A Self-healing Key Distribution Scheme Based on Vector Space Secret Sharing and One Way Hash Chains

Biming Tian, Song Han, Tharam S. Dillon
DEBI Institute
Curtin University of Technology, Australia
{Biming.Tian, Song.Han, Tharam.Dillon }
@cbs.curtin.edu.au

Sajal Das
Department of Computer Science and Engineering
University Texas at Arlington, USA
das@cse.uta.edu

Abstract

An efficient self-healing key distribution scheme with revocation capability is proposed for secure group communication in wireless networks. The scheme bases on vector space secret sharing and one way hash function techniques. Vector space secret sharing helps to realize general monotone decreasing structures for the family of subsets of users that can be revoked instead of a threshold one. One way hash chains contribute to reduce communication overhead. Furthermore, the most prominent characteristic of our scheme is resisting collusion between the new joined users and the revoked users, which is fatal weakness of hash function based self-healing key distribution schemes.

1. Introduction

The idea of self-healing key distribution schemes is that users, in a large and dynamic group communication over an unreliable network, can recover lost session keys on their own, even if lost some previous key distribution messages, without requesting additional transmissions from the group manager. This new approach to key distribution is very useful due to the self-healing property, supporting secure communication wireless networks.

Self-healing key distribution appears to be quite useful in several settings in which session keys need to be used for a short time-period, due to frequent changes in the group structure. Military-oriented applications as well as Internet application, such as broadcast transmissions, pay-per-view TV, are few important examples which can benefit from such approaches. In addition, the self-healing method may be useful in commercial content distribution applications or electronic services in which the contents are highly sensitive.

In this paper, we propose an efficient self-healing key distribution scheme for secure group communication in wireless sensor networks. We use vector space secret sharing technique to realize general monotone decreasing structures for the family of subsets of users that can be revoked instead of a threshold one and dual directional hash chains to reduce communication overhead. Furthermore, random numbers are used in the process of computing session keys. The scheme will not only keep the forward and backward secrecy but also resist collusion between the new joined users and the revoked users. As far as we know, this is first time to solve the collusion in hash function based self-healing key distribution schemes.

The rest of the paper is organized as follows: In Section 2, we present an overview of earlier works in the area of self-healing key distribution. In Section 3, we briefly introduce preliminaries which help to understand the construction of self-healing key distribution scheme. In Section 4, we introduce our system parameters firstly. Security model and concrete construction are presented secondly. In Section 5, we provide a proof of security of our proposed scheme. In Section 6, we focus on performance analysis. We conclude the paper in the last section.

2. Related Works

The first pioneering work of self-healing key distribution was introduced by Staddon et al. in [1]. Formal definitions, lower bounds on the resources as well as some constructions of self-healing key distribution scheme were proposed in it. Liu et al. generalized the definitions in [1] and gave some constructions in [2]. The scheme reduced communication overhead and storage overhead by introducing a novel personal key distribution technique. Blundo et al. in [3] showed an attack that can be applied to the first construction in [1], presented a new mechanism for implementing the self-healing approach, extended the self-healing approach to key distribution, and proposed another key recov-

978–1–4244–2100–8/08/$25.00

ery scheme which enabled a user to recover all lost session keys (for sessions which he was entitled to) by using only the current broadcast message. The constructions in [1] - [3] are all based on Shamir's secret sharing technique.

More et al. in [4] used a sliding window to address the three problems in [1]. The three problems were inconsistent robustness, high overhead and expensive maintenance costs. Sáez first considered applying vector space secret sharing instead of Shamir's secret sharing schemes to realize self-healing key distribution scheme [5] and sponsorization [13]. All of these papers [1] - [5] and [13] focus on unconditionally secure schemes.

By introducing an improved secret sharing scheme, Tian et al. proposed a self-healing key scheme with novel properties in [6]. Firstly, the scheme reduced storage overhead of personal key to a constant. Secondly, the scheme conceals the requirement of secure channel in setup phase. In addition, the long-lived scheme was much more efficient than those in [1] and [3]. However, the efficiency improvements are obtained by relaxing the security slightly. The scheme is a computationally secure scheme.

Jiang et al proposed an efficient self-healing group key scheme with time-limited node revocation based on Dual Directional Hash Chains (DDHC) in [7]. Similarly, the constructions in [10] - [12] are all based on hash chains. Both improved properties and forward and backward security are achieved. However, there is fatal defect in hash-based constructions. The collusion between new joined users and revoked users will recover all the session keys which they are not entitled to.

To sum up, Shamir's secret sharing is the most common technique used to realize self-healing key distribution. It is performs easily. However, the maximum number of revoked users is constraint to the degree of the polynomial. Vector space secret sharing based self-healing key distribution schemes consider a monotone decreasing family of rejected subset of users instead of a monotone decreasing threshold structure. This general case makes the self-healing scheme more flexible and close to practical application. Both forward and backward secrecy can be assured by dual directional hash chains. However, how to resist collusion between revoked nodes and new joined nodes is an open problem, due to the properties of one-way hash function.

In this paper, we will propose an efficient self-healing key distribution scheme for secure group communication in wireless sensor networks. Firstly, we consider general structures instead of threshold ones to provide more flexible performance to the scheme. Secondly, we devote to solve the collusion in hash function based self-healing key distribution schemes.

3. Preliminaries

3.1. Access Structure

Secret sharing schemes play an important role in distributed cryptography. In these schemes, a secret value is shared among a set $U = \{1, \ldots, n\}$ of n players in such a way that only qualified subsets of players can reconstruct the secret from their shares. The family of qualified subsets is the access structure, denoted by Γ. This family $\Gamma \subset 2^U$ must be monotone increasing, that is, if $A_1 \in \Gamma$ and $A_1 \subset A_2 \subset U$, then $A_2 \subset U$. The family of authorized subsets Γ is determined by the collection of minimal authorized subsets Γ_0 called the basis of the structure. The family of non-authorized subsets $\bar{\Gamma} = 2^U - \Gamma$ is monotone decreasing. A structure R is monotone decreasing when $A_1 \in R$ and $A_2 \subset A_1$ imply $A_2 \in R$. A monotone decreasing structure R is determined by the collection of maximal subsets R_0.

3.2 Vector Space Secret Sharing

The vector space secret sharing scheme was introduced by Brickell [8]. Let us suppose that the dealer is D and that there is a public map

$$\psi : U \cup \{D\} \rightarrow GF(q)^l$$

where q is a prime power and l is a positive integer. This map induces the monotone increasing access structure Γ defined as follows: $A \in \Gamma$ if and only if the vector $\psi(D)$ can be expressed as a linear combination of the vectors in the set $\psi(A) = \{\psi(i) | i \in A\}$. An access structure Γ is said to be a vector space access structure if it can be defined in the above way. If Γ is a vector space access structure, we can construct a secret sharing scheme for Γ with set of secrets $GF(q)$(see [8] for a proof). To distribute a secret value $k \in GF(q)$, the dealer takes at random an element $v \in GF(q)^l$, such that $k = v \cdot \psi(D)$. The share of a participant $i \in U$ is $s_i = v \cdot \psi(i)$. Let A be an authorized subset $A \in \Gamma$; then, $\psi(D) = \sum_{i \in A} \lambda_i \cdot \psi(i)$ for some $\lambda_i = GF(q)$. In order to recover the secret k, players in A compute

$$\sum_{i \in A} \lambda_i s_i = \sum_{i \in A} \lambda_i v \cdot \psi(i) = v \cdot \sum_{i \in A} \lambda_i \psi(i) = v \cdot \psi(D) = k$$

A scheme constructed in this way is called a vector space secret sharing scheme.

3.3 One-way Hash Chain

Hash function is the foundation of hash chain. A hash function H takes a binary string M of arbitrary length as input, and outputs a binary string of fixed length, which is

called hash value h: $h = H(M)$. A one-way function H satisfies the following three properties [9]:

1. Given an input M, it is easy to compute h such that $h = H(M)$;
2. Given a hash value h, it is computationally infeasible to find a second input M such that $H(M) = h$;
3. Given a hash value h, it is computationally infeasible to find a second input M' such that $H(M') = h$, where $M' \neq M$.

One way hash chains in this paper are composed of two one-way hash chains with equal length, a forward hash chain K^F and a backward hash chain K^B. It can be derived as follows: (1) generating two random key seed values, K_0^F and K_0^B, for forward and backward hash chains, respectively; (2) repeatedly applying the same one-way function on each seed to produce two hash chains of equal length m. The hash sequences are generated as:

$$\{H(K_0^F), \ldots, H^i(K_0^F), \ldots, H^m(K_0^F)\}$$

$$\{H(K_0^B), \ldots, H^i(K_0^B), \ldots, H^m(K_0^B)\}$$

4 Security Model and Concrete Construction

This section briefly defines security model for self-healing key distribution scheme. The notations in Table 1 are used throughout the paper.

Table 1. Notations

U	set of all users in the networks
U_i	i-th user
n	total number of users in the network
m	total number of sessions
$GF(q)$	a field of order q
S_i	personal secret of user U_i
SK_j	session key in session j
B_j	broadcast message for session j
R_j	the set of revoked users in session j
J_j	the set of joined users in session j
S_F	forward key seed
S_B	backward key seed
K_i^F	i-th forward key
K_i^B	i-th backward key

4.1 Our Security Model

To further clarify our goals and facilitate the later presentation, according but not constraint to the security model of , we define the self-healing key distribution scheme as follows:

To further clarify our goals and facilitate the later presentation, according but not constraint to the security model of [10], we define the self-healing key distribution scheme from three aspects. *Definition 4.1* defines self-healing key distribution scheme with revocation capability. *Definition 4.2* defines forward secrecy and backward secrecy. *Definition 4.3* defined resisting collusion properties.

Definition 4.1 is the same as Definition 2.1 and in [10]. We omit the description of it here due to space limitation.

Definition 4.2. Let $U = \{U_1, \ldots, U_n\}$ and $j \in \{1, \ldots, m\}$. The scheme guarantees both forward secrecy and backward secrecy if

1. for any set $R \subseteq U$, and all are revoked before session j, it is computationally infeasible for the members in R together to get any information about SK_j, even with the knowledge of group keys $SK_1, \ldots, SK_{j-1}$ before session j.
2. for any set $J \subseteq U$, and all $U_l \in U$ join after session j, it is computationally infeasible for the members in J together to get any information about SK_j, even with the knowledge of group keys $SK_{j+1}, \ldots, SK_m$ after session j.

Definition 4.3. Let $B \subset R_r \cup \ldots \cup R_2$ be a coalition of users removed from the group before session r and let $C \subset J_s \cup \ldots \cup J_m$ be a coalition of users who join the group from session s with $r < s$. Suppose $B \cup C \subset \Re$.

The scheme is able to resist collusion. That is, the coalition $B \cup C$ doses not get any information about session keys SK_j, for any $r \leq j < s$.

4.2 Our Construction

Following the idea of scheme in [5] and [10], we present a construction for self-healing key distribution scheme. Firstly, we consider general structures instead of threshold ones to provide more flexible performance to the scheme. Secondly, we devote to solve the collusion in hash function based self-healing key distribution schemes.

In our setting, communication group is a dynamic subset of users of U. A broadcast unreliable channel is available, and time is defined by a global clock. GM sets up and manages, by means of joining and revoking operations, a communication group. We denote the set of users in session j as $G_j = (G_{j-1} \cup J_j) - R_j$ for $j \geq 2$. By definition, there is $G_1 = U$. Each user $U_i \in G_j$ holds a personal key S_i, which is received from GM before or when joining G_j. The personal key S_i can be seen as a sequence of elements from a finite set. For $U_i \in G_j$ and $j = 1, \ldots, m$, the session key SK_j can be determined by S_i and B_j. All of our operations take place in $GF(q)$, where q is a large prime number

$(q > n)$. Let $H : GF(q) \rightarrow GF(q)$ be a cryptographically secure one-way hash function.

Let $\Re \subset 2^U$ be a monotone decreasing access structure of subsets of users that can be revoked by GM and let $\Gamma = 2^U - \Re$ be a monotone increasing access structure. Let us consider a vector space secret sharing scheme realizing Γ over the set U. For simplicity, we suppose that there exists a public map

$$\psi : U \cup \{D\} \rightarrow GF(q)^l$$

which defines Γ as a vector space access structure. The use of a specific ψ fixes the properties of the scheme.

The self-healing key distribution scheme is composed of five procedures. Now we describe the different procedures one by one.

Setup. GM uses the Pseudo Random Number Generator (PRNG) of large enough period to period to produce a sequence of m random numbers $r_1, \ldots, r_m$. GM randomly picks two initial key seeds, S^F and S^B, from $GF(q)$. In the pre-processing time, GM computes two hash chains of equal length m by repeatedly applying the same one-way hash function H on each seed $K_0^F = S^F$ and $K_0^B = S^B$. For $1 \leq i \leq m$, the hash sequences are generated as follows:

$$\{K_0^F = S^F, H(K_0^F), \ldots, H^i(K_0^F), \ldots, H^m(K_0^F)\}$$

$$\{K_0^B = S^B, H(K_0^B), \ldots, H^i(K_0^B), \ldots, H^m(K_0^B)\}$$

Group key for session j is computed as:

$$SK_j = K_j^F + (K_{m-j+1}^B \oplus r_j)$$

GM chooses random vectors $u_1, \ldots, u_m \in GF(q)^l$. For each session $j = 1, \ldots, m$, GM computes the scalar $z_j = K_{m-j+1}^B + \psi(D)^\top u_j \in GF(q)$. Each user U_i is first assigned a prearranged life cycle (s, t) where $1 \leq s < t \leq m$. U_i will be involved in $k = t - s + 1$ number sessions. The user U_i receives its private key from GM consisting of: (1) a forward key K_s^F for session s; (2) k number of random numbers corresponding to the sessions which the user U_i will participate in; (3) secret vectors. GM sends the private key to user U_i via the secure channel between them, as shown below:

$$(K_s^F || r_s, \ldots, r_t || (\psi(i)^\top u_1, \ldots, \psi(i)^\top u_m))$$

Broadcast. Suppose $R_j \subseteq G_{j-1}$ with $R_1 \cup \ldots \cup R_j \in \Re$ for session . By definition we have $R_1 = \phi$. GM chooses a maximal non-authorized subset of users $W_j \in \Re_0 = \bar{\Gamma_0}$ such that $R_1 \cup \ldots \cup R_j \subset W_j$ and with minimum cardinality. The broadcast $B_j = B_j^1 \cup B_j^2$ in session $j = 1, \ldots, m$ is given as follows:

$$B_j^1 = z_j$$

$$B_j^2 = \begin{cases} \{(k, \psi(k)^\top u_j)\}_{k \in W_j} & j = 1, 2 \\ B_{j-1}^2 \cup \{(k, \psi(k)^\top u_j)\}_{k \in W_j} & j \geq 3 \end{cases}$$

Key Recovery. When a non-revoked user U_i received the key distribution message B_j for session j, since $U_i \in G_j$ has $\{(k, \psi(k)^\top u_j)\}_{k \in W_j}$ and his personal key, he computes $\psi(D)u_j$ using $\{(k, \psi(k)^\top u_j)\}_{k \in W_j \cup \{i\}}$ because $W_j \cup \{i\} \in \Gamma$. In effect: as far as $W_j \cup \{i\} \in \Gamma$, the result $\psi(D) = \sum_{k \in W_j \cup \{i\}} \lambda_k \psi(k)$ holds for some $\lambda_k \in GF(q)$. Therefore, $\psi(D)u_j = \sum_{k \in W_j \cup \{i\}} \lambda_k \psi(k) u_j$. From the broadcast information B_j, then U_i recovers the backward key

$$K_{m-j+1}^B = z_j - \psi(D)^\top u_j.$$

At last, U_i computes the j-th forward key $K_j^F = H^{j-1}(S^F)$ and evaluates the current session key

$$SK_j = K_j^F + (K_{m-j+1}^B \oplus r_j).$$

Add Group Members. When a new user wants to join the communication group, the user gets in touch with GM. GM assigns a life circle $(s, t)$and an unused identity $v \in GF(q)$ to the new user. GM computes the personal secret key $S_v = (K_s^F || r_s, \ldots, r_t || \psi(v)^\top u_s, \ldots, \psi(v)^\top u_t))$ to this new user via the secure communication channel between them.

Self-healing. Suppose U_i is a user who receives session broadcast message B_{j_1} and B_{j_2} in session j_1 and j_2 respectively, where $1 \leq j_1 < j_2 \leq m$, but not broadcast message B_j for the session j, where $j_1 < j < j_2$. U_i can still recover all the lost session keys $K_j (j_1 < j < j_2)$ as follows:

(a) U_i recovers from the broadcast message B_{j_2} in session j_2, the backward key K_{m-j+1}^B and repeatedly apply the one-way hash function on this and computes the backward keys for all $j (j_1 < j < j_2)$.

(b) U_i computes the forward keys K_j^F for all $j (j_1 \leq j \leq j_2)$by repeatedly applying the one-way hash function H on the forward key $K_{j_1}^F$ of the session j_1.

(c) U_i recovers all the session keys $SK_j = K_j^F + (K_{m-j+1}^B \oplus r_j)$, for $j_1 < j < j_2$.

5 Security Analysis

In this section we show that our Construction realizes self-healing key distribution scheme with revocation ability. More precisely, we can prove our construction in our security model described in Section 4.1

We will now show that our construction satisfies all the conditions required by *Definition 4.1*

1. The scheme is a session key distribution scheme.

   (a) A non-revoked user $U_i \in G_j$ can recover the session key SK_j efficiently form from broadcast message B_j and personal key S_i. As can be seen in procedure *Key Recovery*.

(b) The session key SK_j for session j is computed from three parts: forward key K_j^F, backward key K_{m-j+1}^B, and random number r_j, where $K_j^F = H^{j-1}(S^F)$, $K_{m-j+1}^B = z_j - \psi(D)^\top u_j$ and r_j is part of personal key received from GM before or joining the session group. So the personal secret keys alone do not give any information about any session key. Since backward seed S^B is chosen randomly, the backward key K_{m-j+1}^B and consequently the session key SK_j is random as long as forward seed S_F and backward key sequence are not get revealed. Furthermore, the broadcast $B_1, \ldots, B_m$ determine $z_1, \ldots, z_m$but the scalars perfectly hide the backward key sequence by mean of $\psi(D)^\top u_j$ because $z_j = K_{m-j+1}^B + \psi(D)^\top u_j$. So it is computationally infeasible to determine session key SK_j from broadcast message B_j or personal key S_i alone.

2. The scheme has $\Re$-revocation capability. For each session j, let $R = R_j \cup \ldots \cup R_2 (R \in \Re)$ be a collection of revoked users collude. It is infeasible for the coalition R to compute the j-th session key SK_j because knowledge of SK_j implies the knowledge of backward key K_{m-j+1}^B or disclosing of personal secret $\psi(i)^\top u_j$ of $U_i \notin R$. A user $U_i \in R$ knows, from the broadcast B_j, vector $\{(k, \psi(k)^\top u_j)\}_{k \in W_j}$ and his personal key $\psi(i)^\top u_j \in GF(q)$ where $i \in W_j \notin \Gamma$, and this does get any information on $\psi(D)^\top u_j$. We get the conclusion because for any scalars $s \in GF(q)$ there exists at least one vector $u \in GF(q)^l$ such that

$$\begin{cases} \psi(D)^\top u = s \\ \psi(k)^\top u = \psi(k)^\top u_j \quad for \quad any \quad k \in W_j \end{cases}$$

because $W_j \notin \Gamma$. Notice that the number of vectors u satisfying this system of equations is independent of the value s.

3. The scheme has self-healing capability. As can be seen from procedure of *Self-healing*.

We will show our construction satisfies forward secrecy and backward secrecy required by *Definition 4.2*.

1. Only those sets belong to Γ can recover the $\psi(D)$and further recovery the session key SK_j. Because of the fact that $R \notin \Gamma$, the collation of R cannot recover $\psi(D)$. Furthermore, because of the one-way property of H, it is computationally infeasible to compute K_s^B from K_t^B for $s < t$. That is, even the users might know the sequence of backward keys $K_m^B, \ldots, K_{m-j+2}^B$, still cannot compute backward key K_{m-j+1}^B for session j and consequently SK_j from the sequence. In addition, the random number strengthens the security of our scheme. Therefore, our construction guarantees the forward secure.

2. The correctness bases on the one-way property of hash function. In order to know $SK_s (s < j)$, $U_l \in J$ requires the knowledge of s-th forward key $K_s^F = H(K_{s-1}^F) = H^{s-1}(S^F)$. However, when a new user U_v joins the group starting from session j, GM gives j-th forward key K_j^F instead of the initial forward key seed S^F, together with the vectors and numbers. Therefore, it is computationally infeasible for the newly joint users to trace back for previous forward key K_s^F for $s < j$ due to $K_j^F = H(K_{j-1}^F)$. In addition, the random number strengthens not only forward secure but also backward secure. Hence we can claim that our scheme keeps backward secure. In fact, the backward secrecy is independent of secret vectors.

We will show our construction satisfies forward secrecy and backward secrecy required by *Definition 4.3*.

The scheme has resisting collusion capability. Let $B \subset R_s \cup R_{s-1} \cup \ldots \cup R_2$ be a coalition of users removed from the group before session s and let $C \subset J_t \cup J_{t+1} \cup \ldots \cup J_m$ be a coalition of users who join the group from session t with $s < t$ such that $B \cup C \notin \Gamma$. Secret information held by users in $B \cup C$ and broadcasts in all the sessions do not get any information about SK_j for sessions $j = s, \ldots, t-1$. This is true because they know, in the worst case, $s_i = (\psi(i)^\top u_t, \ldots, \psi(i)^\top u_m) \in GF(q)^{m-t+1}$ for $U_i \in C$ and $B_1, \ldots, B_m$. Since $B \subset W_j$ for any $j = s, \ldots, t-1$ and personal keys of users in B are public for sessions $j \geq t$, it easy to see that all the possible value for SK_j with $j = s, \ldots, t-1$ have the same probability.

6 Performance Analysis

Table 1 in [10] demonstrated that the constructions in [10] are more efficient than previous schemes in terms of storage, communication, and computation overhead. In this section we make a performance comparison between constructions in [10] and our construction. There are two differences between them. Firstly, we consider general structures instead of threshold ones to provide more flexible performance to the scheme. Secondly, we endow our scheme with resisting collusion property by using random number subtly.

In terms of storage overhead, our scheme requires for each user is $(2m+1)logq$. It comes from forward seed, random numbers and vectors corresponding to sessions. For those users who join later or only involve in part of sessions need to less data. A user who joins in j-th session and will involve in k sessions needs store $[1+k+(m-j+1)]logq = (m-j+k+2)logq$ bits personal keys. More storage overhead than Constructions in [10] comes from random number. However, it is used to resist collusion.

The broadcast message B_j at the j-th session consists of

a set of revoked users and their vectors corresponding to session j. One can ignore the communication overhead for the broadcast message of the set W_j, because the user identities can be selected from a small finite field [2]. Thus the size of the communication overhead of Constructions in [10] in session j is $(t+1)logq$. In our scheme, the broadcast length depends on the particular function ψ. Its length depends on the history of rejected subsets. If there are t revoked users, both our Construction and the Constructions in [10] have the same broadcast length. If there are less revoked users, our construction has lower communication overhead than the constructions in [10]. Furthermore, the constructions in [10] limit the revoked number to a threshold . If more than t users were revoked, the security of constructions can not be guaranteed. Our construction conceals the limitation thus more practical.

7 Conclusions

We developed a computationally secure self-healing key distribution scheme which has more efficient and secure than previous similar schemes. Firstly, we consider general structures instead of threshold ones to provide more flexible performance to the scheme. The scheme realizes general monotone decreasing structures for the family of subsets that can be revoked instead of a threshold in polynomial-based schemes. Secondly, we devote to solve the collusion in hash function based self-healing key distribution schemes. We continue use dual directional hash chains to reduce communication overhead. Different from previous schemes, we add XOR random numbers operation to the process of computing session keys. The scheme not only keeps the forward and backward secrecy but also resists collusion between the new joined users and the revoked users. As far as we know, this is first time to solve the collusion in hash function based self-healing key distribution schemes. The scheme is analyzed in an appropriate security model to prove that they are computationally and can be used for secure communication over an unreliable wireless networks.

References

[1] J. Staddon, S. Miner, M. Franklin, D. Balfanz, M. Malkin and D. Dean, Self-healing Key Distribution with Revocation, *proceedings of IEEE Symposium on Security and Privacy*, pp. 224-240, 2002.

[2] D. Liu, P. Ning and K. Sun, Efficient Self-healing Group Key Distribution with Revocation Capability, *Proceeding of the 10th ACM CCS*, 2003.

[3] C. Blundo, P. D'Arco, A. Santis and M. Listo, Design of Self-healing Key Distribution Schemes, *Design Codes and Cryptography*, No.32, pp.15-44, 2004.

[4] S. M. More, M. Malkin, J. Staddon and D. Balfanz, Sliding Window Self-healing Key Distribution with Revocation, *ACM Workshop on Survivable and Self-Regenerative Systems*, pp.82-90, 2003.

[5] G. Sáez, On Threshold Self-healing Key Distribution Schemes, *Cryptography and Coding*, LNCS, Vol.3796, pp.340-354, 2005.

[6] B. Tian and M. He, Self-Healing Key Distribution Scheme with Novel Properties, *International Journal of Network Security*, Vol.7, No.2, pp.147-152, 2008.

[7] Y. Jiang, C. Lin, M. Shi and X. Shen, Self-healing Group Key Distribution with Time-limited Node Revocation for Wireless Sensor Networks, *Ad Hoc Networks*, Vol.5, No.1pp.14-232007.

[8] E. F. Brickell, Some Ideal Secret Sharing Schemes, *Journal of Combinatorial Mathematics and Combinatorial Computing*, 9, pp.105-113, 1989.

[9] NIST: Secure hash standard, National Institute for Standards and Technology, Gaithersburg, MD, USA, April 1995.

[10] R. Dutta, E. C. Chang, S. Mukhopadhyay, Efficient Self-healing Key Distribution with Revocation for Wireless Sensor Networks Using One Way Key Chains, *ACNS 2007*, LNCS, Vol.4521, pp.385-400, 2007.

[11] F. Kausar, S. Hussain, J. H. Park, and A. Masood, Secure Group Communication with Self-healing and Rekeying in Wireless Sensor Networks, *Proceedings of the Third International Conference, MSN2007*, pp.737-748, 2007.

[12] M. J. Bohio and A. Miri, Self-healing Group Key Distribution, *International Journal of Network Security*, Vol.1, No.2, pp.110-117, Sep 2005.

[13] G. Sáez, Self-healing Key Distribution Schemes with Sponsorization, *IFIP International Federation for Information Processing'05*, LNCS, Vol.3677, pp.22-31, 2005.