Wireless Sensor Network, **2011, 3, 183-188**
doi:10.4236/wsn.2011.36021 Published Online June 2011 (http://www.SciRP.org/journal/wsn)

A New Kind of Dynamic Key Protocol for Wireless Sensor Network

Cai-Xia Zhang[1,2], Liang-Lun Cheng[1], Xiang-Dong Wang[2]
[1]*The Faculty of Automation, Guangdong University of Technology, Guangzhou, China*
[2]*Foshan University, Foshan, China*
E-mail: *zh_caixia*@163.*com*

Received May 1, 2011; *revised June* 2, 2011; *accepted June* 9, 2011

Abstract

For the source limitations and vulnerabilities of the sensor nodes of Wireless Sensor Networks, we propose the new kind of dynamic key protocol for wireless sensor network, using the unidirection of hash function and the thinking of Hill to study the dynamic key matrix. Through theoretical analysis of some aspects, our method can promote security, connectivity expansibility; the results show that this protocol reduces storage space and communication energy consumption also.

Keywords: Wireless Sensor Network, Key Matrix, Key Protocol, Connectivity

1. Introduction

Wireless sensor networks have the characteristics of self-organizing, flexibility, small size and low cost, so it is widely used in military, environmental monitoring and forecasting, urban transport and other fields. Since wireless sensor networks are generally deployed in the unattended wilderness area and nodes can be easily captured to leak sensitive classified information, ensuring the security of wireless sensor networks is very important. Typical sensor nodes suffering the constraints of communications, computing, storage, energy and other performance, can not be deployed sophisticated security mechanism; and its low-cost properties makes some expensive safety feature implemented. Therefore, low power, low storage capacity, and high-security type of security management mechanism for the normal operation of wireless sensor networks is very important.

Key management is the key issue in the security of wireless sensor network. The current key management scheme is divided into symmetric key and asymmetric key management system. Since the complexity of algorithm, the asymmetric key can not be directly applied to the resource-constrained wireless sensor networks.

Symmetric key management scheme can generally be divided into the following three categories: methods based on key distribution centers, pre-distribution and packet-based clustering approach, the three methods will be introduced as following. Literature [1], the Kerberos protocol is the first agreement based on one of the key distribution centre. Its basic idea is that the session key for communication used by sensor is from the key distribution center which is responsible for generating key. The key distribution centers and each sensor node in the networks share a unique key and store all the share keys. The advantage of such agreements is easy to achieve, moreover, both the computational overhead and storage requirements of sensor nodes are low, but the network service relies on the key center excessively. If the center is destroyed, the high network Jibei is broken, while the network scalability is poor. Literature [2] first proposes the concept of random key pre-distribution. In this mechanism, the node first will randomly select a certain number of key from the key pool to pre-loading and after being deployed, the nodes will build the links according to the probability of having the same key. Nodes must be preloaded sufficient number of key to guarantee the connectivity probability, and spend a lot of communication overhead on the same node authentication key. For storage, the energy limited nodes; it will be a great burden. Literature [3-7] makes a more detailed analysis on this basis and the mechanism extends. Literature [8] proposes grid-based and Literature [9] proposes a com

- Special funds for technology of Guangdong Province: (2009A080207008) funded.
- Special funds for technology development project of Foshan City: (FZ2010030) funded.
- Special funds for technology development project of Foshan City: (2009033) funded.

bination of methods which is the further study for the random key pre-distribution mechanism.

Literature [10,11] proposes a key management program which combines network topology. It is a key management program to expand and improve random key pre-distribution and it is based on the cluster group key management scheme. The network is better to calculate and expand, to support large-scale networks and network dynamic change, but hasn't got a good solution to distributing the nodes in the cluster after deleting cluster head.

In summary, many scholars have done a lot of work in order to better solve the limitations on security, connectivity, scalability, node storage of the wireless sensor network in the key management scheme. Based on the analysis of the previous, this paper proposes a key management based on dynamic key matrix for wireless sensor networks to reduce the amount of key storage nodes and enhance connectivity and resistance to attack of the net. Section 1 and Section 2 present the background, and they give the mathematical models and scenario assumptions that adopted by the scheme. Section 3 describes the detailed process to establish key; the results of theoretical analysis are in Section 4; Section 5 is the conclusions of this paper.

2. Prior Knowledge and Application of Assumptions

2.1. Hill Cipher

The realization of this agreement needs to learn some basic password Hill's ideas and methods, then the following is a brief description of it. In the process of transferring information, through the matrix operation, the sender converts the plaintext into the ciphertext and the receiver through the inverse matrix converts ciphertext into plaintext, then the secure transmission of information completes. Details process as follows:

(1) Divide plaintext that needs to be transferred into the same size as several explicit groups, each character of the group encrypted another character of the group;

(2) Every time encrypted one group, if we express each M characters in the group known as the p_1, $p_2, \cdots, p_m$, the corresponding characters in the ciphertext will be known as $c_1, c_2, \cdots, c_m$, encryption algorithm:

$$c_1 = p_1 k_{11} + p_2 k_{21} + \cdots + p_m k_{m1}$$
$$c_2 = p_1 k_{12} + p_2 k_{22} + \cdots + p_m k_{m2}$$
$$\ldots$$
$$c_m = p_1 k_{1m} + p_2 k_{2m} + \cdots + p_m k_{mm}$$

That $\boldsymbol{C} = \boldsymbol{PK}$, the key $\boldsymbol{K}$ is a reversible square $m \times m$ (m indicates the groups' size), if encrypt the plaintext P by linear transformation, the receiver can decrypt to get the plaintext information through $\boldsymbol{P} = \boldsymbol{CK}^{-1}$. To ensure that recipients can express exactly, it requires key matrix $\boldsymbol{K}$ to be reversible.

2.2. Unidirectional One-Way Hash Function

In the realization of this Agreement, it needs to express the dynamic generation of key matrix information, while ensure that even if the plaintext information is captured, the key matrix can not be quickly derived under the information. This can be achieved by using the one-way hash function. One-way hash function maps an arbitrary length message M to a function H that has fixed length hash value h (set the length m) $h = H(M)$.The function has the following characteristics: when M is given, it is easy to calculate h; and when h is given, according to $H(M) = h$, it is hard to inverse M; Finding two random messages M' to make $H(M') = H(M)$ *is* difficult. The above characteristics are suitable for this agreement.

2.3. Application of Assumptions

In practical situations, WSN nodes are not at the risk of being hijacked all the time. When nodes are deployed in the initial stage, the node usually has the ability to resist the hijacking within a certain period of time, because it doesn't perceived by the malicious attacker or the attacker does not know the distribution of the regional node in advance, As the WSN begins to work, the target nodes expose gradually and the nodes are attacked, increasing the possibility of being hijacked. The network attacker can exist in various forms [12]. Key establishment protocol proposed in this paper is based on this assumption on the network deployment phrase. Nodes do not need to preset the initial pair of keys, only according to the address information of neighbor nodes to establish the initial key matrix and the link connection.

There are two common modes of communication in Wireless sensor networks, which are communication between adjacent nodes and communication between nodes and base stations. So to establish link key and base stations can meet the needs of most applications and Link key can be used as the basis for the establishment of other types of keys. This paper focuses on the link's establishment process under the single-hop key. Neighbor sensor nodes which are usually in the range of radio communications can directly communicate with each other and establish a secure connection. The protocol proposed in this paper establishes link key in the case of reducing unnecessary verification and overhead of computing. Non-neighbor node needs to go through the middle Node multi-hop section to communicate.

3. Dynamic Key Agreement Based on Password Matrix

In this paper, the dynamic key agreement can be divided into three stages: the stage of presetting initial information, the stage of establishing link key, and the stage of updating link key.

3.1. Pre-Initial Information Stage

Prior to deployment in the network, a trusted base station will randomly preset a unique address and a 1 × *n* public key matrix $\boldsymbol{K}_0$ for each node. *The* $\boldsymbol{K}_0$ is used to generate the link key matrix, and *it* will be cleared after generating the link key matrix. Then the corresponding information will be loaded into each node. And just need to store its address *i* and public key matrix, which will greatly save storage space.

3.2. Link Key Establishing Stage

After presetting initial information, all nodes can be deployed to the target monitoring area. Because the node hasn't preset the key information, the deployment can be deployed randomly and not depending on the neighbor node. At that time, the sharing link key matrix between neighbors has not been established in the network, we should adopt the following methods to create the initial key matrix:

Each node *i* generates a random number $nonce_i$ which to be the characteristics of random numbers that required in the process of establishing the link key. Using the random numbers in the process of establishing the link key is to introduce spatial differences, which can limit the destruction that the initial key has been captured in the local area, that is, $i \to i \| nonce_i$.

Suppose the node *j* also randomly generates a random number $nonce_j$, after receiving the identity message of the node *i*, it can get the hash value *L* whose fixed length is n by the hash calculation.

$$\boldsymbol{L}_i = Hash\left(g\left(i \| nonce_i \| j \| nonce_j\right)\right)$$

Express the hash value *L* as a *n* × 1 column vector $\boldsymbol{L}$, then do the matrix operations on the vector $\boldsymbol{L}$ and the preset $\boldsymbol{K}_0$ of initial key. After that, the invertible *n* × *n* order matrix of the link key $\boldsymbol{K}_{ij}$ can be found,

$$\boldsymbol{K}_{ij} = \boldsymbol{K}_0 \times \boldsymbol{L}_i$$

Meanwhile, the node *i* also receives the identity message of node *j* and calculates the $\boldsymbol{L}_j$ by hash calculation, then do the operations on the results with the initial key matrix $\boldsymbol{K}_0$ .After that, the reversible link key matrix $\boldsymbol{K}_{ji}$ also has been found. To ensure that the *i* and *j* can produce the same link key $\boldsymbol{K}_{ij} = \boldsymbol{K}_{ji}$ in the circumstance that they only know the identify each other, the following conditions should been met

$$g(i,j) = g(j,i)$$

In order to ensure that different neighbor nodes do not generate the same link key, the function must also be met:

$$g(i,j) \neq g(i,k)$$

3.3. Link Key Update Phase

This Agreement used dynamic update key matrix. In this protocol during each session using a key matrix, the session ended, both sending and receiving updates immediately shared session key matrix.

First, the sending node will divide the plaintext P which is need to been sent into n equal segments $p_1, p_2, \cdots, p_n$ (each length is n and if the last segment is less than n, replace it with the space). Use the **Table 1** to change the segmented information into the corresponding digital information $\boldsymbol{D}_i = (d_1, d_2, \cdots, d_n)$, then multiply it with the link key matrix $\boldsymbol{K}_{ij}$ (*n* × *n* order) in the form of $\boldsymbol{C} = \boldsymbol{K}_{ij} \times \boldsymbol{D}_i^{\mathrm{T}}$ (T expressed to be the transpose of $\boldsymbol{D}_i$) to get the ciphertext message. After that, send the ciphertext to the receiving node and the receiving node will decrypt by the use of the inverse matrix of the link key matrix which is consulted and established by the nodes to get $\boldsymbol{D}_i^{\mathrm{T}} = \boldsymbol{K}_{ij}^{-1} \times \boldsymbol{C}$, then separate $\boldsymbol{D}_i^{\mathrm{T}}$ into the appropriate *n* section and check the corresponding **Table 1** to get the plaintext.

Meanwhile, the sending node and receiving node use the deferent plaintext information to update key matrix, the renewal process is as follows:

(1) In the same time of sending ciphertext information, the sending nodes utilize the plaintext information to update key matrix. The receiving nodes receive the ciphertext and convert it into plaintext, and in the meantime, the receiving nodes transform plaintext information into the new key matrix. That matrix is the same as the new key matrix of the sending node using the same approach as follows:

(2) The sending or receiving nodes divide the plaintext

Table 1. The integer code table of space and letter.

spaces	A	B	C	D	E	F
0	1	2	3	4	5	6
N	O	P	Q	R	S	T
14	15	16	17	18	19	20
G	H	I	J	K	L	M
7	8	9	10	11	12	13
U	V	W	X	Y	Z	
21	22	23	24	25	26	

information into n segments of unequal length randomly, $p_1, p_2, \cdots, p_m$:

(3) Make use of the Hash function which is loaded in advance to do the operation on each segments, and the n message values of which length is n can be generated;

(4) Use the **Table 1** to convert the n message values of which length is n into the *n* integer values of which length is n, so a new key $n \times n$ matrix can be got. If $\boldsymbol{K}$ is not reversible, it will re-do the above operation;

(5) Finally, both the sending and the receiving apply the new key matrix to replace the original matrix. From the above process, we can see that the process of information transmission and the communication overhead of updating key exist simultaneously;

(6) Repeat the above operation when sending information again. The key matrix of the link encryption and decryption should be updated instantly, truly to be "one-time close" and to enhance the resistance to capture of the node.

3.4. Adding New Nodes

After the network deployment, new nodes often need to be added because of the damage in the previous nodes or the requirement to expand the network. Verification is very necessary for how to prevent malicious nodes from masquerading as a new node to join the network and how the new nodes add to the network under the conditions of lacking real-time dynamic link key matrix. Because the pre-existing random key distribution mechanism has no way to make a distinction between the legitimate new nodes and the disguised new nodes that have possessed the captured key, it is vulnerable to be attacked by the node replication. This paper presents a new node authentication technology based on the density function hash. That authentication technology confirms whether the nodes that will join in are the new nodes or not by comparing the relationship between the plaintext information of the new nodes and the key matrix. The technology includes the following steps.

(1) New nodes initialize. Before deployment, the base station will be presorted stochastic plaintext information and the Hash function in the new node; and then spread the new node to a predetermined area;

(2) According to the mean in the stage of establishing link key, the new node will engender $n \times n$ key matrix by doing the Hash operation on the plaintext.

(3) The old node will do operation on the plaintext information and agree to establish links with the new node if the key matrix is the same as the one sent by the new n node; and then resend the plaintext to the new node randomly and build the link key matrix.

In the above process, even if the malicious nodes intercept plaintext message, it can not learn the new node's Hash function in a short time and therefore can not generate the key matrix, so the new node can add safely.

4. Theoretical Analysis

4.1. Safety Analysis

The physical capture and attacks of node in sensor networks is the most serious attack. In order to gain security information, the attacker can read or change the information in the node's memory. What the node saves in this agreement is the key matrix obtained after the Hash function, and due to the unidirection of the Hash function, it is impossible to export the original information from the key matrix in the calculation. The key in the agreement update with the information sending. Even if the node has been captured, there is new information communication between the adjacent node from the time of capturing to reading the security information, and the key has updated already. In this agreement, as the result of updating the key matrix dynamically, the malicious node can not obtain the dynamic information and has been isolated outside the network of the sensor.

4.2. Load Analysis

The electric power of communication consumption nodes is much more than the calculated consumption and thus it requires the communication load in key management protocol should be as small as possible.

The calculation load and communication load of this agreement will be analyzed. In this Agreement, when the order n of the dynamic key matrix is greater, the amount of computing the ciphertext message and generating a new key matrix will be greater, communication load in the network will be greater and the confidentiality of the system will be relatively better. In addition, because the dynamic update features of the key matrix can ensure network security, so you can reduce the calculation energy consumption and communication load which are generated by updating key information by reducing the number of the n key matrix, and also reduce the storage overhead of the sensor nodes.

Suppose you want to send a plaintext message which length is L and the order of dynamic key matrix is n, then $L = \boldsymbol{K}_{ij} \times \boldsymbol{D}_i^{\mathrm{T}}$, namely:

$$c = \sum_{i=1}^{n} \left(\sum_{j=1}^{n} \boldsymbol{K}_{ij} d_j \right)$$

The computational load of the process of generating ciphertext: look-up table for n times, do multiplicative operations n^2 times, do additive operator $n(n-1)$ times; updating the dynamic key requires n times Hash matrix to calculate and N^2 times operation on checking table. Therefore, minimizing the order of the matrix under the premise that security is guaranteed is of great signifycance in improving the operation speed of this agreement, reducing the computation time and lowering energy consumption.

Convert the information into a relatively short ciphertext so that the communication load is reduced while the safety is ensured.

Each node only needs to maintain the communication key of the adjacent neighbor in this protocol. Comparing with the key pre-distribution scheme which need to store a lot of key information, under the circumstances of lower key storage cost, this Protocol can maintain the full connectivity of sensor network in the range of communication.

5. Conclusions

This article proposes a kind of dynamic key management protocol based on password matrix in wireless sensor network. It uses the update mechanism about dynamic password matrix. Each transmission of information using the different encrypted and unencrypted keys, effectively prevents physical capture, eavesdropping and other attacks and enhances the security of the system; the connectivity of network nodes is good and radio range is wide between adjacent nodes connected; as the agreement only need to establish key matrix between the nodes in the neighborhood, it can effectively reduce space where the node store key; information delivery process and key update communication cost are at the same time so that it can reduce the overhead in the transmission of session keys between nodes and save energy consumption. Meanwhile, putting forward the joining system about the new nodes makes the network have better scalability, and when the new sensor node is added to the network, the network key which has been leaked will not affect the new join intrant sensor nodes.

6. References

[1] B. C. Neuman and Ts'o T. Kerberos, "An Authentication Service for Computer Networks," *IEEE Communications*, Vol. 32, No. 9, 1994, pp. 33-38. doi:10.1109/35.312841

[2] S. Slijepcevic, V. Tsiasis and S. Zimbeck, "On Communication Security In Wireless Ad-Hoc Sensor Networks," *Proceedings of the* 11*th IEEE International Workshops on Enabling Technologies*: *Infrastructure for Collaborative Enterprises*, Pittsburgh, 2002, pp. 139-144.

[3] D. Huang, M. Mehta, A. V. D. Liefvoort, *et al.*, "Modeling Pairwise Key Establishment for Random Key Predistribution in Large-Scale Sensor Networks," *IEEE*/*ACM Transactions on Networking*, Vol. 15, No. 5, 2007, pp. 1204-1215. doi:10.1109/TNET.2007.896259

[4] Y. W. Law, L. H. Yen, R. D. Pietro, *et al.*, "Secure K-Connectivity Properties of Wireless Sensor Networks," *Proceedings of the* 4*th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems*, Pisa, 8-11 October 2007, pp. 1-6.

[5] S. A. Camtepe and B. Yener, "Combinatorial Design of Key Distribution Mechanisms for Wireless Sensor Networks," *IEEE*/*ACM Transactions on Networking*, Vol. 15, No. 2, 2007, pp. 346-358. doi:10.1109/TNET.2007.892879

[6] F. Delgosha, E. Ayday and F. Fekri, "MKPS: A Multivariate Polynomial Scheme for Symmetric Key-Establishment in Distributed Sensor Networks," *Proceedings of the ACM International Wireless Communications and Mobile Computing Conference*, Honolulu, 12-16 August 2007, pp. 236-241.

[7] J. Wu and D. R. Stinson, "Minimum Node Degree and K-Connectivity for Key Predistribution Schemes and Distributed Sensor Networks." *Proceedings of the* 1*st ACM Conference on Wireless Network Security*, Alexandria, 31 March-April 2008, pp. 119-124.

[8] H. Chan and A. Perrig, "PIKE: Peer Intermediaries for Key Establishment in Sensor Networks," *Proceedings of the IEEE Computer and Communications Societies*, Miami, 2005, pp. 524 -535. doi:10.1109/INFCOM.2005.1497920

[9] S. A. Camtepe and B. Yener, "Combinatorial Design of Key Distributions for Wireless Sensor Networks," *Proceedings of the European Symposium on Research Computer Security*, Sophia Antipolis, 2004, pp. 293-308.

[10] D. Liu, P. Ning and W. Du, "Group-Based Key Predistribution for Wireless Sensor Networks," *ACM Transactions on Sensor Networks*, Vol. 4, No. 2, 2008, pp. 1-30. doi:10.1145/1340771.1340777

[11] N. T. Canh, P. T. H. Truc, T. H. Hai, *et al.* "Enhanced Group-Based Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge," *Proceedings of the* 6*th Annual IEEE Consumer Communications and Networking Conference*, Las Vegas, 10-13 January 2009, pp. 1-5.

[12] R. Anderson, H. Chan and A. Perrig, "Key Infection: Smart Trust for Smart Dust," *Proceedings of the* 12*th IEEE International Conference on Network Protocols*, Berlin, 2004, pp. 206-215. doi:10.1109/ICNP.2004.1348111

[13] C. Bekara and M. Laurent-Maknavicius, "A New Protocol for Securing Wireless Sensor Networks against Nodes Replication Attacks," *Proceedings of the* 3*rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, New York, 2007. http://doi.ieeecomputersociety.org/10.1109/WIMOB.2007.11

[14] Q. B. Yin, L. R. Shen, R. B. Zhang, *et al.*, "A New Intrusion Detection Method Based on Behavioral Model, Intelligent Control and Automation," *Proceedings of the* 5*th World Congress on Intelligent Control and Automation*, Hangzhou, 15-19 June 2004, pp. 4370-4374.