Efficient Threshold Self-Healing Key Distribution with Sponsorization for Infrastructureless Wireless Networks

Song Han, Member, IEEE, Biming Tian, Mingxing He, Member, IEEE, and Elizabeth Chang, Senior Member, IEEE

Abstract—Self-healing key distribution schemes are particularly useful when there is no network infrastructure or such infrastructure has been destroyed. A self-healing mechanism can allow group users to recover lost session keys and is therefore quite suitable for establishing group keys over an unreliable network, especially for infrastructureless wireless networks, where broadcast messages loss may occur frequently. An efficient threshold self-healing key distribution scheme with favorable properties is proposed in this paper. Firstly, the distance between two broadcasts used to recover the lost one is alterable according to network conditions. This alterable property can be used to shorten the length of the broadcast messages. Secondly, any more than threshold-value users can sponsor a new user to join the group for the subsequent sessions without any interaction with the group manager. Thirdly, the storage overhead of the self-healing key distribution at each group user is a polynomial over a finite field, which will not increase with the number of sessions. In addition, if a smaller group of users up to a threshold-value were revoked, the personal keys for non-revoked users can be reused.

Index Terms—Authentication, infrastructureless wireless network, ad hoc network, self-healing, key distribution, secret sharing, wireless sensor network.

I. INTRODUCTION

An infrastructureless network offers a means of addressing the needs for a more flexible, durable and cost efficient network system than conventional centralized hierarchical fixed infrastructure systems does. Infrastructureless wireless networks, especially mobile wireless ad hoc networks, are ideal candidates for communications in applications such as rescue missions, scientific explorations and even military operations. These potential applications highlight concerns regarding security issues. Theoretically, all key distribution schemes developed for reliable networks (e.g.,[1]-[4]) can be used in wireless networks with minor alternation. However, mobility changes the topology of networks frequently [30].

Due to mobility of nodes, traditional security models designed for fixed-network topologies may not be fully applicable in infrastructureless wireless networks. Fig. 1 presents an example of an infrastructureless wireless network. To better design an efficient and secure key distribution scheme, the designers should consider many factors such as application requirements, network topologies, and packet loss characteristics of the underlying wireless networks.

Wireless networks have a certain number of peculiarities [5]. First of all, there are no fixed infrastructures in wireless networks. The nodes should act independently from any centralized controller. Moreover, the nodes are battery powered and have limited computational capabilities and memory resources. It will reduce the availability of wireless devices to adopt some power-consuming techniques such as public key cryptography. Energy saving is an important system design criterion. That is why symmetric-key ciphers and hash functions have become the most favorable tools for protecting...
wireless communications. In addition, the topology can be highly dynamic, hampering the stability of the links and of the routes. Furthermore, nodes in mobile wireless networks (i.e., wireless sensor networks or mobile ad hoc networks) may move in and out of range frequently and even sometimes be completely separated from the network. Key distribution broadcast for a particular session might not reach a user as scheduled. Thus, techniques without fault-tolerance features cannot fully address the whole problem. Finally, security is difficult to be implemented because of the vulnerability of the wireless links and the limited physical protection of nodes.

Generally speaking, security in wireless networks has six challenges [6]:

1) Lack of fixed infrastructure: this means all operations are performed by nodes rather than centralized controllers.
2) Resource limitations on wireless devices: this implies that power-consuming methods are inherently infeasible.
3) Unknown network topology prior to deployment: this suggests that key pre-distribution schemes are a practical option.
4) Wireless nature of communications: this means the communication channels are unreliable and therefore broadcast packets loss may occur frequently.
5) Very large density of distribution of wireless nodes: this indicates that the security scheme must be scalable.
6) High risk of physical attacks to unattended nodes: this will introduce a threat to the entire network.

Moreover, in some deployment scenarios wireless nodes need to operate in an adversarial environment.

The research of distributing keys for wireless networks has received significant attention [1]-[4], [6]-[8], [12]-[22], [24]-[27], and [30]. The existing literature has focused on concrete aspects. One example is broadcast encryption which addresses the problem of sending an encrypted message to a large user group so that the message can only be decrypted by a dynamically changing privileged subset [9]-[11], [29], and [32]. The other example is key pre-distribution which settles the issue of unknown physical topology prior to deployment [12]-[14]. However, such literature assumes the underlying network is reliable. How to distribute session keys for wireless networks, in a manner that can be resistant to packet loss, is an issue that requires intensive examination.

In an unreliable network, the key distribution broadcast for a particular session might never reach a user. Requiring retransmission would contribute to the traffic on a network that might already be heavily burdened. Especially, when group size is large, such re-transmissions could potentially exhaust the group manager. In addition, in some high security environments, it is suggested that only sending essential messages reduces vulnerability. Hence, non-interactive key distribution solutions are not only favorable but also necessary.

Self-healing key distribution has been reported to be quite useful in several settings in which session keys need to be used for a short time-period [19] and [27], due to frequent changes in the group topology. Military-oriented applications as well as Internet applications [16], such as broadcast transmissions, pay-per-view TV, are a 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 some e-commerce situations, a service provider allows a number of service subscribers to pay for a new customer to use the same service for a limited period. If a key distribution scheme can realize that purpose and permits a coalition of users to sponsor a user outside the underlying group for one session, then the key distribution scheme is of sponsorship capability.

In this paper, we will propose an efficient self-healing key distribution scheme with sponsorship capability. The main contribution of this paper is highlighted by the following properties:

- The distance between two broadcasts which are used to recover the lost one can be set according to the underlying wireless networks. Working in this way facilitates a shorter length of the broadcast messages.
- \( t + 1 \) or more users of the group can sponsor a new user to join the group for subsequent sessions without any interaction with the group manager.
- The storage overhead of personal keys at each group user is a polynomial of \( F_p(n) \), which will not increase with the number of sessions.

This paper presents an analysis of security and efficiency. Findings performed here suggest that the proposed scheme outperforms other self-healing key distribution schemes in terms of the length of broadcasts, sponsorship, and storage overhead.

The rest of the paper is organized as follows: In Section II, we present an overview of existing works in the area of self-healing key distribution systems. In Section III, we introduce our system parameters followed by the security model and concrete construction. An analysis of security and efficiency of our proposed scheme compared with previous schemes are documented in Section IV. This paper concludes with possible future work. The notations in Table I are used throughout this paper.

II. RELATED WORKS

The central idea of self-healing key distribution schemes is that users, in large and dynamic group communications over an unreliable network, can recover lost session keys. The users can facilitate this recovery without requesting additional transmissions from the group manager even if some previous key distribution messages are lost. According to the technologies upon which they are based, self-healing key distribution schemes can be categorized into three classes:

- Polynomial secret sharing based self-healing key distribution schemes;
- Vector space secret sharing based self-healing key distribution schemes;
- Hash chain based self-healing key distribution schemes.

In this section, we review the related schemes according to the three categories. The characteristics of these schemes are summarized in subsection D of Section II.

A. Shamir’s secret sharing based self-healing key distribution schemes

The first pioneering work of self-healing key distribution schemes was introduced by Staddon et al. in [16]. Staddon et
al. proposed formal definitions, lower bounds on the resources and some constructions for a self-healing key distribution scheme. The main goal of the scheme is the self-healing property: if during a certain session some broadcasted packet gets lost, then users are still capable of recovering the group key for that session simply by using the packets they have received during a previous session and the packets they will receive at the beginning of a subsequent one, without requesting additional transmission from the group manager. From the time of Staddon et al.’s publication, self-healing key distribution schemes have become a hot research topic.

Liu, Ning and Sun in [17] generalized the self-healing key distribution definition in [16] and gave some schemes. Liu et al.’s scheme reduced communication overhead and storage overhead by introducing a novel personal key distribution technique. In addition, they developed two techniques that allow trade-off between the broadcast message size and the recoverability of lost session keys. The two methods further reduce the broadcast message size in situations where there are frequent but short-term disruptions of communication and where there are long-term but infrequent disruptions of communication, respectively.

Blundo et al. in [18] showed an attack that can be applied to the first construction in [16], presented a new mechanism for implementing the self-healing approach, extended the self-healing approach to key distribution, and proposed another key-recovery scheme which enabled each user to recover all lost session keys (for sessions in which he belongs to the group) by using only the current broadcast message. More et al. in [19] used a sliding window to address the three problems in [16]. The three problems were inconsistent robustness, high overhead and expensive maintenance costs. Dutta et al. in [22] developed a new self-healing key distribution scheme. The main emphasis of the scheme is that it has significant improvement in terms of both storage overhead and communication overhead. All of these papers mainly focused on unconditionally secure schemes, which are based on information theory [23].

By introducing an improved secret sharing scheme, Tian et al. in [25] proposed a self-healing key scheme with novel properties. 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 [16] and [18]. However, the efficiency improvements are obtained by relaxing the security slightly. The scheme is a computationally secure scheme.

The authors of this paper propose a threshold self-healing key distribution scheme with sponsorization. The scheme proposed belongs to the category of polynomial secret sharing based self-healing key distribution schemes and is therefore based on the idea of Shamir’s secret sharing scheme. The difference between the proposed scheme and Shamir’s secret sharing scheme is discussed at the end of this section.
B. Vector space secret sharing based self-healing key distribution schemes

Sáez in [20] first considered applying vector space secret sharing instead of Shamir’s secret sharing schemes to realize a self-healing key distribution scheme. The scheme made use of general monotone decreasing structures for the family of subsets of users that can be revoked instead of a threshold one. The length of broadcast was variable according to the condition of networks. Sáez in [21] considered the possibility that a coalition of users sponsor a user outside the group for one session. The formal definition, some bounds on the required amount of information, and general construction of a family of self-healing key distribution schemes with sponsorization by means of a linear secret sharing scheme was proposed. The particular case of this general construction when Shamir’s secret sharing scheme is used is analyzed at the end of this section. Subsubsection text here.

C. Hash chain based self-healing key distribution schemes

Bohio and Miri in [26] considered incorporating the self-healing feature to Subset Difference (SD) method, which was proposed by Naor et al. in [31]. Some optimization techniques that can be used to reduce the overhead caused by the self-healing capability are proposed in [26]. In addition, the idea of mutual self-healing was discussed. One motivation behind mutual self-healing is that, if a node has missed a key updating message, it does not have to wait until the next update broadcast to recover the previous session key. Instead, it can look for assistance from its neighboring nodes to recover that key instantly.

Jiang et al. in [27] proposed an efficient self-healing group key scheme with time-limited node revocation based on Dual Directional Hash Chains (DDHC). The performance of the proposed scheme under poor broadcast channel conditions is evaluated by both analysis and numerical results. The result shows that the scheme can tolerate high channel loss rate, and hence make good balance performance and security, which is suitable for wireless network applications.

D. Comparison for the three categories schemes

Shamir’s secret sharing is the most common technique used to realize self-healing key distribution. It performs easily. However, the maximum number of revoked users is constrained by the degree of the polynomial.

Vector space secret sharing based self-healing key distribution schemes consider a monotone decreasing family of rejected user subset instead of a monotone decreasing threshold structure. This general case makes the self-healing scheme more flexible and close to practical applications.

Both forward and backward secrecy can be kept by dual directional hash chains. However, the feature of resisting collusion of revoked nodes and new joined nodes can not be assured, due to the properties of one-way hash functions.

E. Difference between the proposed scheme and Shamir’s secret sharing scheme

Both the scheme proposed by the authors of this paper and Shamir’s secret sharing scheme are suited to applications in which a subgroup of users up to a threshold value may be compromised and a coalition of at least threshold-value users of the rest must cooperate in order to recover the secret key. Shamir’s secret sharing scheme has these properties:

1) to recover the original key given any subset of threshold-value secret pieces;
2) support Join and/or Leave operation;
3) any coalition of users up to the threshold value cannot recover the secret key.

Besides the properties of Shamir’s secret sharing scheme, our scheme has these additional properties:

1) Any \( t + 1 \) users of the group can sponsor a new user to join the group for subsequent sessions without any interaction with the group manager.
2) The distance between two broadcasts which are used to recover the lost one can be set according to the underlying wireless networks. By this way, a shorter length of the broadcast messages is achieved.
3) The storage overhead of personal keys at each group user is a polynomial over \( F_p \), which will not increase with the number of sessions.
4) Both forward security and backward security are kept in our scheme.
5) If some broadcasts get lost during a certain session, users can still recover the group key for that session simply by using the broadcasts they have received before that session and the broadcasts they will receive at the beginning of a subsequent one.

III. PROPOSED SELF-HEALING KEY DISTRIBUTION WITH SPONSORIZATION

This section details the authors proposed system parameters, security model and concrete construction.

A. System parameters

In the model proposed here, 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. We denote the set of users revoked by the group manager in session \( j \) by \( R_j \), and the set of users who join the group in session \( j \) by \( J_j \). Hence, \( G_j = (G_{j-1} \cup J_j) / R_j \) for \( j \geq 2 \). By definition, there is \( G_1 = U \). Each user \( U_i \in G_j \) holds a personal key \( S_i \in F_p \) 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. In particular, we assume that session keys \( K_j \in F_p \) are chosen independently and according to the uniform distribution. For \( U_i \in G_j \) and \( j = 1, \ldots, m \), the session key \( K_j \) can be determined by \( S_i \) and \( B_j \). \( K_j \) can also be computed by a user \( U_i \notin G_j \) sponsored by more than \( t \) users in \( G_j \) by means of \( B_j \) and sponsorization message.

B. Security model

The model proposed here is similar to the one given in [20]. To clarify our scheme, we provide the following formal
The scheme is a session key distribution scheme, meaning that:
(a) Every user $U_i \in G_j$, the key $K_j$ is determined by $S_i$ and $B_j$. Formally, it holds that:
$$H(K_j | B_j, S_i) = 0.$$  

(b) Keys $K_1, \ldots, K_m$ cannot be determined from the broadcast or personal keys alone. That is:
$$H(K_1, \ldots, K_m | B_1, \ldots, B_m) = H(K_1, \ldots, K_m | S_{G_1 \cup \ldots \cup G_m}) = H(K_1, \ldots, K_m) = 0.$$  

3) The scheme is self-healing. This means that the forward security and backward security separately.

(a) Every user $U_i \in G_j$, who has not been revoked after session $j_1$ and before session $j_2$ can recover all keys $K_{j_1}$ for $1 \leq j_1 < j_2 \leq m$ with $j_2 - j_1 \leq T$. Formally, it holds that:
$$H(K_{j_1}, \ldots, K_{j_2} | B_1, \ldots, B_m) = 0.$$  

(b) Let $\overline{R}_{j_1} \subseteq R_{j_1-1} \cup \ldots \cup R_2$ be a coalition of users joined before session $j_1$, where $|\overline{R}_{j_1}| \leq t$. Then, users in $\overline{R}_{j_1}$ cannot get any information about $K_{j_1}$, even with the knowledge of group keys before session $j_1$.
$$H(K_{j_1} | B_1, \ldots, B_{j_1-1}, S_{\overline{R}_{j_1}}, K_1, \ldots, K_{j_1-1}) = H(K_{j_1}).$$

Note that broadcast messages $B_1, \ldots, B_{j_1-1}$ are sufficient in the above equation (5). This is because users in $\overline{R}_{j_1}$ have been revoked before session $j_1$. Although they can get the broadcast messages $B_1, \ldots, B_m$ and after session $j_1$, the users in $\overline{R}_{j_1}$, however, will not be able to perform any Key Computation for $K_{j_1}, \ldots, K_m$. Furthermore, if the broadcast messages are encapsulated using Traffic Encryption Key [28], then users in $\overline{R}_{j_1}$ cannot get $B_1, \ldots, B_m$. Therefore, it is sufficient using $B_1, \ldots, B_{j_1-1}$ in Equation (5). A similar reason for Equation (6) below is held for being sufficient using $B_{j_2+1}, \ldots, B_m$.

(c) Let $\overline{J}_{j_2} \subseteq R_{j_2+1} \cup \ldots \cup J_m$ be a coalition of users join after session $j_2$, where $|\overline{J}_{j_2}| \leq t$. Then, users in $\overline{J}_{j_2}$ together cannot get any information about $K_{j_2}$, even with the knowledge of group keys after session $j_2$.
$$H(K_{j_2} | B_1, \ldots, B_m, S_{\overline{J}_{j_2}}, K_{j_2+1}, \ldots, K_m) = H(K_{j_2}).$$

4) The scheme has sponsorization capability. This means that the following properties are satisfied:
(a) Every user $U_i \in G_j$ can generate a proof of sponsorization $P^j_{li}$ to sponsor a user $U_i \not\in G_j$ for session $j$ using his personal key. In other words:
$$H(P^j_{li} | S_i) = 0.$$  

(b) A user $U_i \not\in G_j$ who receives more than sponsorizations from a subset of users in session $j$, together with the broadcast information, can compute the key $K_j (r \leq j \leq s)$. That is:
$$H(K_j | P^j_{A_1}, B_r, B_s) = 0.$$  

Condition 1) states that every user $U_i \in G_j$, from the broadcast and his own personal key, recovers the current session; while, personal keys and broadcasts alone, do not give any information about any session key. Condition 2) states that a collusion of $t$ or less revoked users does not give information about the current session key. The condition means GM is able to revoke users at most from the group. Condition 3) characterizes the self-healing property: any two broadcasts are enough to recover all lost session keys for the "sandwich" sessions. Condition 3) expresses the mechanism of sponsorization: the information used to sponsor is computed from the user $U_i$'s personal key $S_i$. Conditions 4) indicates the fact that the information obtained from enough sponsorization with corresponding broadcasts allows to compute the session key for each user.

C. Construction

We take a random one way permutation $f$ over $F_p$ such that $f^i(u) \neq f^j(u)$ for all positive integers $i$, $j$, and $u \in F_p$. $f^i(u)$ means the permutation $f$ is applied $i$ times on $u \in F_p$. The self-healing key distribution scheme with sponsorization abilities is composed of six procedures.

C.1. Setup

Suppose $G_1 = U_1, \ldots, U_n$, the corresponding identities of users in $G_1$ are $1, \ldots, n$, respectively. Let $t$ be a positive integer. The GM chooses at random a polynomial $s(x, y) = a_0 + a_1 x + a_2 y + \ldots + a_t x^t y^t$ from $F_p[x, y]$ and a random initial session identifier $sid_0 \in F_p$. GM sends $sid_0$ and personal key $s(i, y)$ to user $U_i$ ($i = 1, \ldots, n$) via secure communication channel. GM also selects randomly $T$ session keys $K_1, \ldots, K_T \in F_p$.

C.2. Broadcast

In the $j$-th ($j \geq 1$) session key distribution, given a set of all revoked users and its corresponding identifier set
$W_j = r^1_i, \ldots, r^j_{w_j}$ for sessions in and before session $j$ (where $r^1_i, \ldots, r^j_{w_j}$ are identifiers of revoked users and $|W_j| = \omega_j \leq t$; for each $r^j_{w_j}$, it is shared by the user $U_{r_j} \notin G_j$ and the GM, and made public by the GM), GM executes the following operations:

1) GM computes its $j$-th session identifier $sid_j = f(sid_{j-1})$.

2) GM constructs $P_j(x) = r_j(x)K_j + s(x, sid_j)$, where $r_j(x) = (x - r^1_i) \ldots (x - r^j_{w_j})$ is called revocation polynomial and $s(x, sid_j)$ is called a masking polynomial.

3) GM broadcasts message $B_j = B^1_j \cup B^2_j$. The first part of the broadcast is defined as follows:

$$B^1_j = \{ P_j(x) \{ Max(P_{j-T}(x), P_1(x), \ldots, P_j(x)) \} \}_{j=3, \ldots, m}$$

The second part of the broadcast is $B^2_j = W_j$.

C.3. Key computation

Any non-revoked user $U_i$ receives the broadcast message $B_j$. He first computes the session identifier $sid_j = f(sid_{j-1})$ and replaces the previous session identifier $sid_{j-1}$ by the current value $sid_j$ for $j \geq 1$. In case $j = 1$, $sid_1$ is stored. $U_i$ constructs $r_i(x)$ by $B^2_j$. Correspondingly, he computes $U_i$ the session key:

$$K_j = \frac{P_j(i) - s(i, sid_j)}{r_j(i)} \quad (9)$$

Note that from the set $B^2_j$ in the broadcast message $B_j$, all users $U_i$ can construct the polynomial $r_j(x)$ and consequently, can compute the value $r_j(i)$. In particular, for revoked users $U_i \in \bigcup_{j=1}^{l} W_j$, there exists $r_j(i) = 0$. Hence the revoked users can not recover the current session key from the broadcast message.

**Remark:** The most expensive computation overhead for $U_i$ is the one of computing $s(i, sid_j)$. One possible way is to compute a single point $\{ i, s(i, sid_j) \}$ at the $t$-degree polynomial $s(x, sid_j)$ per each session. In order to do this, $U_i$ gets $s(x, sid_j)$ from $s(x, y)$ with input $sid_j$. However, the coefficients in $s(x, sid_j)$ are not known to $U_i$. Therefore, it is impossible for her/him to calculate $i, s(i, sid_j)$ at the $t$-degree polynomial $s(x, sid_j)$. Another way is to evaluate a single point $sid_j, s(i, sid_j)$ at the $t$-degree polynomial $s(i, y)$. This is feasible. In fact, $U_i$ has personal key $s(i, y)$ which indicates the coefficients in $s(i, y)$ are known to $U_i$. What she/he needs to do is to take $y = sid_j$ as input for $s(i, y)$ over $F_p$. It takes $O((logt)(log^2p))$ bit operations to get $s(i, sid_j)$.

C.4. Self-healing

Let $U_i$ be a user that receives session key distribution message $B_j$, in session $j$, and $B_j_2$ in session $j_2$, respectively, but not the message $B_j$ for session $j$, where $1 \leq j < j_2 \leq m$ and $j_2 - j_1 \leq T$. User $U_i$ can still recover all the lost session keys $K_j$ for $j_1 < j < j_2$ as follows:

1) In $j_1$-th session, $U_i$ first computes $sid_{j_1} = f^{j_1}(sid_0)$, $r_{j_1}(i)$ and $P_{j_1}(i)$ from the broadcast message $B_{j_1}$. Then $U_i$ recovers the $j_1$-th session key

$$K_{j_1} = \frac{P_{j_1}(i) - s(i, sid_{j_1})}{r_{j_1}(i)} \quad (10)$$

2) In $j_2$-th session, $U_i$ first computes $sid_{j_2} = f^{j_2}(sid_0)$, $r_{j_2}(i)$ and $P_{j_2}(i)$ from the broadcast message $B_{j_2}$. Then $U_i$ recovers the $j_2$-th session key

$$K_{j_2} = \frac{P_{j_2}(i) - s(i, sid_{j_2})}{r_{j_2}(i)} \quad (11)$$

3) For $j = j_1 + 1, \ldots, j_2 - 1$, $U_i$ first computes $sid_j = f^{j}(sid_{j-1})$. He recovers $P_j(i)$ and $r_j(i)$ from the broadcast messages in turn. Finally $U_i$ recovers the $j$-th session key

$$K_j = \frac{P_j(i) - s(i, sid_j)}{r_j(i)} \quad (12)$$

**Remark:** How to recover $r_j(i)$ from the broadcast messages? In fact, notice that the procedure in the Broadcast enables $U_i$ to get broadcast message $B_j = B^1_j \cup B^2_j$. The first part of the broadcast is defined as follows:

$$B^1_j = \{ P_j(x) \{ Max(P_{j-T}(x), P_1(x), \ldots, P_j(x)) \} \}_{j=3, \ldots, m}$$

The second part of the broadcast is $B^2_j = W_j$. Therefore, $U_i$ can get $\{ r^1_j, \ldots, r^j_{w_j} \}$ which is in fact $B^2_j$. By the definition of $r_j(x)$, $U_i$ can recover $r_j(i)$ as follows:

$$r_j(i) = (i - r^1_j)(i - r^2_j)\ldots(i - r^j_{w_j}) \quad (13)$$

C.5. Add and revoke group users

When GM wants to add a user $U_i'(i' \neq 1, \ldots, n)$ starting from session $j$, he sends personal key $s(i', y)$ and $sid_j$ via secure communication channel between them. If a user $U_i$ is revoked in session $j$, his identity $i$ must be included in the second part of the broadcast message in the following sessions. In particular, once a user $U_i$ is revoked, he must be revoked in all the future sessions.

**Remark:** The scheme requires that once a certain user $U_i$ is revoked in session $j$, then he must be revoked in all the future sessions. Otherwise, the revoked user in session $j$ rejoins the group in a later session can recover the key for session $j$ due to the self-healing capability of the scheme.

C.6. Sponsorization

If a user $U_i \in G_j$ wants to sponsor a user $U_i \notin G_j$ for session $j$, then he computes $(l, s(l, sid_j))$ from his personal key $s(l, y)$ and sends it privately to $U_i$. $U_i$ can compute $s(x, sid_j)$ after receiving $t + 1$ sponsored messages from user subset $A \subseteq G_j$ (where $t + 1 \leq |A| \leq |G_j| \leq n$). Therefore, he can compute $s(i, sid_j)$. According to broadcast message $B_j$, he computes $P_j(i)$ and $r_j(i)$. Consequently, the user $U_i$ can compute the session key as:

$$K_j = \frac{P_j(i) - s(i, sid_j)}{r_j(i)} \quad (14)$$

The above self-healing key distribution process has been illustrated in Fig.2. Note that the exact order of these procedures is slightly different from the ones appeared in Fig.2. This is because Add/Revoke Operation may not take place and Broadcast message loss may not occur.
IV. ANALYSIS OF PERFORMANCE

This section begins with our analysis of the security of the proposed scheme according to the Definition 4.1. This is followed by a discussion of implicit authentication of session keys, and a possible solution to the secure distribution and veriﬁcation for broadcast messages. This section concludes with an analysis of the efﬁciency of the proposed scheme in terms of storage overhead and communication overhead. To clarify the advantages of our scheme, a performance comparison of our scheme with some existing schemes is presented.

A. Analysis of security

Findings from an analysis of security of the scheme in relation to Definition 4.1 suggest that the scheme facilitates a threshold self-healing key distribution with sponsorization capability.

1) Our scheme satisﬁes Condition 1) in Definition 4.1; therefore, it is a session key distribution scheme.

(a) Session key recovery by a user is described in Key computation phase of the construction.

(b) On the one hand, since the session keys are chosen according to the uniform distribution and independence of the personal keys, it is straightforward to see that the personal keys alone do not give any information about any session key. On the other hand, it is not difﬁcult to see that every $P_j(x)$, for $j = 1, \ldots, m$, perfectly hides key $K_j$ because $P_j(x) = r_j(x)K_j + s(x, sid_j)$. The set of session keys cannot be determined only by broadcast messages.

2) Our scheme satisﬁes Condition 2) in Definition 4.1, therefore, it has t-revocation capability.

Suppose that a collection $R$ of $t$ revoked group members in session $j$ collude. In order to recover the session key $K_j$ from the broadcast, revoked users in $R$ must compute $r_j(i)$. However, for all revoked users $U_i$, exists $r_j(i) = 0$. Therefore, $K_j$ is completely safe.

3) Our scheme satisﬁes Condition 3) in Definition 4.1, therefore, it is self-healing. The maximum number of session keys that users can recover one time is $T$.

(a) For any user $U_i$ who is a user in session $j_1$ and $j_2(1 \leq j_1 < j_2 \leq m)$, by the method of key computation step in self-healing phase, $U_i$ can subsequently recover the whole sequence of session keys $K_{j_1}, \ldots, K_{j_2}$. In fact, in our construction, a qualiﬁed user can recover all the session keys before session $j_2$. This is a stronger self-healing scheme.

(b) For any user $U_i$ in set $R_{j_1} \subseteq R_{j_1-1} \cup \ldots \cup R_{j_2}$, where $|R_{j_1}| \leq t$, exists $r_j(i) = 0$ $(1 \leq j_1 < j_2 \leq m)$. Session keys are chosen at random and according to uniform, even with the knowledge of group keys before session $j_1$, $U_i$ can not get any information about the current session key $K_j$. Therefore, the backward security is kept.

(c) For any user $U_i$ in set $\overline{R_{j_1}} \subseteq R_{j_1+1} \cup \ldots \cup R_{j_2}$, where $|\overline{R_{j_1}}| \leq t$, the users in $\overline{R_{j_1}}$ together cannot obtain $sid_j(j_1 \leq j \leq j_2)$, even with the knowledge of group keys after session $j_2$. Therefore, $U_i$ can not compute the session key $K_j$, thus the forward security is kept.

4) Our scheme satisﬁes Condition 4) in Definition 4.1, therefore, it has sponsorization capability.

(a) Every user $U_i \in G_j$ can generate a proof of sponsorization $P_i^j$ to sponsor a user $U_i \notin G_j$ for session $j$ using his personal key.

(b) Seen from Sponsorization phase, the user $U_i \notin G_j$ that receives more than $t + 1$ sponsorizations from a subset of users in $G_j$ can recover the session key $K_j$.

Theorem 1. Given user $U_i \notin G_j$ who receives more than $t + 1$ sponsorizations from a subset of users in $G_j$, the session key $K_j$ generated by $U_i$ satisfies the existence and the uniqueness.

Proof: Note that

$$s(x, y) = a_{0,0} + a_{1,0}x + a_{0,1}y + \ldots + a_{t,t}x^ty^t$$  \hspace{1cm} (15)

Then we have

$$s(x, sid_j) = a_{0,0} + a_{1,1}sid_j + a_{1,0}x + \ldots + (sid_j)^t a_{t,t}x^t$$  \hspace{1cm} (16)

which is a $t$-degree polynomial with variable $x$. We rewrite it to get

$$s(x, sid_j) = b_0 + b_1x + \ldots + b_t x^t$$  \hspace{1cm} (17)

Suppose there are $(t+1)$ users $U_{i_l}, U_{l_2}, \ldots, U_{l_{t+1}}$ in $G_j$ to sponsor user $U_i \notin G_j$, then $U_i$ uses her/his personal key $s(l_{i'}, y)$ to compute

$$(l_{i'}, s(l_{i'}, y)),$$

where $1 \leq i' \leq t + 1$. These $(t + 1)$ users, respectively, send privately

$$(l_{i'}, s(l_{i'}, y))(1 \leq i' \leq t + 1)$$

to user $U_i \notin G_j$. After receiving these value-pairs, $U_i$ does the substitution using $(l_{i'}, s(l_{i'}, y))$ for the Equation (17) and gets the following system of linear equations:
can compute a unique $H$.

B. Implicit authentication of session keys

Theorem 1 shown that the above calculations are executed over $F_p$. This is a Vandermonde determinant. If and only if $l_{i1} \neq l_{j2}$, there is $D \neq 0$.

In fact, $U_{i1}, U_{i2}, \ldots, U_{i_{t+1}}$ are different users in $G_j$ and thus their IDs are different, i.e. $l_{i1} \neq l_{i2}$.

Hence, the system of linear equations in (18) has a solution $\{b_0, b_1, \ldots, b_l\}$ and also this solution is unique. Therefore, $s(x, sid_j) = b_0 + b_1 x + \ldots + b_l x^l$ is uniquely determined by $\{b_0, b_1, \ldots, b_l\}$. Therefore, user $U_{i}$ can compute a unique $s(i, sid_j)$ with $x = i$ for Equation (16). By the broadcast message $B_j$, he/she computes $P_j(x)$ and $r_j(i)$. Finally, $U_i$ gets a unique $j$-th session key

$$K_j = \frac{P_j(i) - s(i, sid_j)}{r_j(i)}.$$  

Notice that the above calculations are executed over $F_p$. Therefore, the $j$-th session key $K_j$ exists and is unique. Thus the Theorem 1 is concluded.

B. Implicit authentication of session keys

The session key generated in the self-healing key distribution scheme is implicitly authenticated. This can be demonstrated from four aspects:

1) Any external attacker cannot generate the session key. An external attacker is an adversary who is neither a current user of the group nor a revoked user from the group. Therefore, this attacker does not have the initial session identifier $sid_0 \in F_p$ or personal key $s(i, y)$ to user $U_i (i = 0, \ldots, n)$. Although this attacker can intercept some broadcast messages, he is not yet able to generate the session key

$$K_j = \frac{P_j(i) - s(i, sid_j)}{r_j(i)},$$

which needs two secret elements $sid_0 \in F_p$ and $s(i, y)$.

2) Any internal user who has been revoked cannot generate a new session key.

For all revoked users $U_i$, $r_j(i) = 0$. Therefore, any revoked user cannot generate a new session key.

3) Any internal valid users up to $t + 1$ cannot generate the session key. This property is obviously held since our scheme is a $(t + 1)$-threshold key distribution scheme.

4) Any internal valid user $U_i$ can generate the session key of any session the user belongs to. An internal valid user $U_i$ is a user who has not been revoked in the underlying communication group. Suppose $U_i$ belongs to the session group $G_j$. Hence, $U_i$ can receive the broadcast message $B_j$ from GM. $U_i$ first computes the session identifier $sid_j = f(sid_{j-1})$ and then gets $s(i, sid_j)$ by her/his personal key $s(i, y)$. Notice that $B_j = B_j^1 \cup B_j^2$. Hence, $U_i$ can compute $P_j(i)$ from $B_j^1$ and constructs $r_j(i)$. Finally, can compute the session key

$$K_j = \frac{P_j(i) - s(i, sid_j)}{r_j(i)}.$$  

C. Possible solution to secure distribution and verification of broadcast messages

In Section III, we have focused on the exact construction of the threshold self-healing key distribution with sponsorization capability by Definition 4.1. Therefore, we do not discuss the secure distribution and verification for broadcast messages. However, the secure distribution and verification for broadcast messages are useful to the correctness and availability of a group key distribution scheme. We thus present a possible solution to this issue.

An efficient solution to the secure distribution and verification for broadcast messages is to use a Traffic Encryption Key (TEK) to encrypt broadcast messages [28] by symmetric encryption. In fact the scheme in [27] used a TEK to distribute broadcast messages and achieved implicit authentication for broadcast messages.

Here we reconstruct Broadcast and Key Computation in our scheme to demonstrate how the secure distribution and verification for broadcast message $B_j = B_j^1 \cup B_j^2$ work.

C.2. New 1. Broadcast

In the $j$-th ($j \geq 1$) session key distribution, given a set of identifiers of all revoked users $W_j = \{r_{1j}, \ldots, r_{wj}\}$ for revoked users and $|W_j| = w_j \leq t$, GM executes the following operations:

1) GM computes its $j$-th session identifier $sid_j = f(sid_{j-1})$.

2) GM constructs $P_j(x) = r_j(x)K_j + s(x, sid_j)$, where $r_j(x) = (x - r_{1j}) \ldots (x - r_{wj})$ is called a revocation polynomial and $s(x, sid_j)$ is called a masking polynomial.

3) GM constructs broadcast message $B_j = B_j^1 \cup B_j^2$: where

$$B_j^1 = \{ P_j(x) \} \quad \text{and} \quad B_j^2 = W_j, \quad j = 1, 2, \ldots, m.$$
4) GM computes a Traffic Encryption Key

\[ TEK_j = g(B_{j-1}^1, K_{j-1}, B_{j-1}^2). \]

Here we prefer to use \( g(B_{j-1}^1, K_{j-1}, B_{j-1}^2) \) but not \( g(B_j, K_{j-1}) \). This is because \( B_j \) is not a suitable format for encryption. We also denote a ciphertext on broadcast message \( B_j \) by \( E_{TEK_j}(B_j, sid_j) \).

5) GM finally broadcasts

\[ \{E_{TEK_1(i)}(B_{j}^1(i), sid_j), E_{TEK_2}(B_{j}^1(2), sid_j), \ldots, E_{TEK_n(1)}(B_{j}^1(n), sid_j), B_{j}^2\} \]

to group users.

C.3.new1 Key Computation

After receiving broadcast message \( E_{TEK_i}(B_{j}^1, sid_j) \) and \( B_{j}^2 \), any non-revoked user \( U_i \) obtains the session key \( K_j \) by carrying out the operations below:

1) \( U_i \) uses the session key \( K_{j-1} \) and broadcast messages \( B_{j-1}^1(i) \) and \( B_{j-1}^2(i) \) to derive the new Traffic Encryption Key \( TEK_j(i) = g(B_{j-1}^1(i), K_{j-1}, B_{j-1}^2). \) Note that \( U_i \) got \( K_{j-1}, B_{j-1}^1(i), \) and \( B_{j-1}^2(i) \) in the \((j-1)\)-th session.

2) \( U_i \) uses \( TEK_j(i) \) to decrypt \( E_{TEK_1(i)}(B_{j}^1(i), sid_j) \) to get \( B_{j}^1(i) \) and \( sid_j \).

3) If \( sid_j = f(sid_{j-1}) \), then \( U_i \) accepts \( B_{j}^1(i) = B_{j}^1(i) \cup B_{j}^2(i) \) as a valid broadcast message; Otherwise, \( U_i \) discards the broadcast message.

4) \( U_i \) constructs the revocation polynomial \( r_j(x) \) by \( B_{j}^2 \).

He then computes \( r_j(i), s(i, sid_j) \) and \( P_j(i) \).

5) \( U_i \) then computes the session key \( K_j = \frac{P_j(i) - s(i, sid_j)}{r_j(i)} \).

6) \( U_i \) computes the Traffic Encryption Key

\[ TEK_j(i) = g(B_{j}^1(i), K_j, B_{j}^2) \]

for \((j+1)\)-th session by \( B_{j}^1, B_{j}^2 \) and \( K_j \).

The above new Broadcast and Key Computation procedures provide secure distribution and verification for broadcast messages. This is because the broadcast message \( B_j \) has been encrypted using a symmetric encryption function. This result in authorized access to broadcast messages is only granted to users who have the corresponding Traffic Encryption Keys. On the other hand, the verification of \( sid_j = f(sid_{j-1}) \) implies a verification on broadcast message \( B_j = B_{j}^1 \cup B_{j}^2 \). This is because if \( sid_j \neq f(sid_{j-1}) \), then the received ciphertext is not a correct one. This is an implicit authentication for this broadcast message.

The above new procedures C.2.new1 Broadcast and C.3.new1 Key Computation can ensure the secure distribution and verification for broadcast messages but introduces communication and computation overheads. In the following, we will present another method which can not only ensure the secure distribution and verification for broadcast messages but also keeps communication overhead at a constant. In addition, the new method also has less computation overhead than the one which is described above. The following procedures show how this new method works:

C.2.new2. Broadcast

In the \( j \)-th \((j \geq 1)\) session key distribution, given a set of identifiers of all revoked users \( W_j = \{r^1_{1w}, \ldots, r^1_{w_j}\} \) for sessions in and before session \( j \) (where \( r^1_{1w}, \ldots, r^1_{w_j} \) are identifiers of revoked users and \( |W_j| = w_j \leq l \)), GM executes the following operations:

1) GM computes its \((j+1)\)-th session identifier \( sid_j = f(sid_{j-1}) \).

2) GM constructs \( P_j(x) = r_j(x)K_j + s(x, sid_j) \), where \( r_j(x) = (x - r^1_{1w})(x - r^1_{2w}) \ldots (x - r^1_{w_j}) \) is called a revocation polynomial and \( s(x, sid_j) \) is called a masking polynomial.

3) GM constructs broadcast message \( B_j = B_{1}^1 \cup B_{2}^2 \); where

\[ B_{1}^1 = \left\{ P_j(x) \{Max(P_{j-T}(x), P_1(x)), \ldots, P_j(x)\} \right\} \]

\( j = 1, 2 \)

\[ j = 3, \ldots, m \]

\( and B_{2}^2 = W_j. \)

4) GM computes a Traffic Encryption Key

\[ TEK_j = g(sid_{j-1}) \]

and a ciphertext \( E_{TEK_j}(B_j, sid_j) \) on broadcast message \( B_j \).

5) GM finally broadcasts \( E_{TEK_j}(B_j, sid_j) \) to group users.

Remark: In the above procedure, those revoked users can also compute Traffic Encryption Key \( TEK_j = g(sid_{j-1}) \) since they have \( sid_{j-1} \). Therefore, they can decrypt the encrypted broadcast message \( E_{TEK_j}(B_j, sid_j) \). However, those revoked users cannot work out the new session key \( K_j \). This is because \( r_j(x) = 0 \) for those revoked users.

C.3.new2 Key Computation

After receiving broadcast message \( E_{TEK_i}(B_{j}^1, sid_j) \), any non-revoked user \( U_i \) obtains the session key \( K_j \) by carrying out the operations below:

1) \( U_i \) uses \( sid_{j-1} \) to derive the new Traffic Encryption Key \( TEK_j(i) = g(sid_{j-1}) \). Note that \( U_i \) got \( sid_{j-1} \) in the \((j-1)\)-th session.

2) \( U_i \) uses \( TEK_j(i) \) to decrypt \( E_{TEK_i}(B_{j}^1, sid_j) \) to get \( B_{j}^1(i) \) and \( sid_j \).

3) If \( sid_j = f(sid_{j-1}) \), then \( U_i \) accepts \( B_{j}^1(i) = B_{j}^1(i) \cup B_{j}^2 \) as a valid broadcast message; Otherwise, \( U_i \) discards the broadcast message.

4) \( U_i \) constructs the revocation polynomial \( r_j(x) \) by \( B_{j}^2 \).

He then computes \( r_j(i), s(i, sid_j) \) and \( P_j(i) \).

5) \( U_i \) then computes the session key \( K_j = \frac{P_j(i) - s(i, sid_j)}{r_j(i)} \).

6) \( U_i \) computes the Traffic Encryption Key

\[ TEK_{j+1}(i) = g(sid_j) \]

for \((j+1)\)-th session by \( sid_j \).

Remark: In the above procedure Key Computation, if a user \( U_i \in G_j \) but \( U_i \notin G_{j-1} \), then \( U_i \) himself/herself cannot get \( K_{j-1} \) and therefore cannot recover the Traffic Encryption Key \( TEK_j(i) \). Our solution to this situation is: \( U_i \) asks her/his nearest neighbour \( U_i \in G_j \) to privately transfer Key \( TEK_j(l) \) to him/her. \( U_i \) can use Key \( TEK_j(l) \) to decrypt the encrypted broadcast message \( E_{TEK_j(l)}(B_j, sid_j) \) and get \( B_j \). Finally, he/she can recover the session key \( K_j \).
D. Analysis of efficiency

In this section we make a performance comparison between some existing self-healing key distribution schemes and our scheme.

In terms of storage overhead, our scheme requires that each user stores a personal key of size \((t+1)\log p\) in each session. It comes from the procedure of Setup and after receiving the session key distribution broadcast. In the procedure of Setup, each user stores the initial session identifier \(sid_0\) and a \(t\)-degree polynomial as his personal key. After receiving the session key distribution broadcast, each user stores the \(j\)-th session identifier \(sid_j\). All of these elements are taken from . Moreover, the maximum number of sessions \(m\) is no longer needed to be determined in the procedure of Setup. Consequently, our scheme eliminates the limitation of \(m\) sessions in previous works [16]-[18] and [21]. The storage overhead of the schemes in [16]-[18] and [21] increases with \(j\) and the maximum number of sessions should be determined in the procedure of Setup.

The broadcast message \(B_j\) for the \(j\)-th session consists of a set of revoked users \(W_j\) and \(T\) \(t\)-degree polynomials. Since the user identities can be selected over a small finite field [24], one can ignore the communication overhead brought by the broadcast message of the set \(W_j\). Thus the size of the broadcast message in the \(j\)-th session is \(Tt\log p\).

To clarify the performance of the proposed scheme, a comparison of our scheme and some existing schemes is presented. TABLE II summaries the comparison of storage and communication overheads between these five self-healing key distribution schemes. The reason that we choose the schemes in [16] - [18] and [21] for performance comparison is: the selected schemes in [16] - [18] are all based on Shamir’s secret sharing scheme; and the scheme in [21] is the first scheme which has sponsorization capability. Consider that our scheme is based on Shamir’s secret sharing scheme and also has strong sponsorization capability, we therefore have selected those schemes for performance comparison.

We use \(C\) to denote Construction and \(S\) to denote Scheme, for example, \(C3\) in [16] denote the Construction 3 in [16]. In previous schemes, storage overhead increases with the order of session . Without loss of generality, we set \(j = m/2\) and we set \(T = m/3\). In addition, we assume that \(p\) is a 64-bit integer. Suppose the maximum number of session \(m = 100\) and the number of revoked users varies from 0 to 100.

Fig.3 and Fig.4 illustrate the increasing tendency of broadcasts for the five self-healing key distribution schemes. It is easy to see our scheme has less communication overhead than the previous schemes have for the same value of \(m\) and \(t\). In addition, when \(t\) climbs to 100, the size of broadcast in our scheme is approximately of 20kB while that of other schemes exceeds 80kB. One of the predominant characteristics of wireless networks is they have very limited resources. The proposed scheme makes use of smaller packets and is therefore quite suitable for real applications where available resources are very limited. Fig.5 displays the tradeoff between \(m\) and \(t\) given a maximum of 64kB packet size. When \(t\) increases to 100, the maximum number of sessions in our scheme is still around 250 while that of others is far less than 250.

Obviously, our scheme allows more sessions and can deal with more revoking users under the same conditions.

V. CONCLUSION

In this paper the authors proposed a new self-healing key distribution scheme. The proposed scheme has sponsorization capability and enables a large and dynamic group of users to establish a session key for secure communications over an unreliable wireless network. In order to shorten the length of broadcast messages, the distance between two broadcasts used to recover the lost one is adjustable in our scheme. The scheme also enables a user to recover, from a single broadcast message, \(T\) keys associated with the sessions in which she/he belongs to the group. The storage overhead of personal keys is a polynomial over \(F_p\), which will not increase with the number of sessions. The proposed scheme has been comprehensively analyzed in an appropriate security model to prove that it is secure and self-healing and also achieves both forward security and backward security.

The fact that new personal keys are needed with every session presents an interesting problem in previous schemes [16]-[21]. The long-lived personal key schemes are provided in [16] and [18]. However, they do not properly fit for practical applications where the cost of modular exponentiations
involved may be prohibitive due to the very limited communication resources. It is claimed that in [22] the personal key can be reused without any alternation. In fact, the personal keys can be re-used on condition that less than \( t + 1 \) users are revoked. Therefore, it is interesting to further explore the reusable personal keys.

The proposed scheme similarizes ideas to those found in [21]. That is, a coalition composed of more than \( t + 1 \) users in our scheme can sponsor a new user to join the group for one session without any interaction with the group manager. Different from the sponsorization capability in [21], the one in our scheme can enable a new user to recover all session keys in subsequent sessions for the group communications once she or he is sponsored and joins the group. This is a stronger sponsorization capability. Furthermore, our scheme is much efficient than that of [21] with respect to storage overhead and communication overhead respectively. In addition, our scheme has reserved forward security and backward security, which are imperative properties for group key distributions.

**ACKNOWLEDGMENT**

The authors sincerely thank the Editor for allocating qualified and valuable referees. The authors sincerely thank the anonymous referees for their very valuable comments. This research is partially supported by Curtin University of Technology grant and Australian Research Council grant. This research is also partially supported by the National Natural Science Foundation of China grant.

**REFERENCES**


Song Han is an academic at Digital Ecosystems and Business Intelligence Institute, Curtin University of Technology. He holds a PhD in information security at Peking University. His current research interests include authentication, key management, wireless network security, RFID security and information hiding. Since July 2005, Song Han has been a faculty with Curtin Business School, Curtin University of Technology. In 2006, he received the Emerald Literati Network award. In 2007, he received the New Researcher of the Year award from Curtin Business School. Curtin University of Technology. Song has served the Program Committee for a number of international conferences including some IEEE conferences such as the IEEE International Conference on Communications Systems, the IEEE/IFIP International Symposium on Trust, Security and Privacy for Pervasive Applications, the IEEE International Conference on Industrial Technology, the IEEE International Conference on Digital Ecosystems and Technologies, and the IEEE/IFIP International Conference on New Technologies, Mobility and Security. He is an IEEE member and an ACM member.

Biming Tian is currently working as a visiting research assistant at DEBI Institute, Curtin University of Technology, Australia. She received the B.S. degree in Computer Science and Technology in 2004 and the M.S. degree in Information Security and Cryptography in 2007. Her research interests include broadcast encryption, key pre-distribution and key management in wireless networks.

Mingxing He is a Professor of the School of Mathematics and Computer Engineering, Xihua University, Chengdu, China. His current research interests include cryptography and information security. He has published over 50 papers in refereed professional journals and international conferences. He received the DAAD scholarship reward of Germany in 2002, the Excellent PhD Dissertation Award in Southwest Jiaotong University in 2003, the grant of National Science Foundation of China (NSFC) in 2004 and 2007 respectively. He is included in reviewer lists of several technical journals such as IEEETRANSACTIONS ON INFORMATION THEORY and IEEE COMMUNICATION LETTERS. He is one of the ACM/ICPC Asia regional advisors and a member of the International Association for Cryptologic Research (IACR).

Elizabeth Chang is a Professor in IT and Director of the Research Institute for Digital Ecosystems and Business Intelligence (DEBI Institute). She is also a Director for the Research Centre of Extended Enterprise and Director for Area of Research Excellence (AoRE) on Frontier Technologies for E-Enterprises at Curtin Business School. She has been awarded the Vice Chancellor’s Outstanding Performance Award for 2005 and the Dean’s Best Researcher of Year Award for 2005 and 2004. She has co-authored three books and has published over 350 scientific papers as book chapters, in international journals and at refereed conferences as well as numerous invited keynote papers and tutorials. These also include ACM and IEEE transactions papers which are the top journals in the field. She is currently holding six ARC large Discovery and Linkage grants and a Tier I Centre of Excellence grant and obtained cash from ARC, industry partners as well as Research Centre of Excellence funds of over five million for 2002-2011. All her research and development work has been in the areas of IT Applications for Business. She has supervised or co-supervised 15 PhD graduates to completion in the last six years. Her research interests include user-interface analysis and design, usability evaluation, plug-and-play component-based system, Internet computing, including XML systems, security and privacy, trust and reputation, and ontologies. She is a Senior Member of IEEE and member of the ACM and the Australian Computer Society.