ID-Based Multiple Space Key Pre-distribution Scheme for Wireless Sensor Networks

Tran Thanh Dai and Choong Seon Hong*

Networking Lab, Department of Computer Engineering, Kyung Hee University, Korea
daitt@networking.khu.ac.kr, cshong@khu.ac.kr

Abstract. Providing security services for wireless sensor networks plays a vital role in secure network operation especially when sensor networks are deployed in hostile areas. In order to pave the way for these mechanisms, cryptographic keys must be agreed on by communicating nodes. Unfortunately, due to resource constraints, the key agreement problem in wireless sensor networks becomes quite intricate. To deal with this problem, many public-key unrelated proposals have been proposed so far. One prominent branch of these proposals is based on random key pre-distribution. Inspired by this trend, in this paper, we propose a new random key pre-distribution scheme that is comparable to Du et al.'s scheme [2] in terms of network resiliency and memory usage. On the other hand, our later analysis shows that our scheme outperforms Du et al.'s scheme in terms of computational and communication overhead.

Keywords: ID-based, random key pre-distribution, key agreement, security, wireless sensor networks.

1 Introduction

A typical wireless sensor network (WSN) may contain a large number of microsensor nodes, which are connected by a wireless medium, controlled and managed by one or several powerful control nodes (often called base stations). These sensor nodes are tiny in size and capable of capturing various physical properties, such as temperature, humidity, or pressure, and mapping the physical characteristics of the environment to quantitative measurements. The captured and pre-processed information is delivered to base stations as well as other nodes through immediate neighboring nodes. WSNs encourage several novel and existing applications such as environmental monitoring; health care; infrastructure management; public safety; medical; home and office security; transportation; and military.

Deployment of a WSN can be in random fashion (e.g., scattered from an airplane) or planted manually. When being deployed in hostile environments, WSNs are vulnerable to different kinds of malicious attacks. In such contexts, providing security services based on solving the key agreement problem becomes one of the major concerns. Unfortunately, due to resource constraints, the key agreement problem in

* This work was supported by MIC and ITRC Project. Dr. CS Hong is corresponding author.

F. Boavida et al. (Eds.): WWIC 2007, LNCS 4517, pp. 235 – 246, 2007.

WSNs becomes quite intricate. To deal with this problem, many public-key unrelated proposals which are considered more reasonable in cost than public key based approaches have been proposed so far. One prominent branch of these proposals is based on random key pre-distribution [2], [4], [5], [6]. Another outstanding branch is ID-based key pre-distribution schemes [1], [11] which have the following properties: (i) No previous communication is required; (ii) Key pre-distribution procedure consists of simple computations; (iii) In order to establish the key, each party should input its partner's identifier only into the secret key sharing function.

Inspired by these observations, in this paper, we propose a highly resilient, robust, resource-efficient, and ID-based random key pre-distribution scheme. On the one hand, our scheme as being analyzed later is much like Du et al.'s scheme [2] (*Du's scheme* for short) in terms of network resiliency with the same memory cost. In other words, when the number of compromised nodes is less than a threshold, the probability that any nodes except these compromised nodes is security influenced is negligible. This property means that an attacker's gain is decreased for small scale network breach and this gain has a significant security impact only when the attacker mounts a successful attack on a considerable proportion of the network which is considered to be detected easily. On the other hand, our scheme significantly improves resource usage in terms of computational and communication overhead compared to Du's scheme.

The rest of this paper is organized as follows: section 2 mentions the related work; section 3 summarizes our keystone, the Matsumoto-Imai scheme; section 4 describes our ID-based random key pre-distribution scheme; section 5 analyzes the resiliency of our scheme against node capture attack; section 6 presents the performance analysis in terms of memory usage, communication overhead, and computational overhead; section 7 concludes the paper and states our future work.

2 Related Work

In this section, we briefly review several noticeable random key pre-distribution schemes for WSNs that have been published in the literature so far.

Eschenauer et al. [4] are the first to propose a random key pre-distribution scheme that relies on probabilistic key sharing among the nodes of a DSN. The main idea is that a random pool of keys is selected from the key space. Each sensor node then receives a random key ring from the key pool before deployment. After deployment, any two neighboring nodes able to find a common key within their respective key rings using *shared-key discovery phase* can use that key as their shared secret to initiate communication and to set up the secure connection. In the case that those nodes could not find a common key, they can resort to *path-key establishment phase* to solve the key agreement issue.

Chan et al. [5] further exploited the idea in [4] to developed three mechanisms for key establishment using the framework of pre-distributing a random set of keys to each node. The first one is q-composite keys scheme. This scheme is mainly based on [4]. The difference between this scheme and [4] is that q common keys, instead of just a single one, are needed to establish secure communication between a pair of nodes. The second one is multi-path key reinforcement scheme applied in conjunction with [4] to yield greatly improved resiliency against node capture attack by trading off

some network communication overhead. The third one is random pairwise keys scheme. The purpose of this scheme is to allow node-to-node authentication between communicating nodes.

Du et al. [2] presented a multiple space key pre-distribution scheme for wireless sensor networks. This scheme first uses Blom's key generation scheme [12] as a building block to generate multiple key spaces, a pool of tuple (D, G), where matrices D and G are as defined in Blom's scheme. Then this pool is used as a pool of keys as in [4] to establish a common secret key between any pair of nodes.

Chan et al. [13] proposed a variant of random key pre-distribution scheme for key agreement problem in the clustered DSN. Accordingly, the DSN is sub-grouped into clusters. Different clusters in different regions are assigned different probabilities of node compromise based on the hostile level of those regions. Within each cluster, the scheme in [4] is applied. This scheme is claimed to isolate the effect of node compromise into one specific subgroup and offer an effective scalable security mechanism that increases the resiliency to the attacks on the sensor subgroups.

3 Matsumoto-Imai Scheme (MI Scheme)

First of all, in this paper, we assume that each sensor node has a unique identification whose range is from 1 to N where N is the maximum number of deployable nodes. Each of the unique identifications is represented by $m = \log_2(N)$ bit effective ID in sensor nodes' memory.

This section explains how the sensor nodes' secret keying material is generated and how sensor nodes use this material to establish pairwise keys in the manner of the MI scheme [1].

A central server first generates $l\,(m \times m)$ symmetric matrices M_ωs over finite field $GF(2)$ These M_ωs are kept secret and must not be disclosed to both attackers and sensor nodes. M_ω is used to generate the $\omega-$th bit of a pairwise key between any pair of neighboring nodes, so l is the length of this key. The central server then computes the keying material for each node S_i as follows:

$$\Phi_i^\omega = y_i M_\omega \quad (\omega = \overline{1,l}) \tag{1}$$

$$\Phi_i = \left[\Phi_i^1 \quad \Phi_i^2 \quad \ldots \quad \Phi_i^l\right]^T \tag{2}$$

where $y_i \ (i = \overline{1,N})$ is the m-dimensional vector, effective ID of node S_i. Φ_i needs to be kept secret in the node S_i and should not be disclosed to attackers or other sensor nodes.

Therefore, when nodes S_i and S_j need to find the pairwise key between them, they first exchange their effective IDs y_i and y_j respectively, then use Φ_i and Φ_j to compute their pairwise key as follows:

$$\boldsymbol{S_i}: \quad K_{ij}^\omega = \Phi_i^\omega y_j^T \ (\omega = \overline{1,l}),\ K_{ij}^T = \Phi_i y_j^T \tag{3}$$

$$\boldsymbol{S_j}: \quad K_{ji}^\omega = \Phi_j^\omega y_i^T \ (\omega = \overline{1,l}),\ K_{ji}^T = \Phi_j y_i^T \tag{4}$$

where symbol T denotes transposition operation. Fig. 1 illustrates how the pairwise key $K_{ij} = K_{ji}$ is generated.

This scheme has a noteworthy property that as long as no more than $m-1$ nodes are compromised, the entire network is theoretically secure. In other words, an attacker needs to compromise at least m nodes in order to compute any pairwise key of any two uncompromised neighboring sensor nodes using their effective IDs.

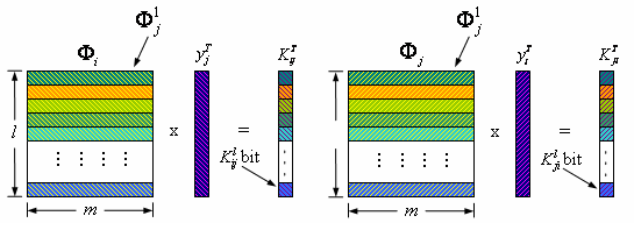

Fig. 1. Pairwise key generating in MI scheme

4 Our ID-Based Multiple Space Key Pre-distribution Scheme

To enhance network resiliency against node capture attack, we propose an ID-based key pre-distribution scheme that uses IM scheme as a keystone in combination with the idea of multiple key spaces proposed in [2]. Accordingly, the entire network is depicted in the graph theory language. There is an edge between two neighboring sensor nodes (two vertices in graph theory) if and only if they can establish a pairwise key between themselves. Using IM scheme we guarantee to create a complete graph. To obtain our aim of key agreement and enhance resilience, all we need is a connected graph, rather than a complete graph since the latter is a very wasteful use of security.

Some terms need to be clarified before detailing our proposed scheme. We define a key space Ω_i as a 3-tuple *(Mi, l, m)* of l $(m \times m)$ matrices $M_{i\omega}$, where $M_{i\omega}$ is defined as in IM scheme. A node is said to choose a key space Ω_i if it carries the secret information generated from Ω_i using MI scheme. Two nodes can derive their pairwise key if they have a key space in common.

4.1 Keying Information Pre-distribution Phase

During this phase, we have to pre-distribute keying material to each node such that after deployment neighboring nodes can derive a pairwise key between them using this material. We also select the security parameters μ, λ, and m, where $2 \le \mu < \lambda$. these parameters are chosen with the security and performance in mind which will be discussed later. This phase is performed as follows:

Step 1 (Generating 3-tuples (Mi, l, m)). A central server generates λ key spaces. Each key space Ω_i consists of l $(m \times m)$ symmetric matrices $M_{i\omega}$s as defined in IM scheme.

Step 2 (Generating Φ_i matrices). We randomly choose μ distinct key spaces from λ key spaces for each node. For each space Ω_i chosen by node S_j, we first compute keying material Φ_{ji} using equations (1), (2) and then store Φ_{ji} at this node. Therefore, each node S_j has μ distinct values of Φ_{ji}s. Using MI scheme; two nodes can derive a pairwise key if they have both chosen a common key space.

4.2 Pairwise Key Establishment Phase

After deployment, each node needs to discover whether it shares any key space with its neighbors. To do so, each node instantly broadcasts a message containing the following information: the node's effective ID and the indices of the key spaces it carries.

Suppose that nodes S_i and S_j are neighbors, and then they have received the above-mentioned broadcast messages. If they figure out that they have an identical index of a key space (or identical key space Ω_s), they can easily compute their pairwise key using equations (3) and (4) of MI scheme respectively. Conversely, there is the case that two nodes who even are neighbors could not establish a pairwise key. To tackle this problem, there are two possible methods that can be used. The first one has already presented in [2]. However this method is vulnerable to node capture attack. Specifically once an attacker can successfully compromise one node on the key path during the key path process, the promising pairwise key K will be disclosed. The second one is a combination of the (k, n) secret sharing method [3] and disjoint path finding methods. Accordingly, S_i first discovers the secured disjoint paths to S_j and then uses the secret sharing method to split K into pieces. After that, each piece is sent on one of the secured disjoint paths as described in [2]. Finally, K can be re-constructed if S_j receives no less than k pieces. For fairness, when making comparison with Du's scheme, only the first method is taken into account.

4.3 Selecting μ, λ

The problem here is that given the size and the density of a network, how we can select the values for μ and λ such that the entire network is securely connected with high probability P_{gc} ? The approach is that we first compute P_{rlc} (the required probability of two neighboring nodes sharing at least one key space in order to obtain the desired global connectivity P_{gc}). Then we compute P_{alc} (the actual probability of two neighboring nodes sharing at least one key space) using μ and λ. Afterward, the values of μ and λ could be found to satisfy the following inequality

$$P_{alc} \geq P_{rlc} \tag{5}$$

Using the result shown in [4], we can obtain the expected degree of a node d (i.e., the average number of secure links between that node and its neighbors) in order to achieve a given value of P_{gc} when N is large:

$$d = (N-1)[\frac{\ln(N) - \ln(-\ln(P_{gc}))}{N}] \tag{6}$$

Using a given density of sensor network deployment and wireless connectivity constraints, the expected number of a node's neighbors n can be estimated. Therefore, the value of P_{rlc} can be estimated as:

$$P_{rlc} = \frac{d}{n} \tag{7}$$

After the values of μ and λ have been selected, the actual probability P_{alc} is determined as follows. Since $P_{alc} = 1 - P$[two neighbors do not share any key space], we have:

$$P_{alc} = 1 - \frac{\binom{\lambda}{\mu}\binom{\lambda-\mu}{\mu}}{\binom{\lambda}{\mu}^2} = 1 - \frac{((\lambda-\mu)!)^2}{(\lambda-2\mu)!\lambda!} \tag{8}$$

For better visualization, we plot the values of P_{alc} in Fig. 2 where λ varies from μ to 100 and $\mu = 2, 4, 6, 8$.

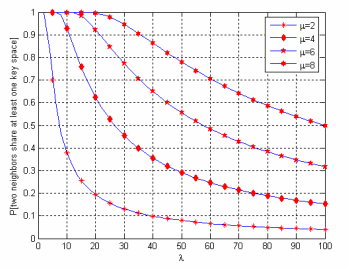

Fig. 2. Probability of finding at least one common key space between two nodes when μ spaces are randomly chosen from λ spaces

Combining equations (5), (6), (7), and (8), we easily derive the following inequality:

$$1 - \frac{((\lambda-\mu)!)^2}{(\lambda-2\mu)!\lambda!} \geq (N-1)\left[\frac{\ln(N) - \ln(-\ln(P_{gc}))}{nN}\right] \tag{9}$$

Correspondingly, to obtain a certain P_{gc} of the entire network connectivity with size N and the expected number of neighbors for each node n, all we have to do is selecting values of μ and λ such that inequality (9) is satisfied.

5 Security Analysis

In this section, we evaluate our proposed scheme concerning its resiliency against node capture. Our evaluation is conducted by finding the answer to two questions:

(i) Given that b nodes are captured, what is the probability that at least one key space is broken? To successfully break one key space, an attacker needs to capture at least m nodes that contain this key space's keying material. Hence, the answer to this question quantitatively shows when the network starts to become insecure. (ii) Given that b nodes are captured, what fraction of the additional communications (communications among un-captured nodes) also becomes compromised? The answer to this question shows how much payoff an attacker can obtain after having captured a certain number of nodes.

5.1 Probability of at Least One Key Space Being Broken

Let B_i denote the event that key space Ω_i is broken, where $i = \overline{1,\lambda}$, and C_b denote the event that b nodes are captured in the network. Moreover, let $B_i \cup B_j$ denote the joint event that either Ω_i or Ω_j, or both is broken. Thus, we have

$$P(\text{at least one space is broken}|C_b) = P(B_1 \cup B_2 \cup ... \cup B_\lambda \mid C_b).$$

Since, $P(B_1 \cup B_2 \cup ... \cup B_\lambda \mid C_b) \le \sum_{i=1}^{\lambda} P(B_i \mid C_b)$ and owing to the fact that each key space has an equal chance to be broken, $\sum_{i=1}^{\lambda} P(B_i \mid C_b) = \lambda P(B_1 \mid C_b)$. Hence,

$$P(\text{at least one space is broken}|C_b) \le \lambda P(B_1 \mid C_b). \tag{10}$$

Our task now is reduced to calculate $P(B_1 \mid C_b)$ - the probability of key space Ω_1 being compromised when b nodes are compromised. The probability that each compromised node contains information about Ω_1 is $\rho = \frac{\mu}{\lambda}$. Let X denote the number of compromised nodes containing information about Ω_1 after b nodes have been compromised. Then, X is a binomial random variable with parameters (b, ρ). Since the event B_1 can only occur after at least m nodes are compromised, we have the following result:

$$P(B_1 \mid C_b) = \sum_{k=m}^{b} \binom{b}{k} \rho^k (1-\rho)^{b-k}. \tag{11}$$

Combining inequality (10) and equation (11), we derive the following result:

$$\begin{aligned} P(\text{at least one space is broken}|C_b) &\le \lambda \sum_{k=m}^{b} \binom{b}{k} \rho^k (1-\rho)^{b-k} \\ &= \lambda \sum_{k=m}^{b} \binom{b}{k} \left(\frac{\mu}{\lambda}\right)^k \left(1-\frac{\mu}{\lambda}\right)^{b-k}. \end{aligned} \tag{12}$$

5.2 The Faction of Additional Network Communications Being Compromised

To understand how resilient our proposed scheme is, it is better to investigate the influence caused by the event that an attacker has already captured *b* nodes over the rest of the network. In other words, we like to find out the fraction of additional communications (communications among uncompromised nodes) that an attacker can compromise based on the information obtained from the *b* captured nodes. In order to evaluate this fraction, what we have to do is to compute the probability that any one of the additional secure communication links is compromised after *b* nodes have been captured. Note that the additional secure communication links here are the communication links secured by pairwise keys that are established by two uncompromised neighboring nodes.

Let *s* denote an additional secure communication link, and *pk* denote the pairwise key used for this link. Let H_i denote the joint event that *pk* belongs to key space Ω_i ($pk \in \Omega_i$) and space Ω_i is compromised. Hence, we have:

$$P(s \text{ is broken}|C_b) = P(H_1 \cup H_2 \cup ... \cup H_\lambda \mid C_b).$$

Since *s* can only use one key and events $H_1,...,H_\lambda$ are mutually exclusive and equally likely. Therefore, we have:

$$P(s \text{ is broken}|C_b) = \sum_{k=1}^{\lambda} P(H_i \mid C_b) = \lambda P(H_1 \mid C_b).$$

However, $P(H_1 \mid C_b) = \dfrac{P((pk \in \Omega_1) \cap (\Omega_1 \text{ is compromised}) \cap C_b)}{P(C_b)}$. Because the event ($pk \in \Omega_1$) is independent of the event C_b or the event (Ω_1 is compromised), we have:

$$P(H_1 \mid C_b) = \frac{P(pk \in \Omega_1).P((\Omega_1 \text{ is compromised}) \cap C_b)}{P(C_b)}$$

$$= P(pk \in \Omega_1).P(\Omega_1 \text{ is compromised}|C_b).$$

$P(\Omega_1 \text{ is compromised}|C_b)$ is computed by equation (11). $P(pk \in \Omega_1)$ is the probability that link *s* uses a key from key space Ω_1. Since the choice of a key space from λ key spaces is equally probable, we have: $P(pk \in \Omega_1) = \dfrac{1}{\lambda}$. Therefore,

$$P(s \text{ is broken}|C_b) = \lambda P(H_1 \mid C_b) = \lambda.\frac{1}{\lambda}.P(B_1 \mid C_b) = \sum_{k=m}^{b} \binom{b}{k} \left(\frac{\mu}{\lambda}\right)^k \left(1-\frac{\mu}{\lambda}\right)^{b-k}. \quad (13)$$

The above equation shows that, given that *b* nodes are compromised, the fraction of the additional secure communication links compromised is equal to the probability of one key space being compromised.

5.3 Further Discussion

Our scheme together with other random key pre-distribution schemes [2], [4], [5], [6] is still vulnerable to several kinds of attacks that uniquely occur in random based schemes such as node replication attack [7] and key-swapping collusion attack [8]. To thwart the node replication attack, the method proposed in [9] can be used in cooperation with our scheme. Regarding the key-swapping collusion attack, there has been no radical proposal to prevent it so far. More efforts need to be put into this attack.

In [2], the authors proposed using two-hop neighbors in order to improve security. The same technique can also be applied to our scheme. However, using two-hop neighbors is vulnerable to man-in-the-middle attack if an intermediate node is compromised before or during the pairwise key establishment process. Fortunately, this attack and the mechanism to thwart it have already been presented in [10].

6 Performance Analysis

In this section we evaluate our proposed scheme with respect to memory usage, communication overhead, and computational overhead using Du's scheme as a benchmark.

6.1 Memory Usage

For each key space, according to MI scheme, each node S_i has to spend $m \times l$ bits on storing the value of Φ_i. Thus the total memory usage (KB) MU for each node with μ chosen key spaces is:

$$MU = \frac{m \times l \times \mu}{8 \times 1024} \tag{14}$$

Since the value of m is equal to the value of $\lambda + 1$ in [2]; the value of μ is equal to that of τ in [2]; and $\frac{l}{8 \times 1024}$ is the memory unit of equation (5) in [2], hence memory consumption of our scheme for pairwise key establishment purpose is exactly identical to that of Du's scheme.

6.2 Communication Overhead

In this subsection, we like to compare the communication overhead of our scheme with that of Du's scheme. Note that to establish a pairwise key, the data that each node in our scheme needs to transmit is its effective ID and the indices of the key spaces in it, while in [2] the data needed to transmit is the node's ID, the indices and the seed of the column of G. Based on that, we draw a comparison as shown in fig. 3. From the figure, some observations are straightforwardly drawn: (i) if we choose the value of m such that the inequality $m < 2 \times (\text{length of a pairwise key})$ is assured, then the communication overhead of each node in our scheme is always less than that

in [2]. For example, if $m = 50$ (as chosen in [2]) and pairwise key length is 64 bits, then from the figure, the extra communication overhead for each node in [2] in comparison with our scheme is about 10 bytes. It is well known that transmitting a single bit costs as much power as executing 1000 instructions, then the communication overhead of our scheme is far less than that of [2]. (ii) The extra communication overhead of [2] in comparison with our scheme is directly proportional to the length of the pairwise key. Thus, although increasing in key size means an increase in security level but also in communication overhead.

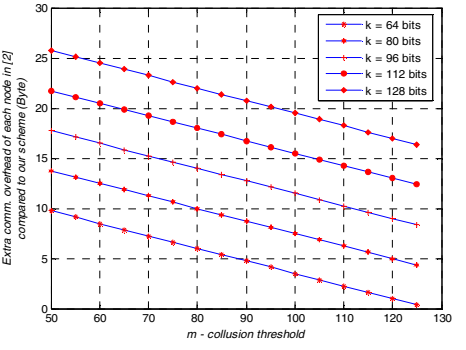

Fig. 3. Extra communication overhead of each node in [2] compared to our scheme

6.3 Computational Overhead

In this subsection, we compare the computational overhead of our scheme with that in [2]. In our scheme, to compute a pairwise key, each node needs to perform a multiplication of a $(l \times m)$ matrix and an $(m \times 1)$ effective ID. Therefore, each node needs $l \times m$ single-precision multiplications while each node in [2] needs to do $2 \times (m-1) \times l^2$ single-precision multiplications. It follows that the computational overhead of our scheme is far less than that in [2]. The numbers in fig. 4 reinforce our argument.

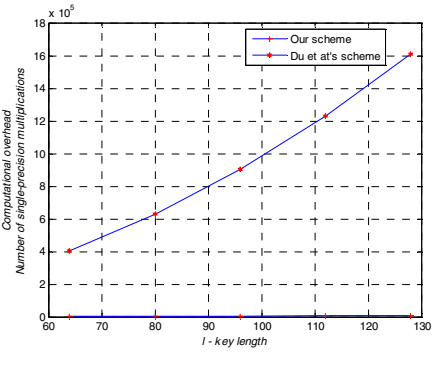

Fig. 4. Computational overhead in each node with various key lengths

7 Conclusions and Future Work

This paper proposes a new key pre-distribution scheme for WSNs that can be considered as a refinement of two types of schemes: ID-based key pre-distribution scheme and random key pre-distribution scheme. As a result, our scheme possesses a number of attractive properties. First, our scheme is scalable and flexible in terms of network size. Second, our scheme substantially improves network resiliency against node capture attack compared to schemes [4], [5], [11], and are comparable to Du's scheme. Furthermore, we have argued that network resiliency can be further improved using a combination of multi-hop neighbors method and a method to thwart man-in-the-middle attack that we proposed in [10]. We have also investigated the performance of our scheme to show its efficiency. Accordingly, our scheme is the same as Du's scheme in terms of memory usage. Moreover, under a certain condition, our scheme is more efficient than Du's scheme concerning communication overhead. Finally, computational overhead of our scheme is argued to be far less than that of Du's scheme.

In the preceding discussion, we have shown that our scheme is still vulnerable to node replication attack and key-swapping collusion attack. Therefore, in our future work, we would like to explore additional mechanisms to efficiently and radically thwart these attacks.

References

1. T. Matsumoto and H. Imai, "On the KEY PREDISTRIBUTION SYSTEM: A Practical Solution to the Key Distribution Problem", CRYPTO'87, LNCS Vol. 293, pp.185-193, Aug. 1987
2. W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, and A. Khalili, "A pairwise key predistribution scheme for wireless sensor networks", ACM Trans. Info. Sys. Sec., Vol. 8, No. 2, pp.228-258, May 2005
3. A. Shamir, "How to share a secret", Communications of the ACM, Vol. 22, No. 11, pp.612-613, Nov. 1979
4. L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks", Proc. of the 9th ACM Conference on Computer and Communications Security, pp.41-47, Nov. 2002
5. H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks", Proc. IEEE Symposium on Security and Privacy, pp.197-213, May 2003
6. D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks", Proc. of the 10th ACM Conference on Computer and Communications Security (CCS'03), pp.52-61, Oct. 2003
7. H. Fu, S. Kawamura, M. Zhang, and L. Zhang, "Replication attack on random key pre-distribution schemes for wireless sensor networks", Proc. IEEE on SMC Information Assurance Workshop, pp.134-141, Jun. 2005
8. T. Moore, "A collusion attack on pairwise key predistribution schemes for distributed sensor networks", Proc. IEEE on Pervasive Computing and Communications Workshops (PERCOMW'06), Mar. 2006
9. B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks", Proc. of IEEE Symposium on Security and Privacy, pp.49-63, May 2005

10. T. T. Dai, C. T. Hieu, Md. M. Rahman, and C. S. Hong, "A Robust Pairwise Key Predistribution Scheme Resistant to Common Attacks for Wireless Sensor Networks", Proc. of 7th WISA 2006, pp.121-135, Jeju Island, Korea, Aug. 2006.
11. T. T. Dai, C. T. Hieu, and C. S. Hong, "An Efficient ID-based Bilinear Key Predistribution Scheme for Distributed Sensor Networks", LNCS 4208, pp.260-269, Sep. 2006.
12. R. Blom, “An optimal class of symmetric key generation systems”, EUROCRYPT '84, LNCS Vol. 209, pp.335-338, 1985
13. S. P. Chan, R. Poovendran, and M. T. Sun, “A key management scheme in distributed sensor networks using attack probabilities”, Proc. IEEE GLOBECOM 2005, pp.1007-1011, 2005
14. W. Stallings, “Cryptography and Network Security: Principles and Practice”, Prentice Hall, 2nd edn, Jul. 1998
15. A. Menezes, P. Van Oorschot, and S. Vanstone, “Handbook of Applied Cryptography”, CRC Press, Inc., 1996