much so, keys are stored in L-sensors: L-sensors have little range of communication storage and capacity.

Chan and Perring [3] improves PIKE protocol based on key establishment by using peer sensor nodes as trusted communication.

In recent studies on security improvement of key pre-distribution, some schemes are proposed in [4, 5] and [6] have studied on threshold key pre-distribution scheme.

3. Network model

Base station (BS) is assumed to be secures and resources such as energy process power and memory are not limited.

H-sensors are more powerful in terms of memory and processing than L-sensors. H-sensors are connected to BS directly.

A. Assumptions

- Assume that H-sensors and L-sensors are distributed randomly in operational environment.
- H-sensors are clusters head and L-sensors are as the cluster members.
- Suppose that networks are secured in distribution phase and only capture node along communication.
- Location of L-sensors and H-sensors are static.
- Range transfer of H-sensors and L-sensors are static.
- Range conduction of H-sensors are high, therefore L-sensors can receive message "Hello" from one or more H-sensors.
- Number of sensor nodes in a cluster is assumed to be not determined.
- Each H-sensor have GPS and report locations.

B. Notations

Using the following notations to describe proposed key management protocol and involved cryptographic operations in this paper.

BS: base station

adv_i: advertisement message by i^{th} cluster head

CH_i: cluster head of i^{th} cluster

K_{BS-i}: pair wise key between BS and node with ID_i

$Seed_{i,j}$: seed related to i^{th} cluster and i^{th} Voronoi space

S: total number of seeds used in entire network

S_b: minimum of seeds needed by protocol

S_d: additional seeds that need after cluster formation

$E_K(M)$: encrypt message M by key K

$D_K(M)$: decrypt message M by key K

Hash(K, seed): hash key K with seed

Dist: distance between Voronoi space neighbors

BK_i: i^{th} base key

DK_{i-j}: i^{th} key hashed by seed j

K_N: shared key used by all nodes in the network

K_{Cm}: cluster key used by all nodes in cluster m

K_{BS-Chi}: pair wise key between Chi and BS

4. Proposed protocol

Proposed scheme is base on [2]. Using the concept of Voronoi space that the main idea is using cluster information based on distance between node and its cluster head in key management. Node in each Voronoi space is selected based on its distance.

Distance is computed base on RSSI [3]. A unique seed is allocated to each Voronoi space that is used to create derived keys. Derived keys are used for secure communication with neighbor Voronoi spaces. IN general Networks are divided to different Voronoi spaces with different keys.

The amount of H-sensors is not too much and number of seeds is enough, so seeds meet key management requirements, therefore a little amount of seeds will belonged to H-sensors (S_b). The number of base keys is equal to division of key pool size on S_b.

A. Pre-distribution phase

In the first stage, a key pool is generated. In this phase base keys are applied, but derived keys are not used. Each node stores one base key of *k* base keys randomly. And each H-sensor stores one base key (*c*), in which $c>>k$.

Pair wise key is used between BS and $K_{BS\text{-}Chi}$. Each L-sensor stores one base key and one key has been stored between nodes and BS that is used for authentication by BS.

Key cluster is generated by using K_N and each node stores one K_N. The node could be L-sensor or H-sensor

B. Computing number of seeds needed for each cluster

Because there is not enough information about nodes location, number of seeds can not be estimated. Minimum number of seeds is equal to number of clusters because, each cluster has one key. Each H-sensor is the cluster head and each cluster head sends its location in operational environment of network grid of 400m*400m in Voronoi spaces. Number of Voronoi spaces is equal to cluster heads.

Each cell in grid has coordinate X_{cell} and Y_{cell}, each node based on minimum distance to cluster head specifies that each cell belongs to which cluster. Cluster head is selected randomly and Voronoi space is formed based on minimum distance to cluster head. Now each cluster head reports its distance from BS and BS sends seed to each cluster head. For example if 5 Voronoi spaces after dividing to 5 BSs, each BS sends seed for all Voronoi spaces.

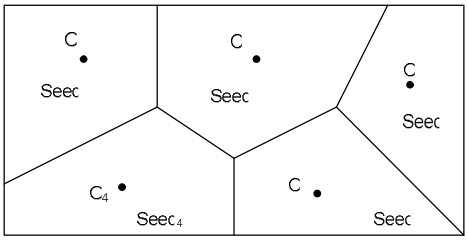

Figure 1. Example of 5 Voronoi spaces and 5 BSs

C. Computing new keys by seed

Some seeds are sent to cluster's nodes by their cluster head. Nodes generate new keys by using distance. Then BS sends seed to cluster head. Each cluster head transmits a Hello message, computes distance by RSSI to makes a seed join to another cluster head that have minimum distance to that node. Cluster key is computed by K_{Cm}= hash (K_N || ID_{CHm}).

BS ➔ CH_i : EBS ([$seed_{i,1}$, $dist_1$] , [$seed_{i,2}$, $dist_2$] , …, [$seed_{i,z}$, $dist_z$])

Cluster head receives seed from BS and computes new key and sends it to node's cluster and this message encrypt by keys of that cluster.

CH_i ➔ Node: K_{Ci} ([$seed_{i,1}$, $dist_1$] , [$seed_{i,2}$, $dist_2$] , …, [$seed_{i,z}$, $dist_z$])

Each Voronoi space generates new key for communication with neighbors Voronoi space by seeds:

$Kj_1^{(1)}$ = hash (kj_1 , $seed_{i,z}$)

$Kj_1^{(2)}$ = hash (kj_1 , $seed_{i,z+1}$)

.

.

$Kj_1^{(n)}$ = hash (kj_1 , $seed_{i,z+n}$)

For security preservation, after first communication seeds will be deleted.

D. Shared key discovery

Each node transmits a message that encrypt by key cluster, this message contains its keys and its Voronoi space. In result, neighbor's nodes find shared key.

If multiple shared keys exist, one of them is selected randomly, but if the shared key does not

exist, each node sends request message that contains its ID, its Voronoi space, its list of keys and other node's ID for cluster head. Cluster head have more chance to have a shared key because the number of key cluster heads is more. Cluster head selects one of its keys and sends it to nodes, but if cluster head did not have a shared key, nodes send message to BS and BS sends a shared key to them.

5. Performance evaluation

In this section, performance of proposed key management scheme is evaluated. Comparison between two scheme had been done proposed scheme and presented scheme in [1]. This comparison was done based on 4 elements, probability of secure connectivity (p), resilience of proposed key management scheme (res), energy consuming (k eng) and message length (ml).

A. Secure connectivity

Probability of secure connectivity between two L-sensors is calculated for l0 sensors and in three different states.

State 1; number of sensor nodes is 1000, radius is 50, number of captured nodes is 50 and all of them are static and similar for two schemes. Number of cluster is variable. Table 1 shows results for state 1.

Table 1. Comparison of secure connectivity in proposed scheme and scheme in [1] (State 1)

		Original			
run	Input	p	res	k eng	ml
1	10	0.04004	0.04009	0.01943	9.22658
2	12	0.03659	0.03656	0.01650	10.29395
3	14	0.03180	0.03194	0.01522	11.61720
4	16	0.02899	0.02911	0.01211	12.34847
5	18	0.02660	0.02680	0.01066	13.09144
6	20	0.02336	0.02310	0.00987	14.61928
7	22	0.02149	0.02115	0.00919	15.70155
8	24	0.02005	0.01969	0.00831	16.62267
9	26	0.01910	0.01849	0.00741	17.64261
10	28	0.01759	0.01644	0.00750	18.98006

		Proposed			
run	Input	p	res	k eng	ml
1	10	0.04341	0.04345	0.00294	2.89800
2	12	0.04345	0.04350	0.00235	3.36500
3	14	0.04325	0.04330	0.00112	3.32900
4	16	0.04357	0.04362	0.00163	3.29100
5	18	0.04383	0.04387	0.00163	3.29100
6	20	0.04491	0.04495	0.00052	3.59700
7	22	0.04295	0.04300	0.00071	3.82700
8	24	0.04358	0.04363	0.00034	3.80600
9	26	0.04407	0.04412	0.00032	3.82600

Figure 1 shows charts about results from Table 1 in different number of clusters.

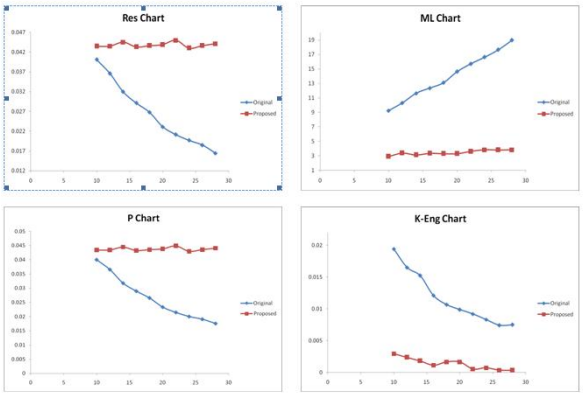

Figure 2. Charts about results from Table 1

State 2; number of sensor nodes is 1000, number of clusters is 10, number of captured nodes is 50 and all of them are static, results of tests variant radius is presented in Table 2.

Table 2. Comparison of secure connectivity in proposed scheme and scheme in [1] (State 2)

		Original			
run	Input	p	res	k eng	ml
1	50	0.03974	0.03980	0.01994	9.35810
2	60	0.04989	0.05004	0.01876	8.75780
3	70	0.06024	0.06065	0.01903	8.29519
4	100	0.08948	0.09127	0.01886	7.57775
5	200	0.18399	0.19316	0.02006	6.82113
6	300	0.27147	0.29106	0.01947	6.44641
7	400	0.36193	0.38901	0.01986	6.35146
8	500	0.36476	0.39110	0.01941	6.22636
9	600	0.35128	0.37824	0.01868	6.13834
10	700	0.36404	0.39152	0.02000	6.19354

		Proposed			
run	Input	p	res	k eng	ml
1	50	0.04400	0.04404	0.00254	3.04900
2	60	0.06294	0.06300	0.00256	3.13800
3	70	0.08237	0.08246	0.00228	3.01300
4	100	0.15327	0.15343	0.00259	2.94100
5	200	0.49987	0.50037	0.00341	2.77500
6	300	0.78441	0.78520	0.00240	2.92600
7	400	0.97181	0.97278	0.00305	2.93200
8	500	0.99829	0.99929	0.00458	2.89300
9	600	0.99900	1.00000	0.00314	2.78800

Figure 3 shows charts about results from Table 2 in different radiuses.

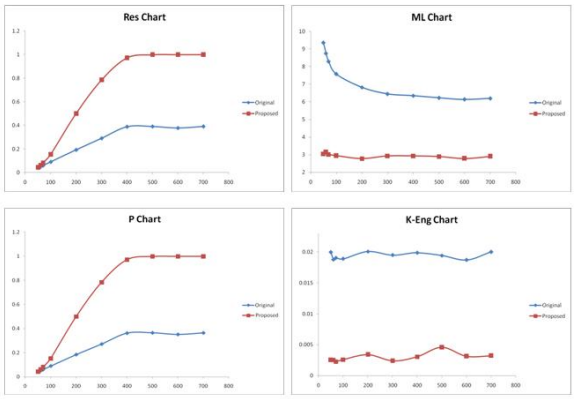

Figure 3. Charts about results from Table 2

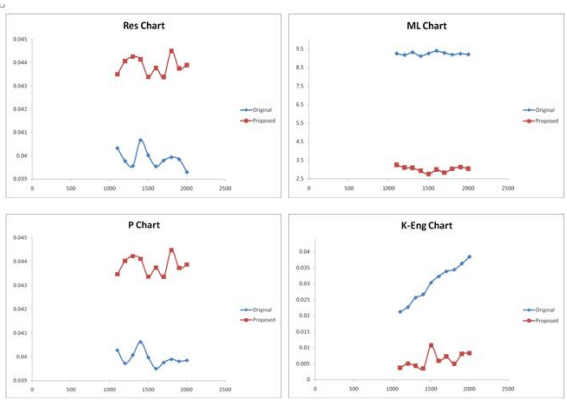

Figure 4. Charts about results from Table 3

State 3 is built for two schemes with flowing factors; number of clusters is 10, number of captured nodes is 50 and radius is 50, the variant factor is number of sensor nodes and results is showed in Table 3. Figure 4 shows charts about results from Table 3 in different number of sensor nodes.

Table 3. Comparison of secure connectivity in proposed scheme and scheme in [1] (State 3)

		Original			
run	Input	p	res	k eng	ml
1	1100	0.04027	0.04033	0.02123	9.25462
2	1200	0.03972	0.03977	0.02270	9.17231
3	1300	0.04008	0.03956	0.02565	9.31462
4	1400	0.04062	0.04067	0.02666	9.11005
5	1500	0.03998	0.04002	0.03033	9.26051
6	1600	0.03950	0.03954	0.03229	9.39509
7	1700	0.03976	0.03980	0.03388	9.28175
8	1800	0.03989	0.03993	0.03437	9.19492
9	1900	0.03981	0.03985	0.03628	9.24743
10	2000	0.03985	0.03929	0.03841	9.20701

		Proposed			
run	Input	p	res	k eng	ml
1	1100	0.04346	0.04350	0.00363	3.23636
2	1200	0.04402	0.04406	0.00502	3.09833
3	1300	0.04422	0.04426	0.00426	3.08154
4	1400	0.04411	0.04414	0.00354	2.92071
5	1500	0.04336	0.04339	0.01064	2.74733
6	1600	0.04374	0.04376	0.00591	2.97875
7	1700	0.04335	0.04338	0.00720	2.83235
8	1800	0.04448	0.04450	0.00490	3.03611
9	1900	0.04373	0.04375	0.00800	3.13316
10	2000	0.04387	0.04389	0.00830	3.03400

B. Security analysis

In this section, resilience of proposed key management scheme against compromise attack mode will be checked. L-sensor t attacks to the network, each L-sensor maintains k preloaded keys and nk derived keys. Similar to study [1], probability of captured nodes by L-sensor t is calculated by:

$$P_{reveal} = 1 - \left(1 - 2k \Big/ (b \times s)\right)^t$$

Proposed scheme was evaluated by simulation and was compared with [1]. Results that are showed in Table 4 are about security analysis when number of sensors is 1000, number of cluster is 10 and radius is 50. Charts about results in Table 4 are presented in Figure 5.

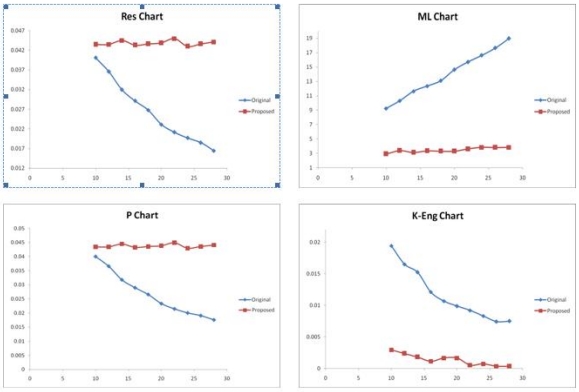

Figure 5. Charts about results from Table 4

Table 4. Comparison of security analysis in proposed scheme and scheme in [1] (State 3)

		Original			
run	Input	p	res	k eng	ml
1	20	0.17867	0.18782	0.00124	6.80858
2	35	0.18301	0.19240	0.00129	6.85880
3	75	0.17378	0.18215	0.00134	6.96039
4	125	0.18576	0.19498	0.00127	6.78461
5	250	0.17758	0.18646	0.00129	6.93642
6	330	0.18391	0.19336	0.00120	6.78118
7	450	0.17893	0.18785	0.00127	6.78461
8	540	0.18425	0.19453	0.00113	6.60541
9	600	0.17745	0.18657	0.00123	6.80744
10	750	0.18152	0.19108	0.00115	6.77319
		Proposed			
run	Input	p	res	k eng	ml
1	20	0.04469	0.04474	0.00256	3.09300
2	35	0.04318	0.04323	0.00437	3.12000
3	75	0.04391	0.04395	0.00400	3.20400
4	125	0.04450	0.04454	0.00384	3.30800
5	250	0.04365	0.04369	0.00294	3.02300
6	330	0.04419	0.04423	0.00253	3.28400
7	450	0.04362	0.04366	0.00295	2.94800
8	540	0.04347	0.04351	0.00429	3.18000
9	600	0.04412	0.04416	0.00235	3.39300
10	750	0.04350	0.04354	0.00234	2.90800

Results with variety of clusters, sensor nodes and radiuses show, that probability *p* in three studies is improved, energy consuming is decreased to 1/3 and message length decrease to 1/4 of initial amounts. Probability of nodes capturing is increased, but this increasing is little in comparison with probability *p* and decreasing energy consuming and message length.

6. Conclusions

In the proposed scheme is a new key management based on pre-distribution randomly keys using Voronoi diagram.

A lot of L-sensors and a few H-sensors used, similar to [1], these H-sensors are cluster heads. In the proposed scheme L-sensors and H-sensors base on keys are pre-distributed and for each L-sensor a seed is assigned based on distance of its cluster heads and the cluster. This study compares the proposed scheme with presented scheme in [1] that probability of shared key (*p*), energy consuming and message length are improved and results show that proposed scheme is more effective and defects are very little.

References

[1] Banihashemian, Saber and Ghaemi Bafghi, Abbas. A new key management scheme in heterogeneous wireless sensor networks. Mashhad: Ferdowsi University of Mashhad (FUM), 2011.

[2] An effective key management scheme for heterogeous sensor networks, Ad Hoc Networks. Du, X., et al. 2007, Vols. 24-34.

[3] peer intermediaries for key establishment in sensor networks. H, H. Chan and Perring, Pike, A. s.l. : 24th annual joint conference of the IEEE computer and computer and communications societies, 2005. INFOCOM 2005.

[4] Location-based pairwise key establishments for static sensor networks. Liu D, D. and Ning, P. New York : proceedings of the lst ACM workshop on security of adhoc and sensor networks, 2003, Vols. 72-82.

[5] Securing sensor networks with location based keys. Zhang, Y., et al. s.l. : Wireless Communications and Networking Conference, 2005.

[6] Non-public key distribution. Blom R, R. s.l. : Advances in cryptologyCRYPTO82, 1982, Vols. 231-236.

[7] Z, Liu, et al. A distributed energy-efficient clustering algorithm with improved coverage in wireless sensor networks. J. Future Generation Computer Systems, s.l. : 2011.

[8] An initialization method for the K-means algorithm using neighborhood model. C, Fuyuan, J, Liang and G, Jiang. s.l. : J Computers and Mathematics with Applications, 2009, Vols. 474-483.

[9] Hierarchical initialization approach for K means clustering. JF, Lu, et al. s.l. : J Pattern Recognition Letters, 2008, Vols. 787-795.