A Random Key Management Scheme Based on Hash Chain in Wireless Sensor Networks

Chuiwei Lu, Zhiyuan Liu, Peng ji

Computer Institute, Hubei Polytechnic University, Huangshi, Hubei, 435003, China

Corresponding author is Chuiwei Lu

Abstract

Key Management Scheme is an important factor to improve the safety performance of Wireless Sensor Networks (WSNs). The existing schemes can't simultaneously meet the high demand of connectivity and anti-destruction to WSNs. We propose a key management scheme base on random Hash chain. The scheme firstly does multidimensional treatment for the key pool, and then uses a Hash Function to hash the key pool into several of key chains, which raises the relation among the keys. The high-relation makes any twice selection of key-set to contain one same key at least, which highly raise the connectivity of WSNs. Our scheme also deploys a small number of auxiliary nodes chain in WSNs. The chain will help WSNs to authenticate node identity and distribute keys among the isolated nodes. Simulation experiments demonstrate that the scheme not only raises the connectivity of WSNs, but also highly enhances the anti-destruction of WSNs.

Keywords: HASH CHAIN, KEY MANAGEMENT SCHEME, WIRELESS SENSOR NETWORKS

1. Introduction

Judging from the current researches, the key management model based on key pre-distribution is most suitable for WSNs. Before deploying the WSNs, a certain number of keys are stored in each sensor and a key-share algorithm is also stored. After deploying the WSNs, all the nodes use the pre-store information and the simple negotiation mechanism to get the shared keys. At the same time, the requirements to the performance of the nodes are also low in the model. Therefore, the key management problems in WSNs are transformed to design a suitable key management scheme based on WSNs's application requirements and network characteristics. In recent years, many scholars have done substantial research on the pre-distribution key of WSNs.

L. Eschnauer et al. first proposed a random key pre-distribution scheme (E-G scheme) based on probability[1].In the initialization step of WSNs, there are n keys are stored in each node, and the keys are randomly selected from a large key pool with a total key number of K. Therefore, the probability that a pair of nodes can directly establish the secure communication link is $p = 1 - C_{K-n}^{n} / C_{K}^{n}$. According to the theory of random graphs, we can select the appropriate n to ensure that the expected probability of the connection of the entire WSNs reach P. The E-G scheme has its drawbacks. If many keys are shared by multiple nodes, the anti-destruction of the scheme will be greatly reduced. In order to enhance the anti-destruction capability of the E-G scheme, the paper [2] does much improvement on the E-G scheme and put forward a key pre-distribution scheme named y-composite. Compared with E-G scheme, it increased the number of the shared keys to q and proposed a multi-path key reinforcement scheme, which improved the

network connectivity and anti-destruction but limits the scale of the network. Hence the scalability of the scheme is poor. Blom et al. proposed a new key generation scheme based on matrix [3]. Through the symmetric properties of the matrix, a new distribution of the session key is established, so that the key-pair can be directly established by any two nodes in the WSNs. However, the scheme has a bad feature with *t*-threshold, that is to say, if the nodes captured by enemy exceeding *t*, all the security communication links will be broken. Therefore, on this basis, many scholars have proposed some improved schemes [4-6] to increase the scalability of the WSNs. Blundo et al. proposed a scheme for the pre-distribution of the key-pair with the method of a symmetric two-variable polynomial on the finite field [7]. After the sensor nodes are deployed, the shared keys among them are calculated by polynomial. But as with Blom's scheme, the scheme also has the bad security features with *t*-threshold. Based on the schemes of Blumdo, D. Liu and P. Ning proposed a scheme (L-N scheme)[8-9] with random key pre-distribution of polynomial. During the procedure of keys generation, the keys are not distributed to nodes directly, instead, the scheme allots polynomials to related nodes to generate the session keys, which avoiding the leakage of the keys and improving the survivability of the WSNs. A number of scholars have proposed the scheme of key pre-distribution [10-13] based on deployed information of sensor node, that is to say, using the location information of the nodes to improve the performance of key pre-distribution scheme. In addition, some valuable solutions are also proposed, such as the key distribution schemes based on the block hybrid design[14-16] and based on the energy dispatching [17-19].

The study found that the aforementioned random key pre-distribution schemes have the following shortages:

(1) Network connectivity is low. Some nodes may have no shared keys with their adjacent nodes so that it's impossible to establish the secure communication connection between them. Thus, the isolated nodes are generated and reduce the connectivity of the network.

(2) Algorithm of secure communication link has the shortages of long path, high cost and low growth rate in connectivity.

(3) The anti-destruction is low. If some nodes are captured by enemies, enemies can extract the keys stored in the nodes and defraud the trust of other normal nodes, which can destroy most of the security communication links in the WSNs.

To overcome those disadvantages, the paper proposed a random key management scheme based on Hash Chains (abbreviates KMSRHC). The scheme divides all the wireless sensor nodes into two categories. One is the majority named ordinary nodes with data communication function which are relied on to transmit data. The other is the minority named auxiliary nodes which are distributed in the WSNs according to the λ-Poisson model, mainly responsible for the key distribution and data delay. In the KMSRHC scheme, the main key pool is hashed into a number of key chains to enhance the relation among the keys. At the same time, the identifiers of the auxiliary nodes are also hashed, which generate an auxiliary chain that helps to authenticate the node identity and assist the key distribution. The results of simulation experiments show that these improvements greatly enhance the connectivity performance of WSNs, reduce the cost, and improve the anti-destruction performance of WSNs.

The KMSRHC scheme consists of three processes: Key pre-distribution, Share-key exchange and Establishment of communication path.

2. Key Pre-distribution

Firstly, *n* keys are randomly selection from the key pool as the head-key in the group *n* and then generate a key chain with the help of the Hash Function, meanwhile, assign continuous identifiers to each key. Secondarily, other *n* keys are randomly selected as the head-key of the second key chain which is also generated through the Hash Function. The identifier of the first key is ID_{n+1}, the other identifiers adding 1 sequentially, and so on. *L* key-chains can be generated and all the key identifiers are from ID_1 to ID_{n+L} sequentially. To improve the security of key generation and key distribution, we use two hash functions to deal with the key pool. The functions are named H_S and H_M. The former employs the SHA-1 algorithm and the latter uses MD5 algorithm. They make the following formula set up together:

$$\begin{aligned} k_i &= H_s(seed_i) \\ Identify_i &= k_i \oplus H_M(seed_i) \end{aligned} \quad (1)$$

In formula 1, the k_i is the key in the key pool, $seed_i$ is the public key seed, and $Identify_i$ is the identification code of the key chain. The Hash Function H_S can help to generate a set of hash key chains for the key pool and produce corresponding identification codes for those hash chains. The hash function H_M can generates another set of hash chain, one-to-one corresponding to the hash chain generated by H_S, which is used for the authentication of the chain key. We set up a small number of auxiliary nodes and a pseudo

random function F_R to ensure the security of the key-distribution. The hash function H_S is employed to generate a hash chain called $H_s(FD_i)$, meantime, the H_M to generate another corresponding hash chain called $V_i = H_M(H_s(FD_i))$, which is responsible for the authentication of the identifiers of auxiliary nodes. The popular node identifier in the wireless sensor network is ID_i and the auxiliary node identifier is FD_i. The structure of the key pool is shown as the figure 1.

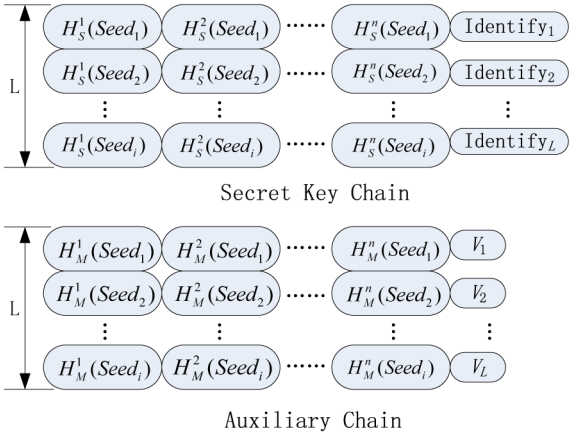

Figure 1. Key Chain Partition

Before distributing the keys, popular nodes get the H_M from the key pool, the pseudo random function F_R, the verification code V_i of n auxiliary chains as well as the identifier ID_i of the neighbor node, and then select a verification code from the V_i randomly to construct the auxiliary session key. The auxiliary nodes select a key and the corresponding $seed_i$ randomly from each chain key, and then employ the function F_R to diffuse and encrypt the hash chain. According to the above information, we can calculate the corresponding position of the chain key on the auxiliary chain. It is shown as formula 2.

$$K_{x,y} = \bigcap_{i=1}^{n} H_M(seed_i) \oplus H_s(seed_i) \tag{2}$$

The final identifier of the auxiliary node can be calculated according to the original identifier FD_i of the auxiliary node and the times m that the identifier has been hashed. It is shown as formula 3.

$$FD_{Last} = \bigcup_{i=1}^{m} H_M(FD_i) \oplus H_s(FD_i) \tag{3}$$

3. Share-key exchange

Before the wireless sensor network start to work, the auxiliary nodes diffuse the chain key and its identifier as well as the corresponding position $K_{x,y} = \bigcap_{i=1}^{n} H_M(seed_i) \oplus H_s(seed_i)$ of the chain key on the auxiliary chain. After multiple transformations, the final identifier of the auxiliary node would be $FD_{Last} = \bigcup_{i=1}^{m} H_M(FD_i) \oplus H_s(FD_i)$, and the identifier will have been diffused m times. In the step, auxiliary node needs to broadcast many data packets, meanwhile, build an only key-pair between two neighbor nodes.

In our scheme, each wireless sensor node is stored n keys in advance and then deployed to the target area. Afterwards, the adjacent nodes will automatically do the work of share-key discovery periodically. To reduce the channel collision, each node uses the algorithm, named random binary exponential backoff, to broadcasts the message that the node discovered. The neighbor nodes must send a response message after receiving the message so that the node knows the distribution of its neighbors. Afterwards, the two adjacent nodes will establish their key-pair according to the message received. Supposing nodes u and v are neighbors, the specific process is as follows.

The node u first broadcasts a key-discovery message to all its neighbor nodes. The message is composed of the following items.

(1)The identification ID_u of the node u

(2) Random number R_u

(3) Plaintext T_u

(4) Ciphertext Cu_i, $1 \le i \le n$

The packet format of the message is shown as $\langle ID_u, R_u, T_u, Cu_i \rangle$, likewise, the node v will also broadcast a similar key-discovery message, and its packet format is shown as $\langle ID_v, R_v, T_v, Cv_i \rangle$. Whenever the node v and node u receive each other's key-discovery message, they must reply a message to testify their existence. The packet format of the message is shown as follows.

$$\begin{aligned} v &\rightarrow *\langle ID_v, T_u, x_v, y_v \rangle \\ u &\rightarrow *\langle ID_u, T_v, x_u, y_u \rangle \end{aligned} \tag{4}$$

As is shown in formula 4, the first packet is sent out by node v and the second by node u. The x and y represent the coordinates of the node in the deployed region. Subsequently, the nodes v and u employ the shared-key discovery algorithm to find out all the shared keys among them, which ensure their later secure communication. The specific methods are shown as follows:

After receiving the key-discovery message from node v, the node u will use the n pre-store keys by itself to decrypt ciphertext Cv_i from the message of the node v. After that, the decryption results will be compared with T_v. If the result is the same, the key is shared by the two nodes, thus the key is incorpo-

rated into the share-key set, and its shared scope is labeled.

The format of share-key set is $K_i(u,v),1 \le i \le n$, in which the i is the number of the key K in the key pool, the $K_i(u,v)$indicates that the key K_i is shared by node u and node v. In the same way, the node v also finds out the sharing key with node u trough the above method. All sharing keys are put into a special set $K_j(v,u),1 \le j \le n$ for secure communication between two nodes. To enhance the security performance, the formula 1 is used to generate the hashed key chain and verification codes for the above mentioned key-set.

In order to improve the speed of random selection, we do XOR operation for the shared key set $K_i(u,v)$ of node u and node v. We denote $K_{u,v} = k_1 \oplus k_2 \oplus k_3 \quad \oplus k_i$, in which the parameters from k_1 to k_i belong to the $K_i(u,v)$. We also denote $S_p K_{u,v}$ as the temporary shared session key between the node u and v. The session key is commonly generated by the shared key of the node u and v as well as two random integers. That is to say, the shared key among the nodes is not the final communication key. Even if the node is captured by the enemy and the shared keys is revealed, the enemy can't contact other normal nodes through them. For the enemy lacks complete materials to rebuild the session key. The method will further improve the security of nodes communication. The $S_p K_{u,v}$ is automatically generated by the function $Z = H_{trapdoor}(x,y)$. The using method is shown as follow.

$$S_p K_{u,v} = H_{trapdoor}(k_{u,v}, Rand_u \oplus Rand_v) \qquad (5)$$

In the formula 5, the parameters $Rand_u$ and $Rand_v$ are two integers which are randomly generated by node u and node v. the two parameters are responsible for identifying communication sessions. The $K_{u,v}$ is a shared key between node v and node u. $Z = H_{trapdoor}(x,y)$ is a kind of trapdoor hash function, and is pre assigned to all nodes in the WSNs to verify the integrity of the key chain. It has the following characteristics: for a given x, if the parameter y is unknown, it is impossible to calculate Z; for a given Z and y, it is also impossible to calculate the x reversely.

In order to reduce the probability of message collision and further improve the security of WSNs, the parameters x and y are divided into 64 blocks, which can reduce the collision probability to 2^{-64}.

At the same time, the hash function has fast computing speed and low energy consumption, which is very suitable for key chain authentication, especially for the wireless sensor network with insufficient resources.

Any pair of neighbor nodes can establish the unique session key though the above method and then perform identity authentication, data encryption and other tasks though the session key. After the establishment of the session key, the nodes must erase the shared key $K_{u,v}$ as well as the random integer $Rand_u$ and $Rand_v$ from memory so that the enemy couldn't steal the vital information through the captured sensor nodes.

4. Establishment of communication path

To assign n keys for each node is the core task of key-share of wireless sensor network, and the n keys are randomly selected from a key pool containing $N(N>2n)$ keys, so there is some probability that the neighbor nodes share one key at least, and the probability is named P_{share}. There are C_N^n ways to select the n keys from N keys unrepeatedly. Thus, the probability that any two selection don't contain the same key is C_{N-n}^n / C_N^n .

$$P_{share} = 1 - \frac{C_{N-n}^n}{C_N^n} = 1 - \frac{(N-n)!}{N!(N-2n)!} \qquad (6)$$

Judging from formula 6, it's not difficult to find that the P_{share} is influenced by the value of N and n. The KMSRHC scheme which is able to make the result of n/N as small as possible so as to ensure any two selection get at least one same share-key. The scheme improves the connection efficiency of the nodes and connectivity of WSNs. If the two neighbor nodes need to establish a secure communication link but have no shared key, they will use other nodes as the relay stations to complete the task in auxiliary. Thus we need to solve two problems. Firstly, the security of the key should be ensured when it is transmitted in the communication channel; secondly, extra cost should be spent to build the multipoint communication connection.

Our solution is as follows: If there is no shared key between the two nodes, a number of auxiliary nodes with shared key will be employed as the relay station to assist the establishment of secure communications link. After the data of node u is encrypted using the shared key with the neighbor auxiliary nodes, it will be sent out. The auxiliary node will receive and decrypt the data, then re-encrypt the data using the key shared with next auxiliary node and send the data to this auxiliary node. In this way, the enciphered message will finally be sent to the target node v. This is a pattern of relay transmission, and there are multiple times of encryption and decryption in the process, which will highly raise the difficulty to the enemies to crack the enciphered message.

5. Performance Analysis and Simulation Experiment

In order to verify the performance of the KMSRH scheme, we compare it with the typical key management scheme E-G and L-N. The comparing contents are the connectivity of network and the anti-destruction performance of WSNs. We employ Opnet10 to design a WSNs with 10000 nodes to do these experiments. The simulation parameters are shown in the table 1.

Table 1. Simulation Parameters

Parameter Name	Value
The number of WSNs nodes	10^4
Capacity of Key Pool	10^5
Network coverage area	500m×500m
Communication radius of node	r = 40m
Node degree	d = 50

5.1. Network Connectivity Experiment and Analysis

WSNs will spend more communication cost due to low network connectivity, which weakens the performance of the network. According to the random graph connectivity theory [20], there exist a threshold $P = \frac{\ln N}{N} + \frac{c}{N}$ that make nodes from non-connectivity to the basic connectivity in the key-share graph of N nodes. In above formula, N is the number of the network nodes, and c is a constant which is determined by the following formula.

$$\lim_{x\to\infty}(P_r[G(N,P)connected]) = e^{-e^{-c}} \quad (7)$$

In the formula 7, the threshold P represents the probability that any two nodes have secure link in the key-share graph. Hence P can also be represented as $P = \frac{d}{N-1}$, in which d is the average number of nodes connections, also known as the node degree. Due to the limited resources of wireless sensor nodes and communication ability, each node can establish secure communication links with at most d neighbor nodes. Then P can only be represented as $P = \frac{d}{n-1} \approx \frac{d}{n}$, in which n is the neighbor number of some node in its wireless communication range.

In order to check up the network connectivity of the KMSRHC, we compare the KMSRHC with scheme E-G. According to formula 5, we use four key parameters in the experiments, namely, N =10^5, n = 150, N=10^5, n = 100, N =10^5, n = 75, N =10^5, n = 50. The experiment results are shown as follows.

The scheme E-G, based on the random graph theory, can't guarantee that there is at least one share-key between two neighbor nodes, which makes some nodes isolated from each other. If there is no share-key between the neighbor nodes, they must depend on the help of some relay nodes to indirectly exchange message and establish key-pair so that the data can be transmitted securely.

As is shown in figure 2 to 5, under the four different parameters, the probability of the scheme E-G successfully establishing a secure communication link with only one hop is respectively 0.37, 0.42, 0.35, and 0.28. While, the establishment of most of the security links needs 2 to 3 hops, namely, the communication link should be helped by 1 to 2 relay nodes. In fact, the secure link established though multiple relay nodes not only has security risks, but also needs more cost.

The KMSRHC divides the key pool into several logical structures and hashes them into a series of key chains, thus, the probability that the neighbor nodes share at least one key rises massively. That is to say,

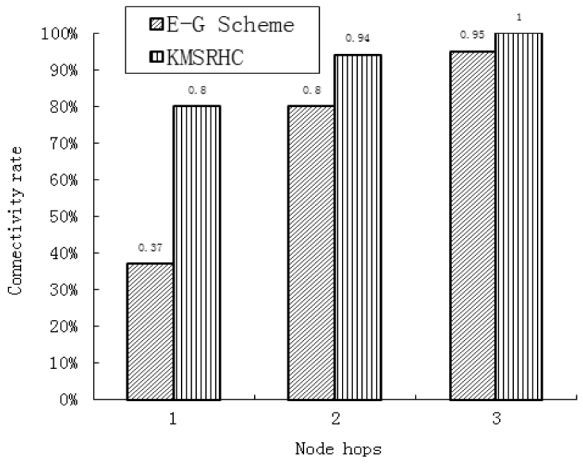

Figure 2. Parameter N =10^5, n = 150

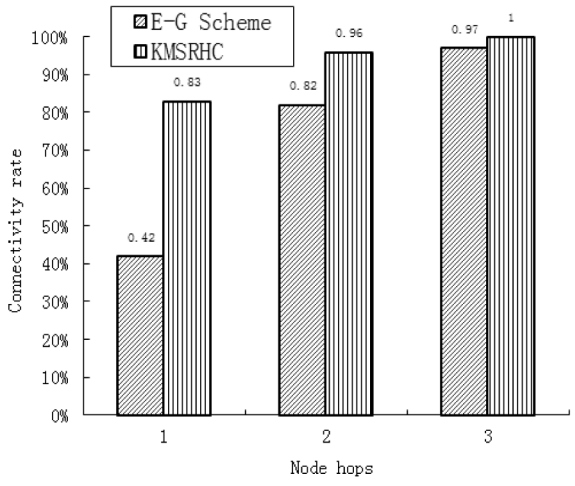

Figure 3. Parameter N =10^5, n = 100

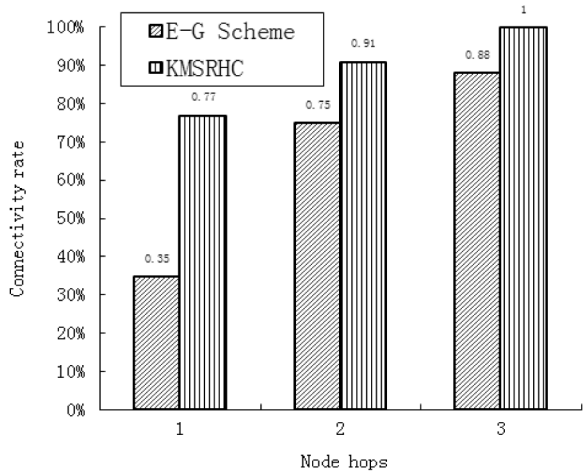

Figure 4. Parameter $N=10^5$, $n = 75$

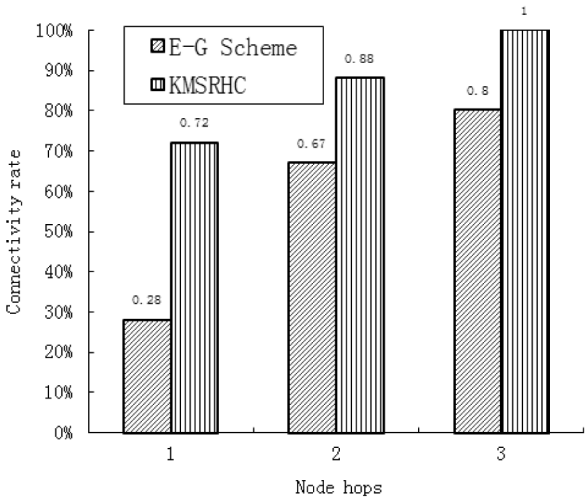

Figure 5. Parameter $N=10^5$, $n = 50$

the scheme KMSRHC could complete the establishment of the secure communication link in one hop basically.

5.2. Analysis of node anti-destruction

Anti-destruction performance refers to the ability that the captured nodes prevent the enemies from cracking the pre-store keys, which is the major safety indicator to evaluate the key pre-distribution scheme in the wireless sensor network. The higher the anti-destruction ability is, the more difficult it will be for the attackers to take advantage of the sensitive information in the captured nodes to crack the secure communication link. We assume that the enemy can capture wireless sensor nodes with a probability of 100%. In the scheme L-N, each sensor node store n keys, then for a given key in primary key pool, the probability that the key doesn't belong to the node is $1-\frac{n}{N}$. If there are m nodes captured, for a given key in primary key pool, the probability that the key doesn't belong to the m nodes is $\left(1-\frac{n}{N}\right)^m$. Meanwhile, supposing the probability of a key belongs to the m nodes is P_d, and then it can be illustrated as $P_d = 1-\left(1-\frac{n}{N}\right)^m$.

We only employ the parameter $N=10^5$, $n = 100$ to do the experiments. Under the parameter, most key management schemes get the best network connectivity rate. In the experiments, the scheme L-N and E-G are the comparing objects. There will be 20 experiments and the average value will be taken as the final result. The experiment result is shown as follows.

The anti-destruction performance of the scheme E-G is determined by the captured nodes number and the parameter n/N. The scheme doesn't take effective security measures to protect the shared keys pool, and that any node is captured may reveal the shared keys, which will cause the communication links of normal nodes cracked by the enemy. As is shown in figure 6, the anti-destruction performance of scheme E-G with is the worst. If 18% nodes are captured, more than 85% communication links of the normal nodes will be cracked by the enemy. The poor security performance is related to its incomplete security policies.

The KMSRHC demonstrates good anti-destruction ability, and it hash the key pool into a set of key chains, and any pair of nodes should establish the session key for the current communication link through the random integer $Rand_x$ and the function $H_{trapdoor}$. To crack the session key, the enemy must know the two input parameters of formula 4. In the scheme KMSRHC, the sensitive parameters will be erased from its RAM after the session key is established. Naturally, the enemy can not get these two parameters from the captured nodes to break the session key. So the cracked rate of communication link in the KMSRHC is very low than the scheme E-G and L-N. In addition, the session key between a pair of neighbor nodes is unique so that the enemy cannot obtain it from the

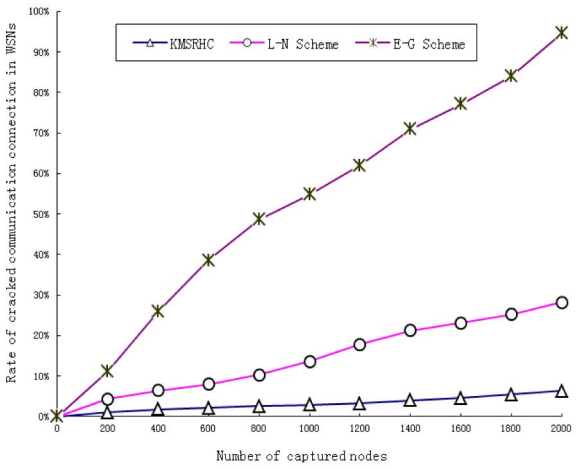

Figure 6. Relationship between captured nodes number and destroyed degree of the WSNs

key pool in other captured nodes. Judging from the experimental results, the strategy brings in better anti-destruction effect so that the enemy cannot crack the session key within the given time.

6. Conclusions

Some famous key management schemes in wireless sensor network, such as scheme E-G and L-N, have two major defects. Firstly, when the scale of the network is large, each node must be allocated more keys to ensure the high connectivity, which may reduce the scalability of the scheme. Secondly, some key *k* may be shared by multiple nodes who will reveal the sensitive message of the shared keys when they are captured, which will threat the communication security among the normal nodes. In view of the above problems, we proposed a random key management scheme based on hash chain. After multidimensional processing to the key pool, any two key-selections have at least one shared key and thus improving the connectivity performance of the WSNs.

In the scheme KMSRHC, the trapdoor hash function is employed to establish a unique key-pair between the neighbor nodes. Afterwards, all the sensitive information about key-pair will be erased so that the enemy couldn't directly obtain or crack the key-pair through the captured nodes. The hash chain technique is also applied to the KMSRHC. The main key pool is hash into several key chains to improve the correlation degree among the keys. Moreover, a few auxiliary nodes are set to help to authenticate the node identity and distribute keys, which further improve the security performance of the KMSRHC. Simulation experiments indicate that the KMSRHC has better connectivity performance and anti-destruction ability than the common schemes, such as scheme E-G and L-N. Furthermore, its scalability is good and can be applied to large-scale wireless sensor networks.

Acknowledgements

The work is supported by the Science Research Project of Hubei Provincial Department of Education (D20144403, B2014026), the Outstanding Youth Science and Technology Innovation Team Project of Hubei Polytechnic University (13xtz10), the Natural Science Foundation of Hubei Province (2013CFB039).

References

1. L. Eschenauer, V. D. Gligor. A key-management scheme for distributed sensor networks. Proceedings of Conference "ACM CCS-2002", Baltimore, 2002, pp.41-47.
2. H. Chan, A. Perrig, D. Song. Random key predistribution schemes for sensor networks. Proceedings of Conference "ISRSP-2003", Richmond. 2003, pp. 197-213.
3. R. Blom. An optimal class of symmetric key generation systems, Proceedings of Conference "LNCS-1985", Paris, 1985, pp. 335-338.
4. W. DU, J. Deng, Y.S. Han. A key pre-distribution scheme for sensor networks using deployment knowledge. IEEE Trans. on Dependable and Secure Computing, 2006, vol.3, no.2, pp.62-77.
5. Y. Zhen, G. Yonb. A key management scheme using deployment knowledge for wireless sensor networks, IEEE Trans. on Parallel and Distributed Systems, 2008, vol.19, no.10, pp. 1411-1425.
6. F. Kai, C. Liu and Q. Dong. Collusion problem of the EBS-based dynamic key management scheme. Journal of Software, 2009. vo1.20, no.9, pp.2531-2541.
7. C. Blundo, A. De Santis, A. Herzberg. Perfectly secure key distribution for dynamic conferences, Proceedings of Conference "LNCS CRYPTO-1992", London, 1993, pp. 471-486.
8. D. Liu and P. Ning. Establishing pair wise keys in distributed sensor networks, Proceedings of Conference "ACM CCS-2003", New York, 2003, pp.52-61.
9. D. Liu, P. Ning, W. Du. Group-based key predistribution for wireless sensor networks, IEEE Trans. on Sensor Networks, 2008, vol. 4, no.2, 1-30.
10. Wuu, LC; Hung, CH; Chang, CM. Quorum-based Key Management Scheme in Wireless Sensor Networks. KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS. 2012, vol.6 no.9, pp. 2442-2454
11. Yao, WB; Han, S; Li, XY. LKH plus Based Group Key Management Scheme for Wireless Sensor Network. Wireless personal communication, 2015, vol.83, no.4, pp.3057-3073
12. Chen, SD; Wei, HH. Key Pre-distribution Schemes based on Symplectic Geometry for Wireless Sensor Networks, 2015, vol.121, no.6, pp. 97-112
13. Erfani, SH; Javadi, HHS; Rahmani, AM. A dynamic key management scheme for dynamic wireless sensor networks. Security and communication networks. 2015, vol.8, no.6, pp.1040-1049
14. Suganthi, N; Sumathy, V. Energy Efficient Key Management Scheme for Wireless Sensor

Networks. International journal of computers communication & control. 2014, vol.9, no.1, pp.71-78
15. Pan Zhongqiang; Chang Xinfeng. A Distributed Key Management Scheme in Wireless Sensor Update. Journal of Sensors and Actuators, 2014, vol.27, no.9, pp.1287-1292
16. Zhou, YC; Wang, T; Wang, YF. A novel WSN key pre-distribution scheme based on group-deployment. 2014, vol.15, no.3, pp.143-148
17. Alcaraz, C; Lopez, J; Roman, R; Chen, HH. Selecting key management schemes for WSN applications. Computers & Security. 2012, vol.31, no.8, pp. 956-966
18. Kim, HY; Lee, C. A Key Management Scheme for Security and Energy Efficiency in Sensor Networks. Journal of internet technology. 2012, vol.13 no.2, pp.223-231
19. GAO Wei; GAO Tiegang. Improved hash chain based key distribution scheme. Application Research of Computers. 2011, vol.28, no.5, pp.1886-1888
20. Engel, A; Monasson, R; Hartmann, AK. On large deviation properties of Erdos-Renyi random graphs. Journal of statistical physics. 2004, vol.117, no.4, pp.387-426

Super-Threshold Computing of FinFET Flip-Flops

Dongmei Li, Jianping Hu, Yuejie Zhang

Faculty of Information Science and Technology, Ningbo University, Ningbo, 315211, China

Corresponding author is Jianping Hu

Abstract
Reducing the source voltage can effetely lower power dissipation of flip flops, but resulting in its performance degradation. This paper presents super-threshold computing of flip flops. The four typical flip flops, named as transmission-gate master-slave flip-flop, simplified static master-slave flip-flop, clocked CMOS master-slave flip-flop, and TSPC dynamic master-slave flip-flop operating on medium strong inversion regions are investigated in terms of settling time, propagation delay, power consumption, and power delay product. All circuits are simulated with HSPICE at a PTM (Predictive Technology Model) 32nm FinFET technology. The simulation results show that super-threshold FinFET flip flops operating on medium strong inversion regions attain about 38% power reduction with a penalty of only about 16%.
Keywords: FINFET, SUPER-THRESHOLD, FLIP-FLOP, LOW-POWER CONSUMPTION

1. Introuduction

As transistor size scales down, short channel effect of bulk MOS transistors results in very significant leakage power consumption [1, 2]. FinFETs as a 3D device have better turn-on current and lower leakage current compared with bulk MOS transistors, and