Multiple Key Sharing and Distribution Scheme With (n, t) Threshold for NEMO Group Communications

Yinxin Jiang, Chuang Lin, *Senior Member, IEEE*, Minghui Shi, and Xuemin (Sherman) Shen, *Senior Member, IEEE*

***Abstract*—In this paper, a novel secure key sharing and distribution scheme for network mobility (NEMO) group communications is proposed. The scheme offers the capability of multiple key sharing and distribution for current and future application scenarios, and a threshold mechanism that effectively improves flexibility and robustness of the key sharing and distribution process. Both forward and backward secrecy are guaranteed by compulsive key refreshment and automatic key refreshment mechanisms, which provide dynamic in-progress group communication joining/leaving and periodic keys renewal, respectively. Security and performance analysis are presented to demonstrate that the proposed scheme meets the special security requirements for NEMO group communications and is competent for key sharing and distribution service.**

***Index Terms*—Forward and backward secrecy, key distribution and management, network mobility (NEMO) group communications, threshold mechanism.**

I. Introduction

GROUP communication, as a growing application area in mobile communications, is a synchronous collaboration session, in which members at remote locations cooperate with an interactive procedure. Network mobility (NEMO) considers managing the mobility of an entire network, which is assumed to be a leaf network capable of changing its attachment to the Internet. Group communications in NEMO environment offer wide variety of mobile applications, such as board meeting, task force, field military meeting, and mobile entertainment. As shown in Fig. 1, a NEMO group communication session usually involves three parties: 1) certificate authority (CA), which is a trusted third party responsible for issuing secret shadow certificate to group communication members; 2) members, which include standalone mobile users (members A and C), entities in the NEMO leafs (Member B), and mobile routers of NEMO leafs; one member is a chairperson who initiates the group communication session; 3) Network Center (NC), which is a mobile communication center in charge of processing the messages from members, reconstructing or renewing keys, and distributing them to members. When a chairperson holds the NEMO group communication session, all members are required to connect to NC via wireless/wired network.

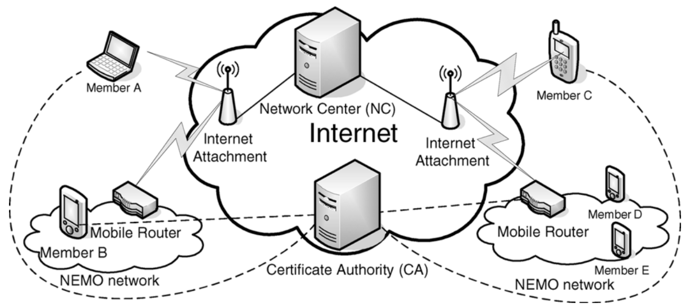


Fig. 1. NEMO group communications architecture.

NEMO group communications are developed from traditional teleconference with considerations of unique NEMO network characteristics. Inheriting all the common security weaknesses in wireless communication, such as eavesdropping and man-in-the-middle attack, a NEMO network changes the attachment to the Internet, where the unreliable wireless connection affects all the nodes within the NEMO network leaf instead of single node. Thus, it is required to provide confidentiality and authenticity with high robustness in NEMO group communications to prevent various intrusions, such as impersonating attack, conversation content eavesdropping, and mobility information tracking of a mobile user [1]. Specifically, five security requirements for group communications are required [2]–[4]: 1) conversation privacy during the group communication session; 2) member identity anonymity to protect a legal member's identity, location, and mobility information from tracking; 3) prevention of fraud by providing mutual authentication mechanism between NC and mobile members; 4) prevention of replaying attack, so that intruders are not able to obtain sensitive data by replaying a previously intercepted message; and 5) forward and backward secrecy[1] to legal members when they randomly join or leave an in-progress group communication session. Those requirements are fulfilled by proper data encryption working in conjunction with the key distribution scheme. Due to relatively low computation power of mobile devices and the dynamics of NEMO networks, the key distribution scheme should be light weight on minimizing message exchanges and computation complexity [5]–[7]. The

[1]Forward secrecy is that an adversary who knows the old keys cannot discover the subsequent new keys, and backward secrecy is that an adversary who knows the current keys cannot find the preceding keys.

Manuscript received June 3, 2005; revised February 6, 2006. This work was supported in part by the National Natural Science Foundation of China under Contract 60573144, Contract 60218003, Contract 60429202, and Contract 90412012, in part by the Projects of Development Plan of the State High Technology Research under Contract 2003CB314804, and in part by Intel IXA University Research Plan.

Y. Jiang and C. Lin are with Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: yxjiang@csnet1.cs.tsinghua.edu.cn; clin@csnet1.cs.tsinghua.edu.cn).

M. Shi and X. Shen are with Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: mshi@bbcr.uwaterloo.ca; xshen@bbcr.uwaterloo.ca).

Digital Object Identifier 10.1109/JSAC.2006.875114

dynamics of NEMO network also increases service interruption probability, which requires that the scheme should be robust even when a few members lose connections temporarily. This paper focuses on proposing a robust multiple key sharing and distribution scheme.

A secure key distribution scheme should guarantee that only legitimate members share a common secret key which can be used in a secure group communication session. A key distribution scheme for traditional conference key distribution was first proposed in [8], which is implemented by using public key cryptography. However, it did not consider the dynamics of the network and lightweight computation requirement. The key distribution scheme was further studied in [9]–[11]. In [9], active members can dynamically join or leave an in-progress group communication session, but two cryptosystems are required, which is not friendly for the mobile devices. In [10], self-encryption is proposed to simplify the key distribution scheme However, the scheme does not offer identity anonymity which gives intruders an easy access to the real identity of a member by message interception, and the unauthorized entities are able to track the member's moving history and current location. In the scheme proposed in [11], a member can prove himself to NC without revealing his secret information and the computation complexity imposed on a member node is light weight. A missing point is that the scheme does not consider complete key refreshment solution and the communication may be compromised by using a stale or compromised key. In addition, it does not consider the impaired NEMO communication environment, where it may not work well if one or more members cannot communicate with NC due to communication interruption. Therefore, those existed key distribution schemes are not quite suitable for NEMO group communications scenario.

In this paper, a novel mobile multiple key sharing and distribution scheme is proposed. The scheme offers several attractive features satisfying the security and performance requirements in NEMO communication networks. Multiple keys can be shared by all members and be applied in current and future applications during a NEMO group communication session. Considering the dynamics of NEMO network, the threshold mechanism allows n members to share these keys in such a way that only any t or more members are needed to cooperatively reconstruct the keys. The mechanism effectively improves the flexibility and robustness of the scheme. The proposed scheme also offers two key refreshment mechanisms: *compulsive key refreshment* is used to resolve the key renewal when a member dynamically joins or leaves an in-progress NEMO group communications; and *automatic key refreshment* is suitable to renew the keys once the keys is expired. Therefore, both forward and backward secrecy can be guaranteed.

The rest of this paper is organized as follows. The multiple key sharing and distribution scheme with threshold mechanism for NEMO group communications is proposed in Section II. The security and performance evaluation of the scheme are detailed in Sections III and IV, respectively. The comparison between the proposed scheme and the existing scheme [11] is discussed in Section V. Section VI gives the concluding remarks.

II. Mobile Multiple Key Sharing and Distribution Scheme

The proposed multiple key sharing and distribution scheme with (n, t) threshold mechanism is based on multisecret sharing [12]–[15] and modular square root (MSR) technique [16]. First, we briefly introduce the concept and properties of MSR. The security of MSR depends on the low solvability of extracting MSRs of a quadratic residue modulo n when two factors of n are unknown. It is computationally infeasible to factorize $n(= p \cdot q)$ when the two prime factors p and q are large enough [17].

A. Modular Square Root (MSR)

MSR technique is based on Euler criterion and quadratic residues [16]. Let a and n be nonzero integers with $\gcd(a, n) = 1$, where a is called a quadratic residues modulo n if the congruence $x^2 = a (\bmod n)$ has solutions. The solutions are called MSRs of *quadratic residue* a modulo n. The *Euler Criterion* is described as that nonzero integer a is a quadratic residue modulo n if and only if $a^{(p-1)/2} = 1 (\bmod p)$, where p is an odd prime and $\gcd(a, p) = 1$. An MSR has the following two important properties.

Property 1: Let $n = p \cdot q$ and $\gcd(a, n) = 1$, where p, q are two primes with $p, q = 3 (\bmod 4)$. Then, a is a quadratic residue modulo n if and only if $a^{(p-1)/2} = 1 (\bmod p)$ and $a^{(q-1)/2} = 1 (\bmod q)$.

This property indicates that if a is a quadratic residue modulo n, i.e., $x^2 = a (\bmod n)$ is solvable, then square roots $r_{1,2,3,4}$ of quadratic residue a modulo n can be computed as

$$X = a^{(p+1)/4} (\bmod p) \tag{1}$$

$$Y = a^{(q+1)/4} (\bmod q) \tag{2}$$

$$r_{1,2,3,4} = (\pm X \cdot q \cdot q^*) \pm (Y \cdot p \cdot p^*) (\bmod n) \tag{3}$$

where $p^* = p^{-1} (\bmod q)$ and $q^* = q^{-1} (\bmod p)$.

Property 2: Let $n = p \cdot q$ and $\gcd(a, n) = 1$, where p, q are two distinct odd primes and $p, q = 3 (\bmod 4)$. Then, the number of quadratic residue modulo n is $(p-1)(q-1)/4$.

This property indicates that the probability of any integer a to be a quadratic residue modulo n is about 1/4.

B. Multiple Key Sharing and Distribution Scheme Architecture

The proposed NEMO secure group communication scheme consists of three phases: a secret shadow certificate issuing phase, secret key establishing phase, and key reconstruction and distribution phase. It is described according to the order of message exchanges. The corresponding security goals are discussed along with descriptions of the message exchanges. Table I shows the notations used in this paper.

C. Secret Shadow Certificate Issue Phase

CA generates two large primes p_{ca} and q_{ca} such that $p_{ca}, q_{ca} = 3 (\bmod 4)$ and computes $n_{ca} = p_{ca} \cdot q_{ca}$. Let g be a generator with order n_{ca} in $Z_{n_{ca}}$. n_{ca} and g are public to all parties, while p_{ca} and q_{ca} are kept secret by CA.

TABLE I
NOTATIONS FOR THE PROPOSED SCHEME

A_i	Alias of mobile member C_i
$E_k(X)$	Symmetric encryption of X with the secret key k
$E_k^{-1}(Y)$	Symmetric decryption of Y with the secret key k
$H(X)$	One-way hash function of X
ts_i	Timestamp when C_i sends a request to NC
$\|\|$	Concatenation of two integer blocks
ID_i	Identity information of member C_i
ID_{NC}	Identity information of NC
CK^*	The set of shared keys
CK_i	The ith shared key in all conference members
L	The lifetime of all the keys CK^*

Let $CK = \{CK_1, CK_2, \ldots, CK_m\}$ be a set of keys and $G = \{C_1, C_2, \ldots, C_n\}$ be a group of n members that share the secrets in key set CK. Let CA randomly generate a $t-1$ degree polynomial

$$f(x) = a_0 + a_1 x + \ldots + a_{i-1} x^{i-1} (\mathrm{mod}\ n_{ca}) \quad (4)$$

where $a_i \in Z_{n_{ca}}$. Then, CA computes a secret key for each $C_i \in G$ $(i = 1, 2, \ldots, n)$ as

$$x_i = f(ID_i) \cdot p_i^{-1} (\mathrm{mod}\ n_{ca}) \quad (5)$$

where

$$p_i = \prod_{C_k \in G \setminus \{C_i\}} (ID_i - ID_k)(\mathrm{mod}\ n_{ca}). \quad (6)$$

For each member $C_i \in G$, CA computes a tuple $SCK_{i,j} = (E_i, F_{i,j})$ as the secret shadow of all the keys $CK_j \in CK$ according to Algorithm 1.

Algorithm 1: Key Shadows for Keys Generation

function *generate-key-shadows*()

for $i = 1$ to n do

$$E_i = g^{x_i} (\mathrm{mod}\ n_{ca}); \quad (7)$$

for $j = 1$ to m do

$$F_{i,j} = CK_j \oplus g^{a_0} (\mathrm{mod}\ n_{ca}); \quad (8)$$

end

construct tuple $SCK_{i,j} = (E_i, F_{i,j})$;

end

return all tuples $SCK_{i,j} = (E_i, F_{i,j})$;

end

After successful secret shadows generation, CA issues a certificate $\{ID_i, (SCK_{i,1}, SCK_{i,2}, \ldots, SCK_{i,m}), (j_i, s_i)\}$ via a secure channel to each member C_i with identity ID_i, where pair (j_i, s_i) are the output of Algorithm 2. Each member keeps its shadow certificate secretly. (j_i, s_i) can be found after four iterations on average according to Property 2 of MSR.

Algorithm 2: Modular Square Root Generation

function *generate-modular-square-root* ()

let $j_i = -1$;

repeat

$$j_i = j_i + 1; \quad a = H(ID_i, j_i); \quad (9)$$

until

$a^{\frac{p_{ca}-1}{2}} = 1(\mathrm{mod}\ p_{ca})$ & $a^{\frac{q_{ca}-1}{2}} = 1(\mathrm{mod}\ q_{ca})$;

compute four square roots of $x^2 = a(\mathrm{mod}\ n_{ca})$ based on (1)–(3);

choose the smallest root as s_i;

return (j_i, s_i);

end

D. Secret Key Establishment Phase

Secret key establishment phase, as shown in Fig. 2, establishes a secret key shared by NC and each member, which is used to protect the authentication communications between them. NC chooses two large primes p_{nc} and q_{nc} such that $p_{nc}, q_{nc} = 3(\mathrm{mod}\ 4)$ and computes $n_{nc} = p_{nc} \cdot q_{nc}$. n_{nc} is public to all members, while p_{nc} and q_{nc} are known only by NC.

Consider that a chairperson (C_1) intends to initiate a NEMO group communications for members $(C_1, C_2, \ldots, C_n)$, and the NEMO group communications requires multiple keys to satisfy different application requirement. Let the keys be $CK^* = \{CK_{a_1}, CK_{a_2}, \ldots, CK_{a_i}\}$ $(|CK^*| \le m, CK^* \subseteq CK, (a_1, a_2, \ldots, a_k) \in \{1, 2, \ldots, m\}$, $\forall x, y \in \{1, 2, \ldots, m\}$, $a_x \ne a_y$ if $x \ne y)$. The secret key establishment phase is described as follows.

Step 1) Chairperson C_1 randomly selects two integers λ_1 and r_1 $(\lambda_1, r_1 < n_{ca})$ and computes the following parameters:

$$\begin{aligned}
\alpha_1 &= \lambda_1^2 (\mathrm{mod}\ n_{ca}) \\
\beta_1 &= \lambda_1^2 (\mathrm{mod}\ n_{ca}) \\
k_1 &= H(r_1) \\
R_1 &= r_1^2 (\mathrm{mod}\ n_{nc}) \\
U_1 &= E_{K_1}(ID_{NC}) \\
V_1 &= E_{K_1}(ts_1 \| A_1 \| (SCK_{1,a_1}, \ldots, SCK_{1,a_i}) \\
&\qquad \| (ID_1, j_1, \alpha_1, \beta_1) \| ID_2 \| \ldots \| ID_n)
\end{aligned}$$

where the parameters s_1, j_1, and $(SCK_{1,a_1}, SCK_{1,a_2}, \ldots, SCK_{1,a_i})$ are directly obtained from the secret shadow certificate $(ID_1, (SCK_{1,1}, SCK_{1,2}, \ldots, SCK_{1,m}), (j_1, s_1))$ of C_1. Then, C_1 sends the three-tuple (R_1, V_1, U_1) to NC and keeps the key k_1 secret, where R_1 and V_1 are used by NC to determine the secret key K_1.

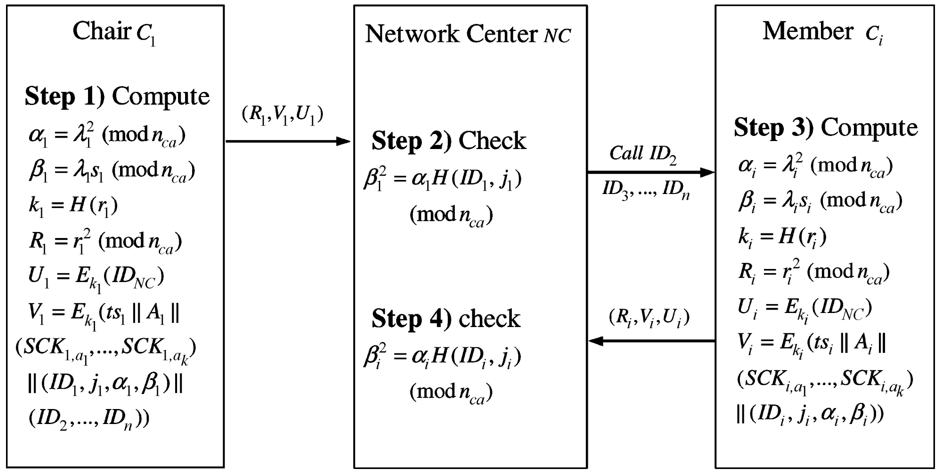


Fig. 2. Secret key establishment phase.

Note that NC uses the tuple $(ID_1, j_1, \alpha_1, \beta_1)$ in V_1 to authenticate member C_1. R_1 is employed to convey the secret key k_1 from to NC. U_1 is used to identify the secret key K_1. The replay attack is prevented by timestamp ts_1. Alias A_1 offers identity anonymity for a member C_1 against location disclosure of a member.

Step 2) After receiving message (R_1, V_1, U_1), NC extracts k_1 and authenticates the identity of C_1 as follows.

1) Compute four MSRs $r_{1,2,3,4}$ of $x^2 = R_1 (\bmod\ n_{nc})$ with the knowledge of p_{nc} and q_{nc} according to (1)–(3) and its corresponding four secret key candidates: $r_{1,2,3,4} = H(r_{1,2,3,4})$.
2) Check which candidate x satisfies $E_x^{-1}(U_1) = ID_{NC}$. The obtained candidate x is key k_1.
3) Decrypt the message V_1 with its corresponding secret key k_1 to obtain plain text $\{ts_1, A_1, (SCK_{1,a_1}, SCK_{1,a_2}, \ldots, SCK_{1,a_k}), (ID_1, j_1, \alpha_1, \beta_1), ID_2, ID_3, \ldots, ID_n\}$, check timestamp ts_1, and verify if $\beta_1^2 = \alpha_1 \cdot H(ID_1, j_1) (\bmod\ n_{ca})$. If it is true, C_1 is authenticated.

After secret key k_1 is established to protect communications between NC and C_1 during the group communication session, NC calls each member $ID_2, ID_3, \ldots, ID_n$, respectively.

Step 3) Similar to Step 1, C_i also randomly chooses two integers λ_i and r_i $(\lambda_i, r_i < n_{ca})$ and computes the parameters as follows:

$$\alpha_i = \lambda_i^2 (\bmod\ n_{ca}) \tag{10}$$
$$\beta_i = \lambda_i \cdot s_i (\bmod\ n_{ca}) \tag{11}$$
$$k_i = H(r_i), \tag{12}$$
$$R_i = r_i^2 (\bmod\ n_{nc}) \tag{13}$$
$$U_i = E_{k_i}(ID_{NC}) \tag{14}$$
$$V_i = E_{k_i}\left(ts_i \| A_i \| \left(SCK_{i,a_1}, SCK_{i,a_2}, \ldots, SCK_{i,a_k}\right) \| (ID_i, j_i, \alpha_i, \beta_i)\right) \tag{15}$$

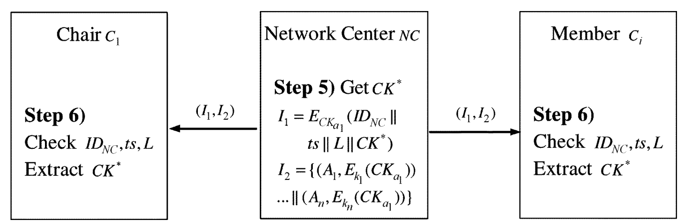


Fig. 3. Key reconstruction and distribution phase.

where the tuple $(SCK_{i,a_1}, SCK_{i,a_2}, \ldots, SCK_{1,a_k})$ denotes the secret shadow of the key in $CK^* = \{CK_{a_1}, CK_{a_2}, \ldots, CK_{a_k}\}$, respectively. Subsequently, mobile member C_i sends the response message, three-tuple (R_i, V_i, U_i), to NC and keeps key k_i secret.

Step 4) On receiving (R_i, V_i, U_i) from C_i, NC computes four MSRs of $x^2 = R_i (\bmod\ n_{nc})$, determines k_i, and authenticates C_i in the same way as in Step 2 by verifying whether the following condition holds or not:

$$\beta_i^2 = \alpha_i \cdot H(ID_i, j_i) (\bmod\ n_{ca}). \tag{16}$$

If the authentication is successful, NC and the member C_i can establish a secret key k_i to protect communications between them during the group communication session.

After executing the above two steps, NC establishes a shared secret key k_i with each member C_i, respectively. Note that the function of three-tuple (R_i, V_i, U_i) is similar to that of tuple in Step 1.

E. Key Reconstruction and Distribution Phase

As shown in Fig. 3, the key reconstruction and distribution phase mainly includes two steps. The requisite keys CK^* are first reconstructed, and then they are distributed to each mobile member. For convenience in referencing each step, we continue the description with Step 5.

Step 5) In order to provide (n,t) threshold mechanism in the proposed group communications scheme, NC must receive at least t members' responses (including chairperson C_1). Suppose that NC receives response $\theta(\theta \geq t-1)$ message (R_i, V_i, U_i) from the called members. Let the θ members and chairperson C_1 constitute a member set $W = \{C_{w_1}, C_{w_2}, \ldots, C_{w_{\theta+1}}\}(|W| = \theta + 1)$. Then, NC randomly chooses the messages of any t members (including chairperson C_1) from W to reconstruct the keys CK^* which are requested by the chairperson C_1. Without loss of generality, let t members be $G^* = \{C_{b_1}, C_{b_2}, \ldots, C_{b_t}\}$ with $|G^*| = t$. Evidently, the three sets G^*, W, and G satisfy the inclusion relation $G^* \subseteq W \subseteq G$. Once NC extracts all the shadows $SCK^* = SCK_{b_i,a_j} | b_i \in \{b_1, b_2, \ldots, b_t\}, a_j \in \{a_1, a_2, \ldots, a_k\}$ from the messages $(R_{b_i}, B_{b_i}, U_{b_i})$ of the t members, each secret key $CK_{a_j} \in CK^*$, $a_j \in \{a_1, a_2, \ldots, a_k\}$ can be reconstructed by using (n,t) threshold mechanism according to Algorithm 3. Then, NC encrypts these keys CK^* with the established keys $k_{w_j}(C_{w_j} \in W)$ in Step 4, respectively.

In order to provide authentication, NC computes the following two messages:

$$I_1 = E_{CK_{a_1}}(ID_{NC} \| ts \| L \| CK^*), \text{ and} \tag{17}$$

$$I_2 = \Big\{ \left(A_{w_1}, E_{k_{w_1}}(CK_{a_1})\right) \| \left(A_{w_2}, E_{k_{w_2}}(CK_{a_1})\right) \| \ldots \| \left(A_{w_{\theta+1}}, E_{k_{w_{\theta+1}}}(CK_{a_1})\right) \Big\} \tag{18}$$

where ts is timestamp, $CK_{a_1} \in CK^*$, and L is the lifetime of all the keys CK^*. Note that the key CK_{a_1} is used to encrypt the information in message I_1. NC broadcasts $\{I_1, I_2\}$ to all members in W, who have sent the response message to NC in Step 3.

Algorithm 3: Keys Reconstruction

function *reconstruct-session-keys* ()

for each $SCK_{b_i,a_j} \in SCK^*$ do

extract E_{b_i} from tuple SCK_{b_i,a_j};

end

compute $\prod_{C_{b_i} \in G^*} E_{b_i} (\bmod\ n_{ca})$;

for each $CK_{a_j} \in CK^*$ do

extract F_{b_i,a_j} from tuple SCK_{b_i,a_j};

$CK_{a_j} = F_{b_i,a_j} \oplus \prod_{C_{b_i} \in G^*} E_{b_i} (\bmod\ n_{ca})$;

end

return all $CK_{a_j} \in CK^*$;

end

Step 6) On receiving the broadcast messages (I_1, I_2) from NC, each member C_{w_j} $(C_{w_j} \in W)$ extracts $E_{k_{w_j}}(CK_{a_1})$ from I_2 according to his alias A_{w_j} and obtains key CK_{a_1} by decrypting $E_{k_{w_j}}(CK_{a_1})$ with his secret key k_{w_j}. Then, the member uses CK_{a_1} to decrypt the messages I_1 as

$$\begin{aligned} E^{-1}_{CK_{a_1}}(I_1) &= E^{-1}_{CK_{a_1}}\left(E_{CK_{a_1}}(ID_{NC} \| ts \| L \| CK^*)\right) \\ &= \{ID_{NC} \| ts \| L \| CK^*\}. \end{aligned} \tag{19}$$

Each member verifies the validity of the timestamp ts and the identity authenticity of NC. If it is true, all the keys CK^* should be authenticated.

So far, all the required keys CK^* have been reestablished among all members W. Depending on application requirements, the content of different group communication conversations can be encrypted and decrypted with the corresponding key in CK^*.

III. Forward and Backward Secrecy

To guarantee the forward and backward secrecy in the proposed scheme, two effective key refreshment mechanisms, compulsive key refreshment and automatic key refreshment, are designed for dynamically joining or leaving a group communication session and periodically renewing keys, respectively. Specifically, CK^* is required to be refreshed if: 1) a member joins an in-progress NEMO group communication session; 2) a member leaves an in-progress NEMO group communication session; and 3) the keys with lifetime L are expired. Note that in the latter two events, all members are forced to be reconfigured; and while in the first event, only the new member is required to be reconfigured.

Periodic Automatic Key Refreshment: In the proposed scheme, all the keys CK^* have a lifetime L. To automatically renew the keys periodically, NC performs the following steps to update the keys CK^* at regular interval L.

1) Once the lifetime L of the keys CK^* is due, NC chooses a set of random numbers $CK^*_{new} = \{CK^*_{a_1}, CK^*_{a_2}, \ldots, CK^*_{a_k}\}$ as the new keys and encrypts CK^*_{new} with the secret keys k_{w_i} of member C_{w_i} $(i = 1, 2, \ldots, \theta+1)$, respectively. Then, NC broadcasts the following two messages to all members:

$$I'_1 = E_{CK'_{a_1}}(ID_{NC} \| ts' \| L \| CK^*_{new})$$

and

$$I'_2 = \Big\{ \left(A_{w_1}, E_{k_{w_1}}(CK^*_{a_1})\right) \| \left(A_{w_2}, E_{k_{w_2}}(CK^*_{a_1})\right) \| \ldots \| \left(A_{w_{\theta+1}}, E_{k_{w_{\theta+1}}}(CK^*_{a_1})\right) \Big\}$$

where the key $CK^*_{a_1} \in CK^*_{new}$ is used to encrypt the information in I'_1.

2) Each member C_{w_1} extracts the corresponding $E_{k_{w_j}}(CK^*_{a_1})$ according to his alias A_{w_j} in I'_2. Then, the member gets $CK^*_{a_1}$ by decrypting $E_{k_{w_j}}(CK^*_{a_1})$ with k_{w_j}. The member uses $CK^*_{a_1}$ to decrypt I'_1 and verifies the validity of the timestamp ts' and the identity of NC.

If it is true, the member gets the new keys CK^*_{new} for the group communication session.

Member Joining: When a new member $C_{w_{\theta+2}}$ joins an in-progress group communication session, the procedures of obtaining CK^* for $C_{w_{\theta+2}}$ can be described as follows.

1) $C_{w_{\theta+2}}$ requires the permission from the chairperson C_1. Then, C_1 sends NC the message $J = E_{k_1}(ID_{w_{\theta+2}} \| ts' \| JOIN)$, where ts' is the timestamp and $ID_{w_{\theta+2}}$ is the identity of $C_{w_{\theta+2}}$.
2) NC decrypts J with k_1 to obtain ts' and $ID_{w_{\theta+2}}$, then it checks the validity of the timestamp ts' and the identity authenticity of member $ID_{w_{\theta+2}}$. If it is true, NC calls $C_{w_{\theta+2}}$.
3) $C_{w_{\theta+2}}$ and NC establish a shared secret key $k_{w_{\theta+2}}$ according to procedures in the secret key establishment phase.
4) NC sends $C_{w_{\theta+2}}$ two messages: $I'_1 = E_{CK_{a_1}}(ID_{NC} \| ts' \| L \| CK^*)$ and $I'_2 = \{A_{w_{\theta+2}}, E_{k_{\theta+2}}(CK_{a_1})\}$.
5) $C_{w_{\theta+2}}$ extracts CK_{a_1} from I'_2, then uses CK_{a_1} to decrypt I'_1 and verifies the validity of the timestamp ts' and the identity of NC. If both are true, $C_{w_{\theta+2}}$ gets CK^* and joins the group communication session.

Member Leaving: When a member leaves an in-progress NEMO group communication session, NC must update all of the previous keys CK^* to assure their freshness. Without loss of generality, assume that member $C_{w_{\theta+1}}$ leaves the group communication session. The procedure of updating keys contains four steps, which can be characterized as follows.

1) Chairperson C_1 sends NC the message $Q = E_{k_1}(ID_{w_{\theta+1}} \| ts' \| QUIT)$, where $ID_{w_{\theta+1}}$ is the identity of member $C_{w_{\theta+1}}$.
2) NC obtains ts' and $ID_{w_{\theta+1}}$ by decrypting Q with k_1, and checks the authenticity of timestamp ts'. If it is true, member $C_{w_{\theta+1}}$ is removed from the member list.

Steps 3 and 4 are similar to the steps 1 and 2 in periodic automatic key refreshment algorithm, respectively. The only difference is that the members are C_{w_i} $(i = 1, 2, \ldots, \theta)$ since $C_{w_{\theta+1}}$ has been eradicated from the group communication session.

IV. Correctness and Security Analysis

In this section, we demonstrate the correctness and the security of the proposed scheme.

A. Correctness Analysis

The correctness analysis verifies the validity of the (n, t) threshold mechanism with multiple keys distribution.

Theorem 1: Algorithm 3 can successfully reconstruct the requisite keys $CK_{a_j} \in CK^*$, by computing

$$CK_{a_j} = F_{b_i,a_j} \oplus \prod_{C_{b_i} \in G^*} E_{b_i}^{\omega_{b_i}} (\bmod\, n_{ca}) \tag{20}$$

where

$$\omega_{b_i} = \prod_{C_{b_k} \in G^* \setminus \{C_{b_i}\}} (-ID_{b_k}) \cdot \prod_{C_{b_k} \in G \setminus G^*} (ID_{b_i} - ID_{b_k}). \tag{21}$$

Proof: As described in Step 5, NC can randomly choose t members from $W = \{C_{w_1}, C_{w_2}, \ldots, C_{w_{\theta+1}}\}$. Let the t members be $G^* = \{C_{b_1}, C_{b_2}, \ldots, C_{b_t}\}$ with $|G^*| = t$.

We first prove that the following equation holds:

$$a_0 = \sum_{C_{b_i} \in G^*} (x_{b_i} \cdot \omega_{b_i}) (\bmod\, n_{ca}). \tag{22}$$

According to Lagrange interpolation polynomial and (4), we have

$$\begin{aligned}
a_0 &= f(0) \\
&= \sum_{C_{b_i} \in G^*} \left\{ f(ID_{b_i}) \cdot \prod_{C_{b_k} \in G^* \setminus \{C_{b_i}\}} (-ID_{b_k}) \cdot (ID_{b_i} - ID_{b_k})^{-1} \right\} (\bmod\, n_{ca}) \\
&= \sum_{C_{b_i} \in G^*} \left\{ f(ID_{b_i}) \cdot \prod_{C_{b_k} \in G \setminus \{C_{b_i}\}} (ID_{b_i} - ID_{b_k})^{-1} \cdot \prod_{C_{b_k} \in G^* \setminus \{C_{b_i}\}} (-ID_{b_k}) \cdot \prod_{C_{b_k} \in G \setminus G^*} (ID_{b_i} - ID_{b_k}) \right\} (\bmod\, n_{ca}) \\
&= \sum_{C_{b_i} \in G^*} \left(f(ID_{b_i}) \cdot p_{b_i}^{-1} \cdot \omega_{b_i} \right) (\bmod\, n_{ca}) \\
&= \sum_{C_{b_i} \in G^*} (x_{b_i} \cdot \omega_{b_i}) (\bmod\, n_{ca}).
\end{aligned}$$

To reconstruct CK_{a_j}, we consider the t key shadows $SCK_{b_i,a_j} \in SCK^*$, $b_i \in \{b_1, b_2, \ldots, b_t\}$. We extract E_{b_i} and F_{b_i,a_j} from $SCK_{b_i,a_j} = (E_{b_i}, F_{b_i,a_j})$. Since $E_{b_i} = g^{x_{b_i}} (\bmod\, n_{ca})$ and $F_{b_i,a_j} = CK_{a_j} \oplus g^{a_0} (\bmod\, n_{ca})$, key CK_{a_j} can be calculated by

$$\begin{aligned}
CK^*_{a_j} &= F_{b_i,a_j} \oplus \prod_{C_{b_i} \in G^*} E_{b_i}^{\omega_{b_i}} (\bmod\, n_{ca}) \\
&= \left(CK_{a_j} \oplus g^{a_0} \right) \oplus \prod_{C_{b_i} \in G^*} g^{(\omega_{b_i} \cdot x_{b_i})} (\bmod\, n_{ca}) \\
&= \left(CK_{a_j} \oplus g^{a_0} \right) \oplus g^{\sum_{C_{b_i} \in G^*} \omega_{b_i} \cdot x_{b_i}} (\bmod\, n_{ca}) \\
&= \left(CK_{a_j} \oplus g^{a_0} \right) \oplus g^{a_0} (\bmod\, n_{ca}) \\
&= CK_{a_j} (\bmod\, n_{ca}).
\end{aligned}$$

■

Theorem 1 indicates that the secret shadow generation algorithm (Algorithm 1) and the key reconstruction algorithm (Algorithm 3) are correct. In other words, the (n, t) threshold mechanism with multiple secrets is effective in the proposed scheme.

Theorem 2: NC can verify the identity authenticity of member C_i by checking if (16) holds.

Proof: According to Algorithm 2, s_i is the smallest square root of $x^2 = a(\bmod n_{ca})$, where $a = H(ID_i, j_i)$. Then, we have $s_i^2 = H(ID_i, j_i)(\bmod n_{ca})$.

Square both sides of (11), (16) can be deduced by

$$\begin{aligned}\beta_i^2 &= \lambda_i^2 \cdot s_i^2 = \alpha_i \cdot s_i^2 (\bmod n_{ca})\\ &= \alpha_i \cdot H(ID_i, j_i)(\bmod n_{ca}).\end{aligned}$$

■

Theorem 2 indicates that the MSR technique can efficiently provide an identity authentication mechanism in the proposed group communications scheme.

B. Security Analysis

Assuring (n, t) Threshold Mechanism: In a secret sharing scheme with (n, t) threshold mechanism, n members share a secret in such a way that only t or more members can cooperatively reconstruct the secret. From the view of information theory, the proposed scheme is a typical threshold scheme where knowing $t-1$ or fewer secret shadows provides no more information about the secret to an opponent than knowing no pieces. The multiple keys CK^* can also be successfully reconstructed. The validity of this mechanism has been verified in Theorem 1.

Member Identity Anonymity: In general, the location of a particular member should be hidden to prevent being tracked. If the identity information ID_i of a member C_i is transmitted in clear text, his location can be traced. The proposed scheme provides identity anonymity mechanism for all the members, because their identities ID_i $(i = 1, 2, \ldots, n)$ are encrypted in each step with secret key k_i $(i = 1, 2, \ldots, n)$ in steps 1 and 3 or key CK_{a_1} in step 5. Since any k_i or CK_{a_i} cannot be obtained, the intruder is not able to extract ID_i from intercepted messages to trace the location of any member.

Prevention of Replay Attack: Replay attack is a method that an intruder stores "stale" intercepted messages and retransmits them at a later time. An efficient measure against a replaying attack is to introduce timestamp ts and secret key lifetime L into the messages and set an expected legal time interval Δt for transmission delay. As shown in Figs. 2 and 3, all messages in each step contain a timestamp. According to the timestamp ts and Δt, the receiver verifies the validity of these messages by checking if $tc - ts_i < \Delta t$, where ts_i is the timestamp of a message, while tc is the current time when it is received. If the inequality holds, the message is valid. Otherwise, NC regards the message as a replaying message. Therefore, a replaying attack can be avoided to a large extent.

Privacy of Member Conversation Content: The conversation content is encrypted with secret key $CK_{a_j} \in CK^*$. An intruder cannot know the content without the knowledge of the key CK_{a_j}. To obtain CK_{a_j}, an intruder has to get a secret key k_i shared between NC and member C_i, and then use k_i to decrypt $E_{k_i}(CK_{a_j})$. However, in the secret establishment phase, the member C_i conveys the secret k_i to NC using MSR technique. Even though $R_i = r_i^2(\bmod n_{nc})$ can be intercepted, the intruder cannot determine r_i and compute key $k_i = H(r_i)$ because he cannot calculate the MSRs of the quadratic residue R_i modulo n_{nc} without knowing the two prime factors p_{nc} and q_{nc} of n_{nc}. Hence, the intruder is unable to obtain any key CK_{a_j} and eavesdrop the conversation content.

Prevention of Fraud: The proposed scheme provides a mutual authentication mechanism for NC and members. In order to prevent being cheated by an illegal member, NC authenticates member C_i by checking whether (16) holds. The validity of this equation has been proved in Theorem 2. In Steps 2 and 4, the legal member C_i uses (11) to compute β_i with the knowledge of his secret shadow certificate issued by CA, so (16) always holds. If an intruder intends to forge the secret shadow certificate $(ID_i, (SCK_{i,1}, SCK_{i,2}, \ldots, SCK_{i,m}), (j_i, s_i))$ of a member C_i, he is required to compute the MSRs of a quadratic residue $H(ID_i, j_i)(\bmod n_{ca})$, and then determine s_i, which is difficult without knowing two distinct prime factors p_{nc} and q_{nc} of n_{nc}.

On the other hand, a member C_i also authenticates NC since an intruder may impersonate the NC. In secret establishment phase, only NC can extract the secret k_i by calculating the MSRs of the quadratic residue R_i modulo n_{nc}, and then distribute the keys CK^* to C_i. C_i decrypts $E_{k_i}(CK_{a_1})$ to obtain CK_{a_1} and uses CK_{a_1} to decrypt the messages I_1 by computing (19).

The member verifies the authenticity of CK^* by checking the validity of the timestamp ts and the identity of NC. Evidently, an intruder cannot generate authentic CK^* without knowing the key k_i. Members can indirectly validate the identity of NC by checking authenticity of CK^*.

Forward and Backward Secrecy: The proposed scheme meets the security requirement for forward and backward secrecy. As described in Section II-F, the key distribution mechanism can update the key CK^* when a member dynamically joins or leaves an in-progress group communication session or the lifetime of the keys is overdue, and then redistributes the new keys to corresponding members.

V. Performance Analysis

The proposed scheme requires $2m$ unicast and one broadcast message exchanges, which is friendly for the wireless transmission. Therefore, the performance analysis will focus on the computational cost for mobile members NC and CA, respectively. The modular modulo n operation and the exponentiation operation are mainly considered, due to their higher computational complexity.

A. Computation Complexity for CA

CA issues a certificate $(ID_i, (SCK_{i,1}, SCK_{i,2}, \ldots, SCK_{i,m}), (j_i, s_i))$ for C_i, and implements two algorithms (Algorithms 1 and 2) in secret shadow certificate issue phase. Let $M(n)$ denote the computation complexity of modular modulo n. The computation complexity of Algorithm 1 is

$$\left\{\frac{3}{2} \cdot \left\lfloor \log\left(\frac{n_{ca}}{2}\right)\right\rfloor \cdot M(n_{ca}) + m \cdot M(n_{ca}) + 2n \cdot M(n_{ca}) + \text{inv}\left(\frac{n_{ca}}{2}\right)\right\}. \quad (23)$$

According to the binary algorithm for fast exponentiation [18], computing g^x will take $(3/2)\lfloor \log x \rfloor$ on average and

$2\lfloor \log x \rfloor$ multipliers in the worst case. So the complexity of computing $E_i = g^{x_i} (\mathrm{mod}\ n_{ca})$ is $(3/2)\lfloor \log x_i \rfloor$ in (7), which is approximately equal to $(3/2)\lfloor \log(n_{ca}/2) \rfloor$ on average. In addition, $\mathrm{inv}(n_{ca}/2)$ in (23) denotes the average complexity of computing the inverse of modulo p_i in (5) and (6). According to extended Euclidean algorithm [19], the average division number of computing an inverse of modulo n is $0.843 \cdot \log_2(n) + 1.47$. Therefore, the complexity of computing the inverse of modulo $n_{ca}/2$ is

$$\mathrm{inv}\left(\frac{n_{ca}}{2}\right) = 0.843 \cdot \log_2\left(\frac{n_{ca}}{2}\right) + 1.47 \tag{24}$$

and the total computation complexity of Algorithm 2 is

$$\begin{aligned} &\left\{6 \cdot \left\lfloor \log\left(\frac{p_{ca}-1}{4}\right)\right\rfloor \cdot M(p_{ca}) + 6 \cdot \left\lfloor \log\left(\frac{q_{ca}-1}{4}\right)\right\rfloor \right.\\ &\quad \cdot M(q_{ca}) + \frac{3}{2} \cdot \left\lfloor \log\left(\frac{p_{ca}+1}{4}\right)\right\rfloor \cdot M(p_{ca})\\ &\quad \left. + \frac{3}{2} \cdot \left\lfloor \log\left(\frac{q_{ca}+1}{4}\right)\right\rfloor \cdot M(q_{ca})\right\}. \end{aligned} \tag{25}$$

If CA issues the shadow certificates for all n members G with keys CK, the total average computation complexity is

$$\begin{aligned} n \cdot &\left\{\frac{3}{2}\left\lfloor \log\left(\frac{n_{ca}}{2}\right)\right\rfloor M(n_{ca}) + m \cdot M(n_{ca}) \right.\\ &\quad \left. + 2n \cdot M(n_{ca}) + Inv\left(\frac{n_{ca}}{2}\right)\right\} + n\\ &\cdot \left\{\frac{3}{2}\left\lfloor \log\left(\frac{p_{ca}+1}{4}\right)\right\rfloor M(p_{ca}) + \frac{3}{2}\left\lfloor \log\left(\frac{q_{ca}+1}{4}\right)\right\rfloor \right.\\ &\quad \times M(q_{ca}) + 6\left\lfloor \log\left(\frac{p_{ca}-1}{4}\right)\right\rfloor M(p_{ca})\\ &\quad \left. + 6\left\lfloor \log\left(\frac{q_{ca}-1}{4}\right)\right\rfloor M(q_{ca})\right\} + \frac{3}{2}\lfloor \log a_0 \rfloor\\ &\quad + m \cdot M(n_{ca}). \end{aligned} \tag{26}$$

Although high computational complexity is required in shadow certificate issue phase, it does not affect the performance of the other two phases because CA issues secret shadow certificates to members in advance. So it does not degrade the performance of mobile devices.

B. Computation Complexity for Members

In the proposed scheme, the computation requirement for members, or mobile devices, is minor. Only three modular multiplications are required for member C_i to compute (α_i, β_i) and $R_i = r_i^2 (\mathrm{mod}\ n_{nc})$. The other operations for member C_i are just hash function, symmetric encryption and decryption operations.

Let $\mathrm{len}(x)$ denote the bit length of x. The approximate storage space (bit length) required for storing the certificate $(ID_i, (SCK_{i,1}, SCK_{i,2}, \ldots, SCK_{i,m}), (j_i, s_i))$, and the other requisite information of member C_i is

$$\begin{aligned} &\mathrm{len}(ID_i) + \mathrm{len}(s_i) + \mathrm{len}(j_i) + \mathrm{len}(n_{ca}) + \mathrm{len}(n_{nc})\\ &\quad + \sum_{i=1}^{m} (\mathrm{len}(E_i) + \mathrm{len}(F_{i,j})). \end{aligned} \tag{27}$$

C. Computation Complexity for NC

When NC calculates four MSRs according to (1)–(3) to obtain k_i sent by a member C_i, $p_{nc}^* = p_{nc}^{-1} (\mathrm{mod}\ q_{nc})$ and $q_{nc}^* = q_{nc}^{-1} (\mathrm{mod}\ p_{nc})$, $p_{nc}^* \cdot p_{nc} (\mathrm{mod}\ n_{nc})$, and $q_{nc}^* \cdot q_{nc} (\mathrm{mod}\ n_{nc})$ are independent of R_i. Therefore, they can be computed offline. Only two exponentiations and two modular multiplications are required to compute four MSRs on-the-fly. The average complexity of computing $R_i^{(p_{nc}+1)/4} (\mathrm{mod}\ p_{nc})$ and $R_i^{(q_{nc}+1)/4} (\mathrm{mod}\ q_{nc})$ is

$$\frac{3}{2}\left\{\left\lfloor \log\left(\frac{p_{nc}+1}{4}\right)\right\rfloor M(p_{nc}) + \left\lfloor \log\left(\frac{q_{nc}+1}{4}\right)\right\rfloor M(q_{nc})\right\}. \tag{28}$$

Consequently, for $\theta (\theta > t)$ members, the average computation complexity required for NC is

$$\begin{aligned} \theta &\left\{\frac{3}{2}\left\lfloor \log\left(\frac{p_{nc}+1}{4}\right)\right\rfloor M(p_{nc}) + \frac{3}{2}\left\lfloor \log\left(\frac{q_{nc}+1}{4}\right)\right\rfloor \right.\\ &\left. \cdot M(q_{nc}) + 2M(n_{nc}) + 2M(n_{na})\right\} + (k+\theta) M(n_{na}) \end{aligned} \tag{29}$$

with the addition of symmetric encryption and decryption operations, where k is the number of keys CK^*, and $(k+\theta)M(n_{na})$ denotes the computation complexity of the algorithm for reconstructing CK^* (Algorithm 3). The majority of computation is assigned to NC, which is a high-performance server and is suitable for such heavy computation.

VI. Performance and Security Comparison With Other Scheme

The performance and security comparisons between the proposed scheme and the one in [11] are shown in Table II. Our focus is on the communication and computation complexity required in the members. The number of modular multiplication, hash operations, symmetric encryption or decryption operations, and transmissions are compared. The security feature comparison includes identity anonymity, threshold mechanism, dynamic member, multiple key sharing and distribution, and forward and backward secrecy.

The main differences between the two schemes are shown in shaded rows. The comparison results indicate that the communication and computation complexity required for the proposed scheme is significantly less than in [11]. In addition, the proposed scheme also achieves the following exclusive security features: the (n, t) threshold mechanism, multiple key sharing and distribution mechanism, and periodically automatic key refreshment.

TABLE II
PERFORMANCE AND SECURITY COMPARISON IN MEMBERS

Compared merits	Scheme in [11]	Proposed scheme
Modular multiplications	13 or more (step 1 or 3)	3 (step 1 or 3)
Hash operation	1 (Step 1 or 3)	1 (step 1 or 3)
Symmetric encryption	2 (step 1 or 5)	2 (steps 1 or 5)
Symmetric decryption	2 (step 1 or 5)	1 (steps 1 or 5)
Message exchanges	$(4m-2)U+1B$	$2mU+1B$
Identity anonymity and location intractability	Yes	Yes
(n,t) threshold mechanism	N/A	Yes
Multiple key share and distribution	N/A	Yes
Automatic key refreshment	N/A	Yes
Dynamic member mechanism	Yes	Yes
Forward and backward secrecy	Yes	Yes

U: Unicast Message; B: Broadcast Message; m: the number of members

VII. CONCLUSION

In this paper, a novel secure mobile key sharing and distribution scheme has been proposed, which offers several attractive features and capabilities. The threshold mechanism effectively enhances the flexibility and robustness in sharing and distributing multiple keys in NEMO network environment. Compulsive key refreshment and automatic key refreshment mechanisms allow dynamically joining or leaving an in-progress group communication session and renewing keys periodically. The proposed scheme also uses small amount of message exchanges, and requires low computation capacity on mobile devices. Therefore, the proposed scheme can be efficiently deployed for NEMO group communications and offer salient security services. The conclusion goes here.

REFERENCES

[1] D. Brown, "Techniques for privacy and authentication in personal communication system," *IEEE Pers. Commun.*, vol. 2, pp. 6–10, Aug. 1995.

[2] A. Aziz and W. Diffie, "Privacy and authentication for wireless local area networks," *IEEE Pers. Commun.*, vol. 2, pp. 25–31, Feb. 1994.

[3] J. E. Wilkes, "Privacy and authentication needs for PCS," *IEEE Pers. Commun.*, vol. 2, pp. 11–15, Aug. 1995.

[4] S. Patel, "Weakness of North American wireless authentication protocol," *IEEE Pers. Commun.*, vol. 4, pp. 40–44, Jun. 1997.

[5] D. S. Wong and A. H. Chan, "Mutual authentication and key exchange for low power wire-less communications," in *Proc. IEEE Mil. Commun. Conf.*, 2001, vol. 1, pp. 39–43.

[6] S. L. Ng and C. Mitchell, "Comments on mutual authentication and key exchange protocols for low power wireless communications," *IEEE Commun. Lett.*, vol. 8, no. 4, pp. 262–263, Apr. 2004.

[7] J. K. Jan and Y. H. Chen, "A new efficient make up for wireless communications," in *Proc. 18th Int. Conf. Advanced Inf. Netw. Appl.*, 2004, vol. 2, pp. 347–350.

[8] I. Ingemarson, D. T. Tang, and C. K. Wong, "A conference key distribution system," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 5, pp. 714–720, Sep. 1982.

[9] M. S. Hwang, "Dynamic participation in a secure conference scheme for mobile communication," *IEEE Trans. Veh. Technol.*, vol. 48, no. 5, pp. 1469–1474, Sep. 1999.

[10] K. F. Hwang and C. C. Chang, "A self-encryption mechanism for authentication of roaming and teleconference services," *IEEE Trans. Wireless Commun.*, vol. 2, no. 2, pp. 400–407, Mar. 2003.

[11] X. Yi, C. K. Siew, C. H. Tan, and Y. Ye, "A secure conference scheme for mobile communication," *IEEE Trans. Wireless Commun.*, vol. 2, no. 6, pp. 1168–1177, Nov. 2003.

[12] R. G. E. Pinch, "On-line multiple secret sharing," *Electron. Lett.*, vol. 32, pp. 1087–1088, 1996.

[13] R. J. Hwang and C. C. Chang, "An on-line secret sharing scheme for multi-secrets," *Comput. Commun.*, vol. 21, no. 13, pp. 1170–1176, 1998.

[14] C. Blundo and B. Masucci, "Randomness in multi-secret sharing schemes," *J. Universal Comput. Sci.*, vol. 5, no. 7, pp. 367–389, 1999.

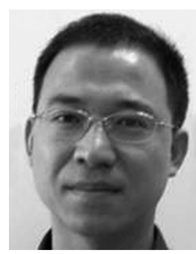
[15] ——, "A note on the randomness in dynamic threshold schemes," *J. Comput. Security*, vol. 7, no. 1, pp. 73–85, 1999.

[16] H. C. Williams, "A modification of the RSA public key encryption procedure," *IEEE Trans. Inf. Theory*, vol. IT–26, no. 6, pp. 726–729, Nov. 1980.

[17] R. L. Adelman and K. S. McCurley, "Open problem in number theoretic complexity," in *Proc. Algorithmic Number Theory Symp.*, 1994, pp. 291–322.

[18] D. M. Gardon, "A Survey of fast exponentiation method," *J. Algorithm*, vol. 27, no. 4, pp. 255–293, 2001.

[19] B. Schneier, *Applied Cryptography: Protocols, Algorithm, and Source Code C*, 2nd ed. New York: Wiley, 1996.



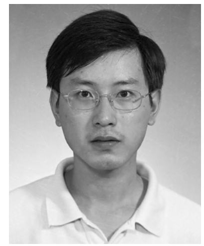
Yixin Jiang received the M.E. degree in computer science from the Huazhong University of Science and Technology, Wuhan, China, in 2002. He is currently working towards the Ph.D. degree at the Department of Computer Science and Technology, Tsinghua University, Beijing, China.

In 2005, he was a Visiting Scholar with the Department of Computer Sciences, Hong Kong Baptist University. His current research interests include security and performance evaluation in wireless communication and mobile computing. He has published more than 20 papers in research journals and IEEE conference proceedings in these areas.

Chuang Lin (M'03–SM'04) received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 1994.

He is a Professor and the Head of the Department of Computer Science and Technology, Tsinghua University. From 1985 to 1986, he was a Visiting Scholar with the Department of Computer Sciences, Purdue University. From 1989 to 1990, he was a Visiting Research Fellow with the Department of Management Sciences and Information Systems, University of Texas at Austin. From 1995 to 1996, he visited the Department of Computer Science, Hong Kong University of Science and Technology. His current research interests include computer networks, performance evaluation, network security, logic reasoning, and Petri net and its applications. He has published more than 200 papers in research journals and IEEE conference proceedings in these areas and has published three books.

Prof. Lin is the Chinese Delegate of IFIP TC6. He served as the General Chair, ACM SIGCOMM Asia Workshop 2005. He is an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and the Area Editor for the *Journal of Parallel and Distributed Computing.*



Minghui Shi received the B.S. degree from Shanghai Jiao Tong University, Shanghai, China, in 1996 and the M.S. degree from the University of Waterloo, Waterloo, ON, Canada, in 2002, both in electrical engineering. He is currently working towards the Ph.D. degree at the University of Waterloo.

His current research interests include wireless LAN/cellular network integration and network security.

Xuemin (Sherman) Shen (M'97–SM'02) received the B.Sc. degree from Dalian Maritime University, Dalian, China, in 1982, and the M.Sc. and Ph.D. degrees from Rutgers University, Piscataway, NJ, in 1987 and 1990, respectively, all in electrical engineering.

From September 1990 to September 1993, he was first with the Howard University, Washington, DC, and then with the University of Alberta, Edmonton, Canada. Since October 1993, he has been with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, where he is a Professor and the Associate Chair for Graduate Studies. He is a coauthor of two books, and has published more than 200 papers and book chapters in wireless communications and networks, control and filtering. His research focuses on mobility and resource management in interconnected wireless/wireline networks, UWB wireless communications systems, wireless security, and ad hoc and sensor networks.

Dr. Shen is a registered Professional Engineer of Ontario, Canada. He received the Premieres Research Excellence Award (PREA) from the Province of Ontario, Canada, for demonstrated excellence of scientific and academic contributions in 2003, and the Distinguished Performance Award from the Faculty of Engineering, University of Waterloo, for outstanding contribution in teaching, scholarship and service in 2002. He was the Technical Program Co-Chair for IEEE GLOBECOM '03 Symposium on Next-Generation Networks and Internet, ISPAN '04, IEEE Broadnet '05, QShine '05, and Special Track Chair of the 2005 IFIP Networking Conference. He serves as an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *ACM/Wireless Networks*, *Computer Networks*, *Wireless Communications and Mobile Computing* (Wiley), and the *International Journal of Computers and Applications*. He also serves as Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the *IEEE Wireless Communications*, and the *IEEE Communications Magazine*.