sensors

MDPI

Article

Exponential Arithmetic Based Self-Healing Group Key Distribution Scheme with Backward Secrecy under the Resource-Constrained Wireless Networks

Hua Guo [1,*,†,‡], Yandong Zheng [1,‡], Xiyong Zhang [2,‡] and Zhoujun Li [1,‡]

[1] State Key Laboratory of Software Development Environment, Beihang University, Beijing 100000, China; zy1406423@buaa.edu.cn (Y.Z.); lizj@buaa.edu.cn (Z.L.)

[2] State Key Lab of Mathematical Engineering and Advanced Computing, Wuxi 214000, China; xiyong.zhang@hotmail.com

* Correspondence: hguo@buaa.edu.cn; Tel.: +86-10-8233-8247

† Current address: State Key Laboratory of Software Development Environment, Beihang University, Beijing 100000, China.

‡ These authors contributed equally to this work.

Academic Editor: Rongxing Lu

Received: 2 March 2016; Accepted: 21 April 2016; Published: 28 April 2016

Abstract: In resource-constrained wireless networks, resources such as storage space and communication bandwidth are limited. To guarantee secure communication in resource-constrained wireless networks, group keys should be distributed to users. The self-healing group key distribution (SGKD) scheme is a promising cryptographic tool, which can be used to distribute and update the group key for the secure group communication over unreliable wireless networks. Among all known SGKD schemes, exponential arithmetic based SGKD (E-SGKD) schemes reduce the storage overhead to constant, thus is suitable for the the resource-constrained wireless networks. In this paper, we provide a new mechanism to achieve E-SGKD schemes with backward secrecy. We first propose a basic E-SGKD scheme based on a known polynomial-based SGKD, where it has optimal storage overhead while having no backward secrecy. To obtain the backward secrecy and reduce the communication overhead, we introduce a novel approach for message broadcasting and self-healing. Compared with other E-SGKD schemes, our new E-SGKD scheme has the optimal storage overhead, high communication efficiency and satisfactory security. The simulation results in Zigbee-based networks show that the proposed scheme is suitable for the resource-restrained wireless networks. Finally, we show the application of our proposed scheme.

Keywords: wireless networks; self-healing group key distribution; exponential arithmetic; backward secrecy

PACS: J0101

1. Introduction

Wireless sensor networks have drawn a lot of attention because they have demonstrated applicability in practical applications, such as emergency rescue operations and military application. In these applications, the security of the wireless sensor networks should be highly regarded. In wireless sensor networks, resources, including storage and communication bandwidth, are constrained since nodes are powered by battery. To guarantee the secure communication in wireless networks, secure group keys should be distributed to nodes for the purpose of encryption and authentication.

As an issue in wireless networks, packet losses are inevitable and have a negative impact on the group key distribution flows. The packets may never arrive for some target nodes. The most

direct way is to request retransmission. However, retransmitting data packets consumes additional communication resources, which increases the burden on the group manager (GM) in the large communication group. In addition, nodes could expose their current locations, which might be considered as privacy in some applications. Designing a secure and efficient group key distribution protocol for the resource-constrained wireless networks is a challenging task.

A self-healing mechanism, introduced by Staddon [1], can solve the above problem for the unreliable networks. The core idea of a self-healing mechanism is that, GM adds some redundant messages to the broadcast messages so that the group members have the capability to recover the missing session keys without requiring the GM to retransmit the missing messages. After Staddon's first self-healing group key distribution (SGKD) scheme, many SGKD schemes were proposed. Up to now, four kinds of SGKD schemes exist: polynomial based SGKD (P-SGKD) schemes [2–6], vector space secret sharing based SGKD schemes [7–10], bilinear pairings based SGKD schemes [11–13] and exponential arithmetic based SGKD (E-SGKD) schemes [1,2,14,15]. Among them, vector space secret sharing based SGKD schemes and the bilinear pairings based SGKD schemes are inefficient compared with the other two in terms of the storage overhead and the communication cost. In addition, some other self-healing mechanisms are proposed such as mutual-healing [16] and full-healing [17].

Related Works. Staddon *et al.* [1] first proposed a P-SGKD scheme, and transformed it to an E-SGKD scheme. Blundo *et al.* [2] attacks construction 1 in [1] and proposed a novel self-healing mechanism. For the purpose of reducing the communication overhead and storage overhead, Liu *et al.* [18] proposed a new SGKD scheme and proposed two constructions which allow the trade-off between the capability of self-healing and the size of broadcast messages. More *et al.* [19] proposed a novel SGKD scheme, using a sliding window to make the communication overhead and the self-healing capability more balanced. Later, some SGKD schemes were designed [20,21].

Liu *et al.* [18] introduced the first revocation polynomial based SGKD (RP-SGKD) scheme with low storage and communication cost. Hong *et al.* [22] simplified the scheme in [18], which has lower communication overhead. After that, to reduce the communication overhead of the known RP-SGKD schemes, some new RP-SGKD schemes combined with hash chains have been proposed [23–26]. Unfortunately, these RP-SGKD schemes with hash chains are not resistant to the collusion attack. Recently, Chen *et al.* proposed an efficient RP-SGKD schemes [6] which has constant-size storage cost. Unfortunately, Guo *et al.* [27] found that their scheme is insecure. Zou *et al.* [28] proposed the first access polynomial based SGKD (AP-SGKD) scheme. After that, some AP-SGKD schemes are proposed [29–34], which can guarantee the group nodes' identity privacy and reduce the storage overhead to the constant. More recently, Sun *et al.* [35] proposed an AP-SGKD scheme. However, it is easy to find that almost all of the AP-SGKD schemes are insecure [35,36].

Some vector space secret sharing based SGKD schemes [7–10] are proposed. These schemes are more flexible, since any particular access structure will not be imposed. However, the access structure requires being chosen in advance so that the maximum number of the revoked users is determined. The shortcoming of these SGKD schemes is that the revoked users should be predetermined. Some bilinear pairings based SGKD schemes [11–13] are proposed, which can reduce the storage overhead to constant and resist collusion attack of any revoked users. However, the communication overhead is prohibitively large.

The E-SGKD scheme is an extension of the polynomial based SGKD scheme. Rams *et al.* [21] claimed that almost all of the polynomial-based SGKD schemes can be converted to the E-SGKD schemes. Up to now, there are only four published E-SGKD schemes, *i.e.*, Construction 5 in [1] and Scheme 4 in [2], the scheme in [14,15]. Staddon *et al.* proposed the first E-SGKD scheme (see Construction 5 [1]) based on bivariate polynomial and Lagrange Interpolation. Later, Blundo *et al.* simplified Staddon *et al.*'s scheme and proposed an E-SGKD scheme (see Scheme 4 [2]) based on univariate polynomial and Lagrange Interpolation with lower communication overhead. However, both construction 5 [1] and Scheme 4 [2] do not have backward secrecy and the size of the broadcast is

too large. Rams *et al.* [21] pointed out that all known E-SGKD schemes can not offer backward secrecy. In order to solve the backward secrecy and reduce the size of the broadcast message, Rams *et al.* [14] proposed an efficient E-SGKD scheme based on Lagrange Interpolation and sliding windows with backward secrecy. Then, Rams *et al.* [15] improved the scheme [14] and proposed an E-SGKD scheme with lower storage overhead. The sliding windows allow the trade-off between the size of the broadcast message and the self-healing capability.

In this paper, we propose a new mechanism to achieve backward secrecy of the E-SGKD scheme. Compared with existing E-SGKD schemes [14,15] with backward secrecy, our proposed scheme has full self-healing properties, that is, user nodes can recover all of the lost session keys. In Schemes [14,15], user nodes can only recover part session keys determined by sliding windows. In addition, the communication overhead in our proposed scheme is low.

Except for the method of Lagrange Interpolation, there is another method to construct E-SGKD schemes. The core idea is that the computational operations of recovering the session key are moved to the exponent. Based on this idea, in this paper, we present a secure E-SGKD scheme with high efficiency. To make the new scheme easily understood, we first present a basic E-SGKD scheme based on Hong *et al.*'s Construction 2 [22]. In the basic construction, to reduce the users' storage overhead, only one secret polynomial is selected as a secret polynomial. However, if this secret polynomial is repeatedly used, all basic security properties are destroyed. Hence, a random value v_j for each session is chosen to update the secret polynomial for each session. Unfortunately, such an E-SGKD scheme still does not have backward secrecy.

Based on the basic construction, we further present the new E-SGKD scheme with backward secrecy, optimal storage and low communication bandwidth using two strategies. The first strategy is used to construct the revocation polynomials in the broadcast messages, thus achieving backward secrecy. More precisely, the group users, whose identities are used to compute the revocation polynomials, are divided into different subgroups according to their joined sessions. The second strategy, dual chains, are used for efficient seal-healing of the lost session keys. As we know, the hash chain is a useful tool to reduce the communication overhead in efficient seal-healing mechanisms. However, we find that the P-SGKD schemes with hash chains can not be converted to the E-SGKD schemes directly, and we will discuss the details later. To minimize the communication overhead, we introduce the dual chains. The first chain is a traditional hash chain, and the second chain is a key chain. Two chains are combined together to help the active group users compute the lost session keys, which reduces the number of the broadcast messages. Note that these two strategies, especially the first one, can be applied to transform other P-SGKD schemes to E-SGKD schemes. The new E-SGKD scheme has the following advantages:

- The new E-SGKD scheme solves the backward secrecy of E-SGKD schemes perfectly, *i.e.*, the proposed scheme can satisfy the backward secrecy, and furthermore can resist the collusion attack. The construction method of this scheme can be applied to convert other P-SGKD schemes to secure E-SGKD schemes.
- The storage overhead of the new schemes is optimal, *i.e.*, one element in Z_p.
- Thanks to the dual chains, the new E-SGKD scheme minimizes the communication cost, *i.e.*, the number of the broadcast messages is reduced to the number of the sessions in which new group users join in.
- The new E-SGKD scheme is computationally secure, *i.e.*, its security is based on the discrete logarithm problem.

The rest of the paper is arranged as follows. Section 2 defined the security model of this paper. Section 3 presents the basic E-SGKD scheme. Section 4 shows the novel E-SGKD scheme. Section 5 introduces the security analysis and performance comparison. Section 6 presents the practicality in the ZigBee network of the novel E-SGKD scheme. Application to Supervisory Control And Acquisition (SCADA) in smart grid is shown in Section 7. The conclusions are presented in Section 8.

2. Security Model

In this section, we introduce the network model, the notations and the hypothesis following Rams *et al.*'s survey [21].

2.1. Network Model

The network consists of a user node set $U = \{U_1, \cdots, U_N\}$ and a single GM. GM has rich resources such as large memory space and unlimited energy resources, and powerful ability including high computational ability. Instead, the resources and ability of user nodes are limited. In the resource-constrained networks, especially, the resources of the user nodes are lower.

GM communicates with the group user nodes under the unreliable channel. Message encryption and authentication by a symmetric group key can guarantee the secure group communication. The network is dynamic, and the user nodes may frequently join and leave the network. The leaving nodes may disclose the group key, thus breaking the security of group communication. Hence, the group key should be changed when there are user nodes joining and leaving the group. In addition, a minimal time interval should be set to change the group key even if the network is changeless. Thus, achieving secure group key distribution is necessary.

2.2. General Description of SGKD

In order to achieve secure group communication, group keys need to be changed frequently. Group lifetime is divided into epochs called sessions, where each session has a unique group key. In each session, GM distributes a new session key K_j to nodes in G_j by broadcasting the key updating messages.

Generally speaking, an SGKD scheme consists of six algorithms.

- **SetUp:** GM constructs personal secret S_i for each legitimate group node, and sends it to U_i by secure channel. U_i can use personal secret S_i to recover session keys from broadcast messages.
- **Broadcast:** GM creates message B_j from K_j according to the following conditions:
  - There exists a algorithm η, which for all $i : U_i \in G_j$, can recover K_j with the knowledge of S_i, that is: $K_j = \eta(B_j, S_i)$.
  - For any set of nodes $R \subset U \backslash G_j$, there exists no computational algorithm, ς, which can recover K_j with the knowledge of personal secrets of all nodes in R that is: $K_j = \varsigma(B_j, \{S_l\}_{l:U_l \in R})$ is not feasible.
- **SessionKeyRecovery:** This algorithm is executed by user nodes. Each member $U_j \in G_j$ recovers key K_j from broadcast message B_j with her personal secret S_i that is: $K_j = \eta(B_j, S_i)$.
- **SelfHealing:** This algorithm is executed by user nodes to recover lost session keys. Given l, r, there exists an algorithm ζ, which can recover K_j with the knowledge of B_r by node $U_i \in G_l \cap G_j \cap G_r$, that is $K_j = \zeta(B_r, S_i)$, where $l < j < r$.
- **GroupMemberAddition:** When a node U_i joins the group, GM sends his personal secret S_i via a secure channel.
- **GroupMemberRevocation:** When U_i is revoked from the group. The GM starts a new session and updates the session key, which can not be computed by U_i.

2.3. Definition of Self-Healing Group Key Distribution

In this subsection, we introduce the definition and security properties of SGKD scheme. In order to facilitate the narrative, we first list the notations in Table 1.

Table 1. Notations.

U_i	the i-th user node
m	the maximum sessions
t	the maximum revoked users
v	the number of the sessions in which there are users joined the group ($1 \leq v \leq j$)
F_p	a finite field of order p, where p is a prime
F_q^*	a multiplicative group of finite of order q
g	a generater of F_q^*
$\mathcal{S}(i)$	U_i's personal secret
$E_k(\cdot)/D_k(\cdot)$	symmetric encryption/decryption function
B_j	the j-th key updating broadcast message
$h_1(\cdot), h_2(\cdot)$	one-way hash function
ε_j	the unique session identifier, chosen at random by GM for users who joined the group in session j, $\varepsilon_j \in F_q$ and $\varepsilon_{j_1} \neq \varepsilon_{j_2}$ for $j_1 \neq j_2$
k_j^1	the initial value of j-th key chain chosen at random by GM for session j, $k_j^1 \in F_q$, and $k_{j_1}^1 \neq k_{j_2}^1$ for $j_1 \neq j_2$
$k_j^{j'}$	the j'-th key in the j-th key chain
$R_j^{j'}$	the set of users joining group in session j' and revoked before or in session j ($j^{'} \leq j$)
$\lvert R_j^{j'} \rvert$	the number of users in $R_j^{j'}$, and $\lvert R_j^{j'} \rvert \leq t$
R_j	the set of users who are revoked before and in session j, and $R_j = \{R_j^1, \cdots, R_j^j\}$
$\lvert R_j \rvert$	the number of users in R_j
$G_j^{j'}$	the set of group members joining the group in session j and still legitimate in session j ($j' \leq j$)
$\lvert G_j^{j'} \rvert$	the number of users in $G_j^{j'}$
G_j	the set of legitimate group user in session j, and $G_j = \{G_j^1, \cdots, G_j^j\}$
$\lvert G_j \rvert$	the number of users in G_j

Definition 1. *(self-healing key distribution with mt-revocation capability). The scheme has mt-revocation capability and self-healing property if*

(1) For a legitimate user U_i, $U_i \in G_j^{j'}, 1 \leq j^{'} \leq j$, the session key K_j can be computed by the j-th broadcast message B_j, and U_i's personal secret $\mathcal{S}_i$.

(2) Either broadcast packet B_j or personal secret $\mathcal{S}_i$ alone can obtain any information about K_j ($j \geq 1$).

(3) mt-revocation capability: For all $U_i \notin R_j$, U_i can compute K_j if given the j-th broadcast message B_j. However, the revoked user $U_i \in R_j$ can not, where $R_j = \{R_j^1, R_j^2, \cdots, R_j^j\}$ and $R_j^{j'}$, denote the users joining the group in session $j^{'}$ and revoked before and in session j.

(4) Self-healing property: For any j ($1 \leq j_1 \leq j \leq j_2$), a user, U_i ($U_i \in G_{j_1} \cap G_{j_2}$), can recover the session key K_j from broadcast messages B_{j_2}.

Definition 2. *(mt-wise forward secrecy). The scheme has mt-wise forward secrecy, if all users in R_j can not obtain information about K_{j+1} even knowing session keys $K_{j'}$ (j'<j), where $R_j \subseteq U$, $\lvert R_j \rvert \leq jt$, and R_j contains all users revoked before session j.*

Definition 3. *(any-wise backward secrecy). The scheme has any-wise backward secrecy if users in D_j can not obtain information K_j even knowing session keys $K_{j'}$ ($j' > j$), where D_j denotes users joining the group after session j ($D_j = \{D^{j+1}, D^{j+2}, \cdots\} \subseteq U$) and $D^{j'}$ contains users joining the group in session j' ($j' \geq j+1$).*

Definition 4. *(resistance to mt-wise collusion attack). The scheme has mt-wise collusion resistance capability if given any two disjoint sets R_{j_1},D_{j_2}, users in R_{j_1} colluding with users in D_{j_2} can not recover $K_j (j_1 \leq j \leq j_2)$ even knowing $\{B_1, B_2, \cdots, \{\mathcal{S}_i | U_i \in R_{j_1}\}\} \bigcup \{B_1, B_2, \cdots, \{\mathcal{S}_i | U_i \in D_{j_2}\}\}$.*

3. The Basic E-SGKD Scheme

Rams *et al.* [21] pointed out that almost all of the P-SGKD schemes can be converted to the E-SGKD schemes. Up to now, all P-SGKD schemes are divided into two classes based on if they use Lagrange Interpolation or not. As we surveyed in Section 1, the published E-SGKD schemes, Construction 5 [1], Scheme 4 [2], the schemes [14,15] are constructed based on the P-SGKD schemes with Lagrange Interpolation.

The other kind of P-SGKD schemes, without Lagrange Interpolation, can be divided into another two classes based on if they use hash chains or not. We checked all P-SGKD schemes without Lagrange Interpolation one by one, and found that the P-SGKD schemes with hash chains can not move the computational operations from the polynomial to the exponential, since the recursion of the one-way hash chain could not hold on once transferring the computation to the exponential. Precisely speaking, it is easy to compute $H(H(x))$ from $H(x)$ while it's hard to compute $g^{H(H(x))}$ from $g^{H(x)}$. On the other hand, the revocation polynomial based SGKD schemes without hash chains are suitable to be transformed to E-SGKD schemes, such as Scheme 3 in [18] and Scheme 2 in [22]. Since the transformation method is similar, in this paper, we take Hong *et al.*'s Scheme 2 as an example to construct the basic E-SGKD Scheme.

3.1. The Basic Construction

The basic construction includes five procedures: SetUp, Broadcast, SessionKeyRecovery, GroupMemberAdddtion and GroupMemberRevocation.

- **SetUp**

  Suppose $G_1 = \{U_1, U_2, \cdots, U_N\}$ denotes the users who join the group in the initial session. Each user U_i has a unique identity i. GM randomly selects a t-degree polynomial $f(x) = a_0 + a_1 x + \cdots + a_t x^t \in F_p[x]$ as a secret masking polynomial. Then, the GM distributes the personal secret $\mathcal{S}_i = \{f(i)\}$ to each user $U_i \in G_1$ via a secure channel, where using secret splitting algorithms in [37] has a better secrecy compared with distributing $f(i)$ to U_i directly.

- **Broadcast**

  Suppose $R_j = \{r_1^j, r_2^j, \cdots, r_{\omega_j}^j\}$ denotes a set of users who are revoked before and in session j, where $|R_j| = \omega_j \leq t$.

  – The GM constructs the j-th revocation polynomial as

  $$r_j(x) = (x - r_1^j)(x - r_2^j) \cdots (x - r_{\omega_j}^j)$$

  where $r_{j'}^j$ $(1 \leq j' \leq \omega_j)$ denotes the identity of user $U_{j'}$

  – The GM selects a random value K_j, v_j from F_p, and computes g^{v_j} and

  $$g^{P_j(x)} = g^{r_j(x)K_j + v_j \cdot f(x)}$$

  Then, the GM constructs the broadcast message as

  $$B_j = (\cup_{i=1}^{j} R_i) \cup \{g^{P_1(x)}, g^{P_2(x)}, \cdots, g^{P_{j-1}(x)}, g^{P_j(x)}\} \cup \{g^{v_{j'}}\}_{j'=1,2,\cdots,j}$$

  Note that if $P_j(x) = b_0 + b_1 x + \cdots + b_t x^t$, $g^{P_j(x)} = g^{b_0} \cdot (g^{b_1})^x \cdots (g^{b_t})^{x^t}$. Let $g^{P_j(x)}$ in B_j denote the sequence of $\{g^{b_0}, g^{b_1}, \cdots, g^{b_t}\}$.

- **SessionKeyRecovery**
  - For a legitimate user U_i, $U_i \in G_j$ recovers the j-th session key g^{K_j} by broadcast message B_j as follows:
    - U_i first uses his personal secret $f(i)$ to compute $(g^{v_j})^{f(i)}$.
    - U_i computes $g^{P_j(i)}$.
    - U_i evaluates $r_j(i)$.
    - Since
    $$g^{P_j(i)} = g^{r_j(i)K_j + v_j f(i)}$$
    U_i computes the session key as
    $$g^{K_j} = \left(\frac{g^{P_j(i)}}{g^{v_j f(i)}}\right)^{\frac{1}{r_j(i)}}$$
  - Similarly, U_i can recover the lost session keys $g^{K_{j'}}$ by using $R_{j'}$, $g^{P_{j'}(x)}$, $g^{v_{j'}}$ $(1 \leq j' < j)$ adopting the same method, *i.e.*, self-healing property.
  - For a revoked user $U_i \in R_j$, $r_j(i) = 0$. Thus, he can not obtain information about the session key g^{K_j}.
- **GroupMemberAddition**

  When a user, U_k, joins the group in session j, the GM randomly selects a unique identity $k \in F_p$ at random, and U_k gets his personal secret $\mathcal{S}_k = \{f(k)\}$ from the GM via a secure communication channel. For security, GM starts a new session.
- **GroupMemberRevocation**

  When a user, U_i, is revoked in session j, the GM then includes $(x - i)$ in $r_j(x)$ and starts a new session.

Remark 1. *In order to guarantee that the users' personal secret can be reused, we choose a mask value v_j in session j to multiply the secret polynomial. Thus, different sessions have different secret values, which contributes to the constant storage overhead.*

3.2. The Security Problem

It is easy to analyze that the above basic E-SGKD scheme satisfies the forward security and has t-revocation capability. Unfortunately, it has an obvious weakness, *i.e.*, it can not achieve the backward secrecy. More precisely, for a user U_i who joins the group in session j+1, he can recover the session key g^{K_j} by the j-th broadcast message B_j as follows:

- U_i computes $g^{P_j(i)}$.
- Since U_i is not a revoked user in session j, *i.e.*, $i \notin R_j$, $r_j(i) \neq 0$. Thus, U_i computes $r_j(i)$.
- Note
$$g^{P_j(i)} = g^{r_j(i)K_j + v_j f(i)}$$
U_i computes the j-th session's session key g^{K_j} as
$$g^{K_j} = \left(g^{P_j(i)} / (g^{v_j})^{f(i)}\right)^{\frac{1}{r_j(i)}}$$

Hence, a new user U_i, who joins the group in session $j + 1$, recovers the session key g^{K_j}, even if he is not a legitimate user in the session j. Thus, the basic E-SGKD scheme does not satisfy backward secrecy.

3.3. The Countermeasure

The reason why the basic E-SGKD scheme can not satisfy the backward secrecy lies in the fact that a user's personal secret $\mathcal{S}(i)$ does not change in the different sessions. As we mentioned above, E-SGKD schemes reduce the size of the personal secret to a constant, *i.e.*, a personal secret $\mathcal{S}(i)$ only relates to a user's identity, no matter when he joins the group. This means a new user, who joins the group later, can use his personal secret to recover the past session keys.

From the above analysis, we know that to achieve the backward secrecy, a user's personal secret should be changed in the different sessions. However, allocating different personal secrets to different sessions would create linear storage overhead. How to balance the storage overhead and security is a challenging task.

To solve this problem, we consider binding each user's personal secret with a changed value. More precisely, users are divided into different subgroups according to the sessions in which they join the group. The core idea is described as follows: a unique session identifier ε_j is assigned to each session, and is multiplied with the secret polynomial to produce a new secret polynomial. Thus the personal secret for a user U_i who joins the group in session j is $\varepsilon_j \cdot f(i)$. As a result, user U_i can not recover the previous session keys with his personal secret $\varepsilon_j \cdot f(i)$, *i.e.*, backward secrecy is achieved. On the other hand, it is easy to see that the storage overhead is optimal since $\varepsilon_j \cdot f(i)$ is a random value in F_p.

The above idea, binding each user's personal secret with a changed value, can help the scheme gain the backward secrecy. However, this idea can not be directly applied to the basic construction, since the self-healing property is destroyed. To gain efficient self-healing, dual chains are introduced in the basic construction. The first chain is a hash chain, and the second chain is a key chain.

4. The Novel E-SGKD Scheme

Motivated by the above idea, in this section, we present a new E-SGKD scheme which consists of five procedures, *i.e.*, SetUp, Broadcast, SessionKeyRecovery, GroupMemberAdddtion, GroupMemberRevocation.

- **Setup**

  GM picks a t-degree polynomial $f(x) = a_0 + a_1 x + \cdots + a_t x^t \in F_p[x]$ at random and keeps it secret. The GM randomly selects a one-way hash function $h_2(\cdot)$: $\{0,1\}^{512} \rightarrow \{0,1\}^{128}$. Then, the GM selects a session identifier $\varepsilon_1 \in F_p$ at random. Note that the GM will randomly select a session identifier $\varepsilon_j \in F_p$ in session j. Each user $U_i \in G_1$ obtains his personal secret $\mathcal{S}_i = \{\varepsilon_1 \cdot f(i)\}$ through a secure channel, where G_1 includes the group members who join the group in the initial session and, using secret splitting algorithms in [37], has a better secrecy compared with distributing $\varepsilon_1 \cdot f(i)$ to U_i directly.
- **Broadcast**

  Suppose $R_j = \{R_j^1, R_j^2, \cdots, R_j^j\}$ and $R_j^{j'} = \{U_{r_1^{j'}}, U_{r_2^{j'}}, \cdots, U_{r_{\omega_{j'}}^{j'}}\}$, where $R_j^{j'}$ consists of the users joining the group in session j' and is revoked before and in session j, and $|R_j^{j'}| = \omega_{j'}$ for each $j'(1 \leq j' \leq j)$. Let $r_1^{j'}, r_2^{j'}, \cdots, r_{\omega_{j'}}^{j'}$ denote the revoked users' identities in $R_j^{j'}$. Note that $R_j^{j'} = \emptyset$ if no users leave the group in session j'.

  – The GM randomly selects $k_j^1 \in F_p$, $r_0 \in F_p$, and a one-way hash function $h_1(\cdot)$: $\{0,1\}^{128} \rightarrow \{0,1\}^{128}$. The GM uses r_0 as a seed and $h_1(\cdot)$ as a hash function to construct a hash chain as follows:

$$\begin{aligned} r_1 &= h_1(r_0) \\ r_2 &= h_1(r_1) = h_1(h_1(r_0)) = h_1^2(r_0) \\ &\cdots, \\ r_{j-1} &= h_1(r_{j-2}) = \cdots = h_1^{j-1}(r_0) \end{aligned}$$

Then, the GM constructs the j-th keys chain as follows:

$$\begin{aligned} k_j^2 &= k_j^1 \cdot r_1 \\ k_j^3 &= (k_j^2) \cdot r_2 \\ &\cdots, \\ k_j^j &= (k_j^{j-1}) \cdot r_{j-1} \end{aligned}$$

- The GM selects the random value $r_k^{'j'} \in F_p (1 \leq k \leq t - \omega_{j'})$ for $R_j^{'j'}$, where $R_j' = \{R_j'^1, R_j'^2, \cdots, R_j'^j\}$ and $R_j^{'j'} = \{r_1^{'j'}, r_2^{'j'}, \cdots, r_{t-\omega_{j'}}^{'j'}\}$. $r_k^{'j'} (1 \leq k \leq t - \omega_{j'})$ are different from each other and are never used for users' identities. Then, the revocation polynomials are constructed as:

$$A_j^{j'}(x) = \Pi_{i=1}^{|R_j^{j'}|}(x - r_i^{j'}) \Pi_{i=1}^{|R_j^{'j'}|}(x - r_i^{'j'}), j' = 1, 2, \cdots, j$$

where $|R_j^{'j'}| = t - |R_j^{j'}|$.

Note that $R_j^{j'}$ denotes the set of users joining the group in session j' and is revoked before and in session j. The number of users in $R_j^{j'}$ is less than t, thus the degree of $\Pi_{i=1}^{|R_j^{j'}|}(x - r_i^{j'})$ is less than t. However, the degree of $f(x)$ is t. In the following computation, $A_j^{j'}(x) \cdot k_j^{j'} + v_j \cdot \varepsilon_{j'} \cdot f(x)$ may expose some coefficients of $\varepsilon_{j'} \cdot f(x)$. In order to keep the coefficients the polynomial $\varepsilon_j f(x)$ secret, the set $R_j^{'j'} = \{r_1^{'j'}, r_2^{'j'}, \cdots, r_{t-\omega_{j'}}^{'j'}\}$ is randomly selected to pad the revocation polynomials to be t-degree, where $r_k^{'j'} \in F_p (1 \leq k \leq t - \omega_{j'})$ for $R_j^{'j'}$ is different from each other and never used for users' identities. Thus, the coefficients of the secret polynomial $\varepsilon_{j'} f(x)$ can be protected better.
- The GM randomly selects a random value $v_j \in F_p$. Then, the GM computes

$$g^{\Phi_j^{j'}(x)} = g^{A_j^{j'}(x) \cdot k_j^{j'} + v_j \cdot \varepsilon_{j'} \cdot f(x)}, \; j' = 1, 2, \cdots, j$$

and constructs the broadcast message as

$$B_j = R_j \cup R_j' \cup \{g^{\Phi_j^{j'}(x)}\}_{j'=1,2,\cdots,j} \cup \{E_{h_2(g^{k_j^{j'}})}(r_{j'})\}_{j'=1,2,\cdots,j-1} \cup \{E_{h_2(g^{k_j^{j'}})}(K_{j'})\}_{j'=1,2,\cdots,j} \cup g^{v_j}$$

Note that if $\Phi_j^{j'}(x) = b_0 + b_1 x + \cdots + b_t x^t$, $g^{\Phi_j^{j'}(x)} = g^{b_0} \cdot (g^{b_1})^x \cdots (g^{b_t})^{x^t}$. Let $g^{\Phi_j^{j'}(x)}$ in B_j be the sequence of $\{g^{b_0}, g^{b_1}, \cdots, g^{b_t}\}$.

- **SessionKeyRecovery**

  - For a legitimate user U_i, $U_i \in G_j^{j'}$ uses the j-th broadcast message to compute the current session key K_j and recover the lost session keys as:

    * U_i computes $g^{\Phi_j^{j'}(i)}$, where

$$g^{\Phi_j^{j'}(i)} = g^{A_j^{j'}(i)\cdot k_j^{j'} + v_j \cdot \varepsilon_{j'} \cdot f(i)}$$

      U_i computes $A_j^{j'}(i)^{-1}$, and uses his personal secret $\varepsilon_{j'} \cdot f(i)$ to compute

$$g^{k_j^{j'}} = \left(\frac{g^{\Phi_j^{j'}(i)}}{g^{v_j \cdot \varepsilon_{j'} \cdot f(i)}}\right)^{\frac{1}{A_j^{j'}(i)}}$$

    * U_i computes $h_2(g^{k_j^{j'}})$ and evaluates $r_{j'}$ by decrypting $E_{h_2(g^{k_j^{j'}})}(r_{j'})$. Then, U_i obtains $\{r_{j''}\}(j' \le j'' \le j-1)$ by using $h_1(\cdot)$.
    * U_i computes $g^{k_j^{j''}}$ $(j' \le j'' \le j)$ as

$$g^{k_j^{j'+1}} = (g^{k_j^{j'}})^{r_{j'}}$$

$$g^{k_j^{j'+2}} = (g^{k_j^{j'}})^{r_{j'} r_{j'+1}}$$

$$\cdots$$

$$g^{k_j^{j}} = (g^{k_j^{j'}})^{r_{j'} \cdots r_{j-1}}$$

    * Finally, U_i computes any $\{K_{j''}\}_{j' \le j'' \le j}$ by decrypting $\{E_{h_2(g^{k_j^{j''}})}(K_{j''})\}_{j' \le j'' \le j}$.

  - For a revoked user U_i, $A_j^{j'}(i) = 0$. Thus, he can not obtain any information about K_j.

- **GroupMemberAddition**

  When a user, U_v, joins the group in session j, GM randomly selects a unique identity v and a session identifier ε_j from F_p, and distributes a personal key $\mathcal{S}_v = \varepsilon_j \cdot f(v)$ to him via a secure communication channel. For security, GM starts a new session.

- **GroupMemberRevocation**

  When a user, U_i joins the group in session j' and is revoked in session j, GM includes $(x - i)$ in $A_j^{j'}(x)$ and starts a new session.

Remark 2. *In this scheme, the algorithm "Self Healing" is contained in algorithm "Session key recovery". That is, for a legitimate user U_i, by running the algorithm "Session key recovery", he can compute the current session key and recover lost session keys from current broadcast, that is, a self-healing property.*

Remark 3. *For a certain session, say j'', if there are not users joining group in session j'', then $R_j^{j''} = \emptyset$, thus we do not need to compute $g^{\Phi_j^{j''}(x)}$ and $\{E_{h_2(g^{k_j^{j''}})}(r_{j''})\}$. Suppose the number of the sessions in which the new users join is v, Then, the number of the polynomials $\{g^{\Phi_j^{j'}(x)}\}_{j'=1,2,\cdots,j}$ and the encrypted message*

$\{E_{h_2(g^{k_j^{j'}})}(r_{j'})\}_{j'=1,2,\cdots,j-1}$ *in the broadcast messages* B_j *are* v*, respectively. Note that* $v \leq j$*. Especially, if* v *is much smaller than* m*, the communication overhead would be reduced remarkably.*

5. Security Analysis and Performance Comparison

In this section, the security and the performance of new E-SGKD scheme will be analyzed.

5.1. Security Analysis

We present four theorems and proofs for the new E-SGKD scheme, which demonstrate that the new E-SGKD scheme has the security properties as defined in security model.

Theorem 5. *The new E-SGKD scheme is a secure SGKD scheme with a self-healing property and mt-revocation capability.*

Proof. According the definition in Section 2.3, the new E-SGKD scheme has a self-healing property and *mt*-revocation capability, because it satisfies the following conditions:

(1) For a legitimate user $U_i \in G_j^{j'}$, he can recover the session key K_j by combining B_j with his personal secret $\mathcal{S}_i$ as described in the SessionKeyRecovery procedure.

(2) On one hand, the session key K_j has a relationship with the initial value of the j-th masking key chain, k_j^1, and $r_{j'}(1 \leq j' \leq j)$. However, because of the revocation polynomial, it is difficult to compute k_j^1 and $r_{j'}(1 \leq j' \leq j)$ only using the broadcast messages. Therefore, using the broadcast message B_j alone can not obtain any information about K_j. On the other hand, K_j is chosen randomly and is independent of the personal secret so that using the personal secret alone can not obtain any information about K_j.

Thus, either the broadcast messages or the personal secrets alone can not obtain any information about K_j.

(3) We first consider a single user $U_i \in R_j$. For any revoked user $U_i \in R_j$, if $U_i \in R_j^{j'}$, $A_j^{j'}(i) = 0$. Hence, he can not obtain any information about $k_j^{j'}$. Thus, for any revoked user, he alone can not obtain any information about K_j. Furthermore, we consider the collusion of the users in R_j. Because the personal secret of a user has a relationship with a session in which he joins the group, only the users joining the group in the same session can collude together. According to the Lagrange Interpolation method, only at least $t+1$ users coalesce to recover the corresponding $\varepsilon_j \cdot f(x)$. Since $|R_j^{j'}| \leq t$, the coalition of users in R_j can not obtain $\varepsilon_j \cdot f(x)$. Therefore, $U_i \in R_j$ can not obtain information about K_j.

(4) From the SessionKeyRecovery procedure, we learn that a legitimate user can recover the lost session keys from his joined session to the current session, which demonstrates that the new scheme has a self-healing property.

Theorem 6. *The new E-SGKD scheme has mt-wise forward secrecy.*

Proof. We first consider a single user $U_i \in R_j$ who tries to recover the session key K_{j+1}. For a revoked user $U_i \in R_j^{j'}$, $A_{j+1}^{j'}(i) = 0$. Therefore, U_i can not obtain any information about $k_{j+1}^{j'}$. Hence, U_i can not recover K_{j+1}.

Now we consider the collusion of the users in R_j. As described above, only at least $t+1$ users who join the group in same session can collude to recover $\varepsilon_{j'} \cdot f(x)$. However, $|R_j^{j'}| \leq t$ so that $\varepsilon_{j'} \cdot f(x)$ can not be recovered. Thus, U_i can not obtain any information about $k_{j+1}^{j'}$ and the session key K_{j+1}. Therefore, the new E-SGKD scheme achieves *mt*-wise forward secrecy.

Theorem 7. *The new E-SGKD scheme guarantees any-wise backward secrecy.*

Proof. Users in D_j have to know at least t+1 users' personal secrets $\varepsilon_{j'} \cdot f(i)$ $(j' \leq j)$ which are distributed to those users who join the group in the same session, so that they can recover $\varepsilon_{j'} \cdot f(x)$, and, furthermore, recover the session key K_j. However, according to the definition of D_j, users in D_j join the group after the session j, so they only have $\varepsilon_{j''} \cdot f(i)(j'' \geq j+1)$. Thus, no matter how many users in D_j coalesce, they do not have enough personal secrets to recover K_j. Therefore, the new E-SGKD scheme guarantees *any*-wise backward secrecy.

Theorem 8. *The new E-SGKD scheme has mt-collusion attack resistance capability.*

Proof. Suppose R_{j_1} consists of all users revoked before and in session j_1, and D_{j_2} includes all users joining the group after session j_2 $(j_1 < j_2)$. Even if users in R_{j_1} collude with users in D_{j_2}, they can not recover K_j with the knowledge of B_{j_1}, B_{j_2} and users' personal secrets.

On one hand, a user $U_i \in R_{j_1}^{j'}(j' < j_1)$ only has $\varepsilon_{j'} \cdot f(i)$, and a user $U_v \in D_{j_2}^{j''}(j'' > j_2)$ only has $\varepsilon_{j''} \cdot f(v)$. However, only users joining the group in the same session can collude together and $|R_{j_1}^{j'}| \leq t$, $|D_{j_2}^{''}| \leq t$. Even if users in R_{j_1} collude with users in D_{j_2}, they can not obtain enough information to recover $\varepsilon_{j'} \cdot f(x)$ and $\varepsilon_{j''} \cdot f(x)$.

On the other hand, from Theorems 2 and 3, we learn that either the collusion of users in R_{j_1} or the collusion of users in D_{j_2} alone can not recover K_j.

Therefore, the new E-SGKD scheme has *mt*-collusion attack resistance capability.

5.2. Performance Comparison

In this subsection, we compare the basic E-SGKD scheme and the new E-SGKD scheme with the previous E-SGKD schemes from the security performance and the efficiency performance. Except for the published E-SGKD schemes, only Liu *et al.*'s Scheme 3 [18] and Hong *et al.*'s Scheme 2 [22] can be converted to the E-SGKD schemes. Here, "Liu *et al.*'s improved scheme" means the E-SGKD scheme constructed from Liu *et al.*'s Scheme 3 using the similar method in Section 3. In general, let p be a 128-bit integer and q be a 512-bit integer.

5.2.1. The Security Performance

From Table 2, it is easy to find that Construction 5 [1], Scheme 4 [2], "Liu *et al.*'s improved scheme" and our basic scheme do not satisfy the backward secrecy and are not resistant to the collusion attack. The Schemes [14,15] and our new scheme have all of the basic security properties, *i.e.*, forward secrecy, backward secrecy and resistance to collusion attack capability.

Additionally, our new scheme allows more users to be revoked and more users to be colluded together compared with the E-SGKD schemes [14,15]. Note that our new scheme has the capability of resisting *mt*-wise collusion attack and *mt*-wise forward secrecy, when there are users joining the group in every session. Our new scheme has the capability of resisting *vt*-wise collusion attack and *vt*-wise forward secrecy, when the number of the sessions in which there are users joining group is $v(v < m)$. Specially speaking, the collusion users are less than t for each session. Thus, the total number of the collusion users is less than vt.

Table 2. Comparison of security properties.

Scheme	Revocation Limit	Forward Secrecy	Backward Secrecy	Collusion Resistance	The Maximum Number of Collusion Attack Resistance
Construction 5 in [1]	t	Yes/t	No	No	0
Scheme 4 in [2]	t	Yes/t	No	No	0
Liu *et al.*'s improved scheme	t	Yes/t	No	No	0
Scheme in [14]	t	Yes/t	Yes/t	Yes	t
Scheme in [15]	t	Yes/t	Yes/t	Yes	t
Our basic scheme	t	Yes/t	No	No	0
Our new scheme	mt	Yes/mt	Yes/*any*	Yes	mt

5.2.2. The Storage Overhead

Now, we focus on the efficiency performance, including the storage overhead and the communication overhead. From Table 3, it is obvious that only scheme [15], our basic scheme and our new scheme have the constant storage overhead, *i.e.*, $\log_2 p$, which is optimal compared with other E-SGKD schemes.

Table 3. Comparison of storage overhead.

Scheme	Storage Overhead	Communication Overhead
Construction 5 in [1]	$(m^2+2)\log_2 p$	$(mt^2+2mt+m)\log_2 q + t\log_2 p$
Scheme 4 in [2]	$m\log_2 p$	$(tj+m+j)\log_2 q + tj\log_2 p$
Liu *et al.*'s improved scheme	$2m\log_2 p$	$[(m+j+1)t+(2m+1)]\log_2 q$
Scheme in [14]	$(m-j+2)\log_2 p$	$(d+1)[(t+2)\log_2 q + (3t+1)\log_2 p]$
Scheme in [15]	$\log_2 p$	$(d+1)[(t+2)\log_2 q + (t+1)\log_2 p]$
Our basic scheme	$\log_2 p$	$(t+2)j\log_2 q$
Our new scheme	$\log_2 p$	$[(t+1)v+1]\log_2 q + (v+j)\log_2 p$

5.2.3. The Communication Overhead

From Table 3, we can find that the communication overhead of Blundo *et al.*'s Scheme 4 [2], $[tj+m+j]\log_2 q + tj\log_2 p$, is smaller than that of Construction 5 in [1] and Liu *et al.*'s improved scheme. In addition, the communication overhead of scheme [15], $(d+1)[(t+2)\log_2 q + (t+1)\log_2 p]$ is smaller than that of scheme [14], $(d+1)[(t+2)\log_2 q + (3t+1)\log_2 p]$, where d is the size of sliding windows. Therefore, we mainly compare our basic E-SGKD scheme and our new E-SGKD scheme with Blundo *et al.*'s scheme 4 and scheme [15].

In the basic scheme, the broadcast message B_j in session j includes R_i, $\{g^{P_{j'}(x)}\}$, $\{g^{v_{j'}}\}$, where $j' = 1, 2, \cdots, j$. Because the users' identities can be chosen from a small finite, the communication overhead of R_j can be neglected. Thus, the communication overhead is $(t+2)j\log_2 q$, which is obviously less than the communication overhead of Blundo *et al.*'s Scheme 4, based on the fact that $j \leq m$. Similarly, in the new scheme, the broadcast message B_j in session j includes R_j, R'_j, $\{g^{\Phi_j^{j'}(x)}\}$, $\{E_{h_2(g^{k_j^{j'}})}(r_{j'})\}$, $\{E_{h_2(g^{k_j^{j'}})}(K_{j'})\}$ and g^{v_j}, where $j' \in \{1, 2, \cdots, j\}$. Note that the communication overhead of R_j, R'_j can be neglected. Thus, the communication overhead is $[(t+1)v+1]\log_2 q + (v+j)\log_2 p$, where v is the number of the sessions in which the new users join. Let $B_1 = (t+2)j\log_2 q$, $B_2 = [(t+1)v+1]\log_2 q + (v+j)\log_2 p$ and $v = j$ which means all sessions have new users join in, and let p be a 128-bit integer and q be a 512-bit integer. Then, we have:

$$\begin{aligned} & B_1 - B_2 \\ = \; & (t+2)j\log_2 q - \{[(t+1)v+1]\log_2 q + (v+j)\log_2 p\} \\ = \; & (tj+2j-tj-j-1)\log_2 q - 2j\log_2 p \\ = \; & (j-1)\log_2 q - 2j\log_2 p \\ = \; & 512(j-1) - 128\times 2j \\ = \; & 256(j-2) \end{aligned}$$

It is obvious that the communication overhead in our new scheme is lower than the communication overhead of our basic scheme as long as $j > 2$.

Now, we analyze the relationship between the maximum size of the broadcast message and the degree t for different E-SGKD schemes. Suppose $[x]$ denotes that x rounds down to the nearest whole unit. Let $m = 30$, and t varies from 10 to 30.

(1) The comparison among our basic E-SGKD scheme, our new E-SGKD scheme and Blundo *et al.*'s Scheme 4.

Figure 1 describes the maximum size of the broadcast message changing with t, where $v = [0.2m]$, $[0.3m]$, $[0.5m]$ and $[v = m]$ are discussed, respectively. From Figure 1, when $m = 30, t = 30$, the maximum size of the broadcast message in our new E-SGKD scheme is nearly 12.25, 18.1094, 29.8281 and 59.125 KB with v = 6, 9, 15 and 30, while the maximum size of the broadcast message in Liu *et al.*'s improved scheme, Blundo *et al.*'s Scheme 4 and our basic E-SGKD scheme is nearly 118.1875, 74.0625 and 60 KB, respectively. Thus, our new E-SGKD scheme has the best performance, especially when the number of the joining sessions v is small since the size of the broadcast messages reduce with the value of v reduction. Note that when $v = [0.2m]$, the maximum size of the broadcast message in our new E-SGKD scheme is nearly 12.25 KB, which is even smaller than that (say 14.5313 KB) of Hong *et al.*'s Scheme 2 [22], which is the most efficient known P-SGKD scheme.

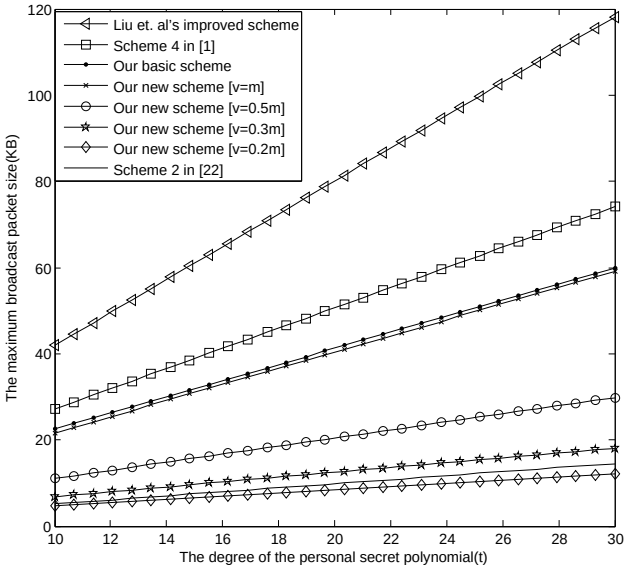

Figure 1. The first comparison of the broadcast message.

(2) The comparison between our new E-SGKD scheme and scheme [15]

The communication overhead in scheme [15] is $(d+1)[(t+2)\log_2 q + (t+1)\log_2 p]$, which has a relationship with the size of the sliding windows. In general, we assume that the size of sliding windows (say d) is equal to the number of sessions (say v), in which there are users joining the group. Figure 2 describes the maximum size of the broadcast message changing with t, where $v = d = [0.2m]$, $[0.3m]$, $[0.5m]$ and $[v = m]$ are discussed, respectively. From Figure 2, when $m = 30, t = 30$, the maximum size of the broadcast message in our new E-SGKD scheme is nearly 12.25, 18.1094, 29.8281 and 59.125 KB with v = 6, 9, 15 and 30, while the maximum size of the broadcast message in scheme [15] is nearly 17.39, 24.84, 39.75 and 77.01 KB, respectively. Thus, our new E-SGKD scheme has lower communication overhead, when $v = d$.

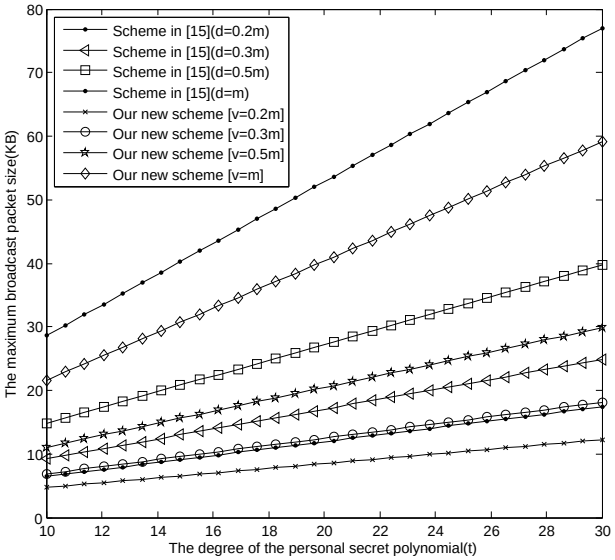

Figure 2. The second comparison of the broadcast message.

To sum up, our new E-SGKD scheme has a smaller broadcast size compared with the E-SGKD Schemes [1], [2] and "Liu *et al.*'s improved scheme". In addition, our new E-SGKD Scheme has a smaller broadcast size than the E-SGKD Scheme [15] when $v = d$. Hence, our new E-SGKD Scheme is efficient in terms of the communication overhead.

5.3. Practicality

Many specific issues should be taken into consideration when an SGKD Scheme is applied to real-word scenarios. First, the SGKD Sscheme should work well and efficiently complete the task of the key distribution in the specific scenarios. Second, the system parameters for these scenarios should be determined so that the SGKD Scheme and corresponding parameters can work efficiently.

As we know, for most of the SGKD schemes, the largest broadcast packet is supposed to be 64 KB, so the system parameters should be selected according to the principle of reducing the largest broadcast packet. The largest broadcast packet in the E-SGKD scheme is mainly determined by p, q, m and t. In general, suppose p is a 128-bit integer and q is a 512-bit integer. Then, we simulate the E-SGKD schemes that are applied in the wireless network in which the broadcast packet is 64 KB. The simulation results will contribute in analyzing how to select parameters.

(1) The relationship between m and t.

Figure 3 describes the relationship between m and t in the known E-SGKD schemes, our basic E-SGKD scheme and our new E-SGKD scheme, when the largest broadcast packet is constrained to 64 KB. From Figure 3, when $m = 10$, the largest degree of personal secret polynomial t in

our basic E-SGKD scheme and our new E-SGKD scheme with $v = m$ are closed, *i.e.*, 100 and 94, respectively, while t reaches 46 and 80 in "Liu *et al.*'s improved scheme" and Blundo *et al.*'s scheme [2]. When $m = 30$, the values of t in our basic E-SGKD scheme and our new E-SGKD scheme with $v = m$ are the same, *i.e.*, 32, while the values of t in "Liu *et al.*'s improved scheme" and Blundo *et al.*'s Scheme [2] are 15 and 25, respectively.

It is obvious that the range of m and t in our new scheme is larger than the range in other E-SGKD schemes when $v < m$. For example, when $v = [0.5m]$ and $v = [0.2m]$, where $m = 10$, the values of t in our new E-SGKD scheme are increased remarkably, say 202 and 509, respectively. Note that, when $v = [0.2m]$, the value of t even is larger than the value of t (say 410) in the most efficient known P-SGKD scheme, *i.e*, Hong *et al.*'s Scheme 2 [22]. Thus, our new E-SGKD scheme has the best performance.

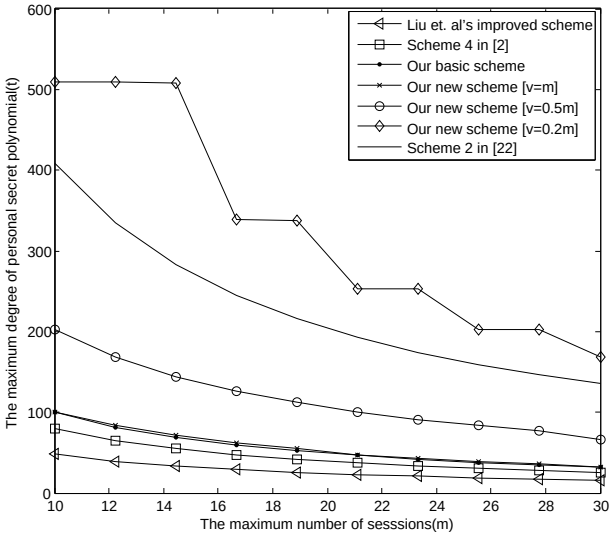

Figure 3. The trade-off between m and t as the size of the broadcast message is 64 KB.

- The relationship between m and the maximum revoked users in all sessions $|R_m|_{max}$.

  Figure 4 presents the relationship between m and the maximum revoked users in all sessions $|R_m|_{max}$, where $v = [0.2m]$, $[0.3m]$, $[0.5m]$ and $[v = m]$, respectively. From Figure 4, when $m = 30$, $|R_m|_{max}$ in our new E-SGKD scheme is nearly 1000 with v = 6, 9, 15 and 30, while $|R_m|_{max}$ in Liu *et al.*'s improved scheme, Blundo *et al.*'s Scheme 4 and our basic E-SGKD scheme is nearly 15, 25 and 32, respectively. Additionally, we can find that $|R_m|_{max}$ in the most efficient P-SGKD scheme, *i.e.*, Hong *et al.*'s Scheme 2 [22], is 135. Thus, our new E-SGKD scheme allows more revoked users than all other known E-SGKD schemes and P-SGKD schemes, *i.e.*, the new scheme can resist more users' collusion attacks.

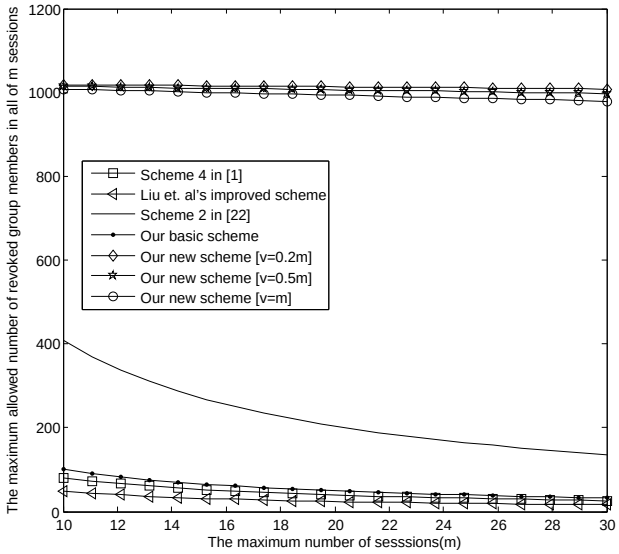

Figure 4. The trade-off between m and $|R_m|_{max}$ as the size of the broadcast message is 64 KB.

6. Practicality in ZigBee Network

In this section, we mainly discuss how to apply our new E-SGKD scheme to special kinds of resource-constrained wireless sensor networks, *i.e.*, ZigBee networks. For the resource-constrained wireless networks, resources including the users' storage and communication bandwidth is limited.

ZigBee protocol is designed for low-data-rate wireless networks and is very suitable for low-rate, low-cost and low-energy-consumption networks. As we know, for the ZigBee protocol [38,39], the maximum size of the Mac layer data is from 89 to 119 bytes. When the the maximum size of the Mac layer data is 89 bytes, if the data of the application layer are more than 89 bytes, the data will be divided into blocks. Assume that the maximum size of the broadcast message are 4 KB. Then, the broadcast message will be divided into 46 small packets with 88 bytes/packet. Without loss of generality, suppose packets are lost randomly and independently. When the packet loss rate is 1%, only 37.01% of packets reach their destination. When the packet loss rate is 5%, only 9.45% of packets reach their destination, *i.e.*, only one packet in ten reaches its destination and is received by a group member. Hence, suppose m is at least 10.

Under the above assumption, *i.e.*, m equals 10 and the maximum size of the broadcast message is 4 KB, we now check if the known E-SGKD schemes are suitable for ZigBee wireless networks or not. From Figure 5, we can find that when t = 10, the size of the broadcast message is more than 4 KB in the most efficient E-SGKD scheme, *i.e.*, our new scheme with $[v = m]$, and is less than 4 KB in our new E-SGKD scheme with $v = [0.2m]$, $[0.3m]$ and $[v = m]$ and Hong *et al.*'s efficient P-SGKD Scheme 2. Meanwhile, the size of the broadcast message increases as t increases. When t reaches 20, the size of the broadcast message is more than 7 KB in our new scheme with $[v = 0.5m]$, while is still less than 4 KB in our new E-SGKD scheme with $v = [0.2m]$, $[0.3m]$ and Hong *et al.*'s efficient P-SGKD scheme. Note that when $v = [0.2m]$, the communication overhead is even lower than the most efficient P-SGKD scheme.

Thus, we can conclude that all of the known E-SGKD schemes can not be applied to the ZigBee network, and only our new E-SGKD scheme with $v = [0.2m]$, $[0.3m]$ is suitable for the ZigBee network, since it also has optimal storage overhead.

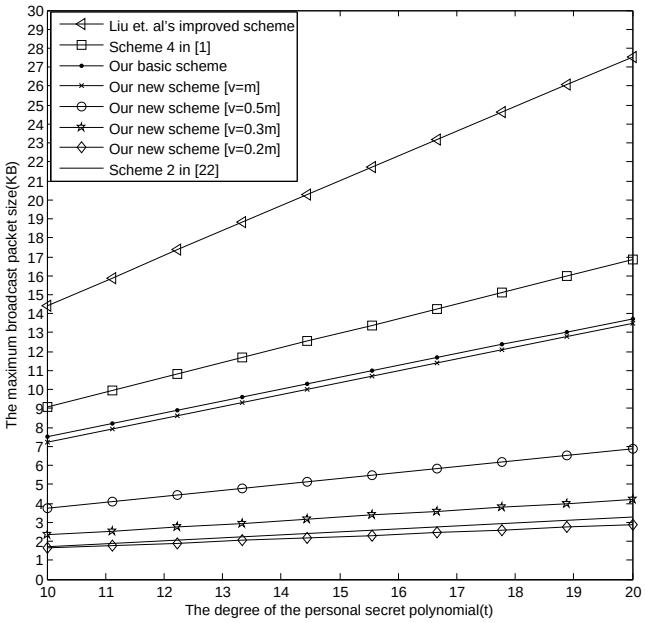

Figure 5. The comparison of the broadcast message when m is 10.

Now, we discuss how to select the system parameters when applying our new E-SGKD scheme to the ZigBee-based wireless network. The simulation results will contribute to analyzing how to select parameters. Suppose the largest broadcast packet in ZigBee-based wireless networks is 4 KB. As discussed above, we only analyze our new E-SGKD scheme with $v = [0.2m]$, $[0.3m]$.

From Figure 6, the relationship between m and t in our new E-SGKD scheme with $v = [0.2m]$, $[0.3m]$ and Hong *et al.*'s efficient P-SGKD Scheme 2 [22] is described when the largest broadcast message is constrained to 4 KB. It is obvious that the range of m and t in our new E-SGKD scheme with $v = [0.2m]$ is the largest among three schemes. Thus, if the number of the sessions in which the new users join, *i.e.*, v, is more smaller, m and t can be bigger. For example, if $v = [0.2m]$, then $m = 10$ and $t = 29$, or $m = 15$ and $t = 25$, or $m = 20$ and $t = 15$, or $m = 28$ and $t = 10$.

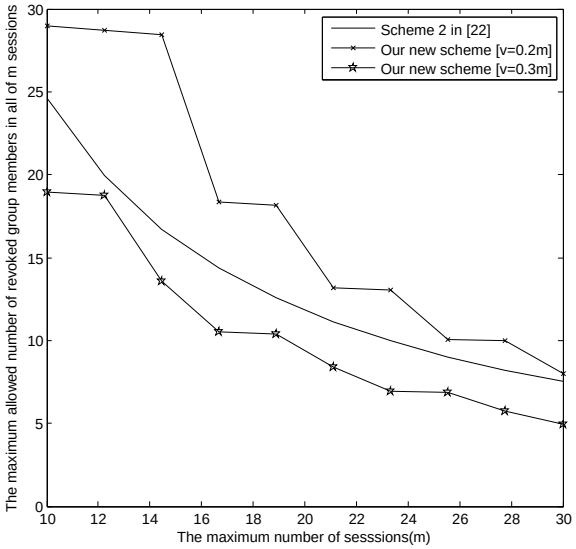

Figure 6. The trade-off between m and t as the size of the broadcast message is 4 KB.

Figure 7 presents the relationship between m and the maximum revoked users in all sessions $|R_m|_{max}$, which demonstrates that our new E-SGKD scheme with $v = [0.2m]$ and $[0.3m]$ allows more revoked users than Hong *et al.*'s efficient P-SGKD scheme 2. For example, when $m = 10$, $t = 29$ and 19, with $v = 2$ and 3, the maximum allowed revoked members in all m sessions are 58 and 57, while the allowed group members in all m sessions in Hong *et al.*'s efficient P-SGKD scheme 2 [22] is only 24. Under the same conditions, when $m = 30$, $t = 8$ and 4, with $v = 6$ and 9, the maximum allowed revoked members in all m sessions are 48 and 36, while the allowed group members in all m sessions in Hong *et al.*'s efficient P-SGKD Scheme 2 [22] is only eight.

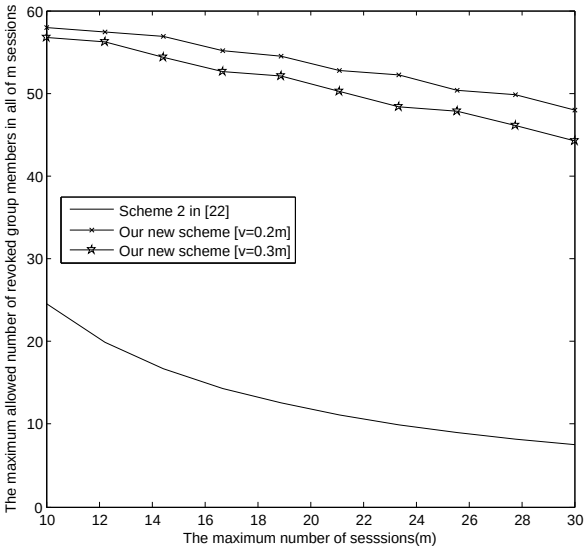

Figure 7. The trade-off between m and $|R_m|_{max}$ as the size of the broadcast message is 4 KB.

In conclusion, if the new users do not join the group frequently, the number of the sessions in which new users join is small. In this case, our new scheme is efficient in terms of the storage overhead and the communication overhead. When $v = [0.2m]$, the overall efficiency is higher than the most efficient known P-SGKD.

7. Application to Supervisory Control And Data Acquisition (SCADA) in Smart Grid

Smart grids are becoming more and more important in modern society. SCADA systems are applied to monitor and control smart grids. The SCADA system consists of human-machine interface (HMI), master terminal unit (MTU), and remote terminal unit (RTU). The structure of these entities is as described in Figure 8 ([40]). HMI is a human-computer interaction device. MTU is in charge of supervisory control to the RTUs. As shown in Figure 8, the SCADA system consists of one MTU and multiple sub-MTUs, and these MTUs have rich resources such as storage space and computational capability. Thus, the public key cryptography can be used to protect the security among them. Compared with MTU, the resources of RTUs are limited. In addition, the RTUs are often located in remote places and the security can not be guaranteed.

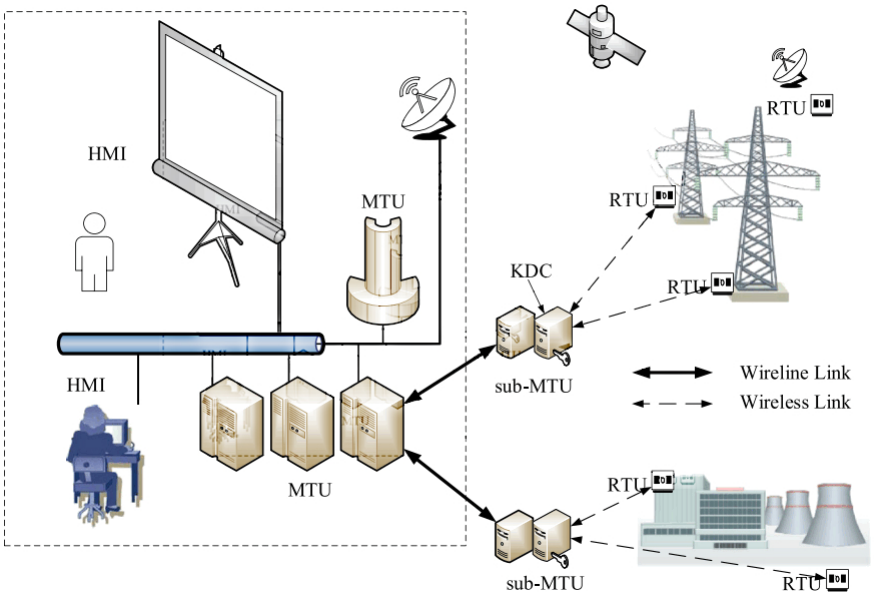

Figure 8. A simple supervisory control and data acquisition system architecture.

In SCADA systems, the sensitive data will be transmitted among different parts of the power grid. Key management mechanisms can be used to protect the security of the data. Due to a self-healing property and low storage overhead requirements, our proposed E-SGKD scheme is suitable for achieving the key distribution and resolving the transmission availability and security in resource-constrained SCADA systems, where the sub-MTUs are as the GM and RTUs are as group manager nodes, which can efficiently achieve the key distribution and updating in SCADA systems.

8. Conclusions

In this paper, we proposed two E-SGKD schemes. The basic E-SGKD scheme was constructed from a known polynomial-based SGKD, and it has offered the optimal storage overhead while not having backward secrecy. The new E-SGKD scheme was constructed from the basic E-SGKD scheme. To consider the communication overhead and the backward secrecy, a novel approach is introduced for message broadcasting, which makes the new E-SGKD scheme obtain all basic security properties. Compared with known E-SGKD schemes, our new scheme has optimal storage overhead and low communication overhead. We discussed how to select the parameters and simulated it in the ZigBee network. Finally, we introduce the application of our proposed E-SGKD scheme to SCADA systems in the smart grid.

Acknowledgments: This work is supported by the National Natural Science Foundation of China (No. 61300172, 61572027) and High Technology Research and Development Program of China (No. 2015AA016004).

Author Contributions: Hua Guo, Yandong Zheng, Xiyong Zhang and Zhoujun Li designed the scheme and wrote the paper together.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Staddon, J.; Miner, S.; Franklin, M.; Balfanz, D.; Malkin, M.; Dean, D. Self-healing key distribution with revocation. In Proceedings of the 2002 IEEE Symposium on Security and Privacy, Berkeley, UK, 12–15 May 2002; pp. 241–257.
2. Blundo, C.; D'Arco, P.; Santis, A.; Listo, M. Design of Self-Healing Key Distribution Schemes. *Des. Codes Cryptogr.* **2004**, *32*, 15–44.

3. Blundo, C.; D'Arco, P.; Santis, A. Definitions and bounds for self-healing key distribution schemes. In Proceedings of the Fuzzy Sets and Systems, 31st International Colloquium on Automata, Languages and Programming ICALP 04, Turku, Finland, 12–16 July 2004; pp. 234–245.
4. Blundo, C.; D'Arco, P.; Santis, A. On Self-Healing Key Distribution Schemes. *IEEE Trans. Inf. Theory* **2006**, *52*, 5455–5467.
5. Dutta, R. Anti-collusive self-healing key distributions for wireless networks. *Int. J. Wirel. Mob. Comput.* **2014**, *7*, 362–377.
6. Chen, H.; Xie, L. Improved One-Way Hash Chain and Revocation Polynomial-Based Self-Healing Group Key Distribution Schemes in Resource-Constrained Wireless Networks. *Sensors* **2014**, *14*, 24358–24380.
7. Dutta, R.; Mukhopadhyay, S.; Collier, M. Computationally secure self-healing key distribution with revocation in wireless ad hoc networks. *Ad Hoc Netw.* **2010**, *8*, 597–613.
8. Dutta, R.; Mukhopadhyay, S.; Das, A.; Emmanuel, S. Generalized self-healing key distribution using vector space access structure. In Proceedings of the 7th International IFIP-TC6 Networking Conference on *Ad Hoc* and Sensor Networks, Singapore, 5–9 May 2008; pp. 612–623.
9. Saez, G. On threshold self-healing key distribution schemes. *Cryptogr. Coding* **2005**, *3796*, 340–354.
10. Tian, B.; Han, S.; Dillon, T.; Das, S. A self-healing key distribution scheme based on vector space secret sharing and one way hash chains. In Proceedings of the 2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks, Newport Beach, CA, USA, 23–26 June 2008; pp. 1–6.
11. Du, X.; Wang, Y.; Ge, J.; Wang, Y. An ID-based broadcast encryption scheme for key distribution. *IEEE Trans. Broadcast.* **2005**, *51*, 264–266.
12. Tian, B.; Han, S.; Dillon, T. A Self-Healing and Mutual-Healing Key Distribution Scheme Using Bilinear Pairings for Wireless Networks. In Proceedings of the IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC '08), Shanghai, China, 17–20 December 2008 ; pp. 208–215.
13. Han, S.; Tian, B.; Zhang, Y.; Hu, J. An effieient self-healing key distribution scheme with constant-size personal keys for wireless sensor networks. In Proceedings of the 2010 IEEE International Conference on Communications (ICC), Cape Town, South Africa, 23–27 May 2010 ; pp. 1–5.
14. Rams, T.; Pacyna, P. Self-healing group key distribution with extended revocation capability. In Proceedings of the 2013 International Conference on Computing, Networking and Communications (ICNC), San Diego, CA, USA, 28–31 January 2013; pp. 347–353.
15. Rams, T.; Pacyna, P. Long-Lived Self-Healing Group Key Distribution Scheme with Backward Secrecy. In Proceedings of the 2013 Conference on Networked Systems (NetSys), Stuttgart, Germany, 11–15 March 2013; pp. 59–65.
16. Tian, B.; Han, S.; Hu, J. A mutual-healing key distribution scheme in wireless sensor networks. *J. Netw. Comput. Appl.* **2011**, *34*, 80–88.
17. Liu, Y.N.; Mao, L.; Harn, L. Group key distribution with full-healing property. In Proceedings of the 2014 23rd International Conference on Computer Communication and Networks (ICCCN), Shanghai, China, 4–7 August 2014; pp. 1–6.
18. Liu, D.; Ning, P.; Sun, K. Efficient self-healing group key distribution with revocation capability. In Proceedings of the 10th ACM Conference on Computer and Communications Security, Washington, DC, USA, 27–30 October 2003; pp. 27–31.
19. More, S.M.; Malkin, M.; Staddon, J. Sliding-window self-healing key distribution. In Proceedings of the 2003 ACM workshop on Survivable and Self-Regenerative Systems: In Association with 10th ACM Conference on Computer and Communications Security, Fairfax, VA, USA, 31 October 2003.
20. Tian, B.; Han, S.; Parvin, S. Self-Healing Key Distribution Schemes for Wireless Networks. *Comput. J.* **2011**, *54*, 549–569.
21. Rams, T.; Pacyna, P. A Survey of Group Key Distribution Schemes With Self-Healing Property. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 820–842.
22. Hong, D.; Kang, J. An efficient key distribution scheme with self-healing property. *IEEE Commun. Lett.* **2005**, *9*, 759–761.
23. Song, H.; Tian, B.; He, M. Efficient threshold self-healing key distribution with sponsorization for infrastructureless wireless networks. *IEEE Trans. Wirel. Commun.* **2009**, *8*, 1876–1887.

24. Dutta, R.; Chang, E.; Mukhopadhyay, S. Constant storage self-healing key distribution with revocation in wireless sensor network. In Proceedings of the 2007 IEEE International Conference on Communications, Glasgow, UK, 24–28 June 2007; pp. 1323–1332.
25. Dutta, R.; Mukhopadhyay, S. Designing scalable self-healing key distribution schemes with revocation capability. *Parallel Distrib. Process. Appl.* **2007**, *4742*, 419–430.
26. Dutta, R.; Change, E.C.; Mukhopadhyay, S. Efficient self-healing key distribution with revocation for wireless sensor networks using one way key chains. *Appl. Cryptogr. Netw. Secur.* **2007**, *4521*, 385–400.
27. Guo, H.; Zheng, Y.; Wang, B.; Li, Z. A Note on an Improved Self-Healing Group Key Distribution Scheme. *Sensors* **2015**, *10*, 25033–25038.
28. Zou, X.; Dai, Y.S. A robust and stateless self-healing group key management scheme. *Commun. Technol.* **2006**, 1–4.
29. Tian, B.; Han, S.; Dillon, T. An efficient self-healing key distribution scheme. In Proceedings of the 2010 IEEE International Conference on Communications (ICC), Cape Town, South Africa, 23–27 May 2010; pp. 1–5.
30. Dutta, R.; Mukhopadhyay, S.; Emmanuel, S. Low bandwidth self-healing key distribution for broadcast encryption. In Proceedings of the 2008 Second Asia International Conference on Modelling & Simulation (AMS), Kuala Lumpur, Malaysia, 13–15 May 2008; pp. 867–872.
31. Piao, Y.; Kim, J.; Tariq, U.; Hong, M. Polynomial-based key management for secure intra-group and inter-group communication. *Comput. Math. Appl.* **2013**, *65*, 1300–1309.
32. Xu, Q.; He, M. Improved constant storage self-healing key distribution with revocation in wireless sensor network. *Inf. Secur. Appl.* **2009**, *5379*, 41–55.
33. Wang, Q.; Chen, H.; Xie, L.; Wang, K. Access-polynomial-based Self-healing Group Key Distribution Scheme for Resource-Constrained Wireless Networks. *Secur. Commun. Netw.* **2012**, *5*, 1363–1374.
34. Jiao, D.; Li, M.; Yu, Y. Self-Healing Key-Distribution Scheme with Collusion Attack Resistance Based on One-Way Key Chains and Secret Sharing in Wireless Sensor Networks. *Int. J. Distrib. Sens. Netw.* **2012**, *2012*, 283–294.
35. Sun, X.; Wu, X.; Huang, C. Modified access polynomial based self-healing key management schemes with broadcast authentication and enhanced collusion resistance in wireless sensor networks. *Ad Hoc Netw.* **2016**, *37*, 324–336.
36. Zheng, Y.; Guo, H. On the Security of a Self-Healing Group Key Distribution Scheme. Available online: http://eprint.iacr.org/2015/697.pdf (accessed on 25 April 2016).
37. Ogiela, M.R.; Ogiela, U. Shadow Generation Protocol in Linguistic Threshold Schemes. *Commun. Comput. Inf. Sci.* **2009**, *2009*, 35–42.
38. Gutierrez, J.A. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs). IEEE Standard for Information Technology 802.15.4., Institute of Electrical and Electronics, New York, NY, USA, 2003.
39. ZigBee Document 0949r00ZB, ZigBee RF4CE Specification. Version 09. 17 March 2009. Available online: http://www.zigbee.org/zigbee-for-developers/network-specifications/zigbeerf4ce/ (accessed on 27 April 2016).
40. Jiang, R.; Lu, R.; Luo, J. Efficient self-healing group key management with dynamic revocation and collusion resistance for SCADA in smart grid. *Secur. Commun. Netw.* **2015**, *8*, 1026–1039.