International Journal of Innovative Computing, Information and Control
Volume **8**, Number **1(B)**, January **2012**

ICIC International ©2012 ISSN 1349-4198
pp. 599–615

CERTIFICATELESS AUTHENTICATED GROUP KEY AGREEMENT SCHEME WITH PRIVACY-PRESERVATION FOR RESOURCE-LIMITED MOBILE DEVICES

Chung-Fu Lu[1], Tzong-Chen Wu[2] and Chien-Lung Hsu[3]

[1]Department of Information Management
Chihlee Institute of Technology
No. 313, Sec. 1, Wunhua Road, Banciao District, New Taipei City 22050, Taiwan
peter6125@gmail.com

[2]Department of Information Management
National Taiwan University of Science and Technology
No. 43, Sec. 4, Keelung Road, Taipei 106, Taiwan
tcwu@cs.ntust.edu.tw

[3]Department of Information Management
Chang Gung University
No. 259, Wen-Hwa 1st Road, Kwei-shan, Taoyuan 333, Taiwan
clhsu@mail.cgu.edu.tw

Received September 2010; revised January 2011

Abstract. *A group key agreement scheme is to establish a secret key shared among some participants for secure group-oriented applications. Many authenticated group key agreement schemes have been proposed, but few of them provide user anonymity for wireless mobile networks. Considering the user privacy issues and the characteristics of wireless mobile networks, we proposed a certificateless authenticated group key agreement scheme with privacy-preservation based on elliptic curve discrete logarithms. Resource-limited mobile devices can efficiently, cooperatively and anonymously establish an authenticated group key with entity authentication using no public key certificates in mobile wireless networks. The proposed scheme has the following properties: (1) It achieves contributory group key agreement with entity authentication. (2) It provides mutual authentication, explicit key authentication, key confirmation, forward secrecy, group key updating, user anonymity and some potential attacks resistance. (3) No public key certificates are used and the authenticity of public keys is implicitly verified. (4) It is efficient and suitable for unbalanced mobile wireless networks in terms of computational complexities, communication overheads, key storage and key management.*
Keywords: Group key agreement, Certificateless, Elliptic curve, Mobile wireless networks, Privacy

1. **Introduction.** A group key establishment scheme allows a number of users to cooperatively establish a secret group key for securing their group communication [1-5]. A group key agreement scheme is a key establishment technique whereby a shared secret key is derived by two or more specified parties as a function of information contributed by, or associated with, each of these, such that no party can predetermine the resulting value. Therefore, group key agreement schemes construct the session key by shared equally contributed information from every group member and enable group members to agree on a session key to secure their communication. An authenticated group key agreement scheme is a group key agreement scheme which provides implicit key authentication [6].

In 1999, Seo et al. [7] proposed a simple authenticated key agreement (AKA) scheme which is used to establish a common session key between two authenticated entities. A

multi-party AKA scheme is often called an authenticated group key agreement (AGKA) scheme. Based on multi-party extensions of the well-known Diffie-Hellman key agreement protocol, Ateniese et al. [8] presented a multi-party AKA scheme. After that, AGKA schemes were proposed in literature [9-16]. Recently, Bresson et al. [12] proposed an authenticated group key agreement scheme for low-power mobile devices based on public key technology. In 2005, Nam et al. pointed out the critical security flaws inherent in Bresson et al.'s scheme to show their scheme cannot achieve forward secrecy, implicit key authentication and known key security [13]. They also proposed a patch to fix the security flaws. Later, Nam et al. [14] further proposed a group key agreement scheme with constant-round for the unbalanced wireless networks under decisional Diffie-Hellman assumption. The Nam et al.'s scheme, however, is non-authenticated one which implies it cannot provide user and message authentication.

In 2006, Tseng [15] showed that Bresson et al.'s and Nam et al.'s schemes are not contributory key agreement ones in which the group secret keys are derived from the contributions of all participant nodes. Furthermore, Tseng proposed a real contributory key agreement scheme [16] to allow every group node to contribute their shares to the group key generation. It is more efficient than Bresson et al.'s and Nam et al.'s schemes in terms of the computational complexities but less efficient than those in terms of the communication overheads. In Tseng's and Bresson et al.'s schemes, each low-power node must generate and transmit digital signature to the powerful node for entity authentication. They both suggest that low-power nodes with limited computing capabilities can prepare these digital signatures beforehand by off-line pre-computing them in advance. However, these two schemes are both vulnerable to the so-called impersonation attacks since no timestamp or nonce is bound in signing messages. The adversary can intercept the signatures and masquerade as the intended legal nodes by replaying the intercepted digital signatures to cheat the powerful node and other participating low-power node(s) [17].

In recent years, the security and privacy protection for group communication on the open network has become an increasing concern. An authenticated group key agreement ensures that entities communicate with each other securely through open channels. However, an authenticated group key agreement scheme is designed without consideration of privacy protection. If the user's identities of group members are all disclosed during the scheme execution, an adversary can trace the user and launch some attacks. To protect privacy, providing the user anonymity is an effective solution. Many authenticated group key agreements have been proposed, but few of them are suitable for the group and with user anonymity scheme at the same time [18-22].

In 2008, Wan et al. proposed an authenticated group key agreement with anonymity scheme [23], which adopts the ID-based public key cryptosystem. In 2009, Park et al. showed that Wan et al.'s scheme was insecure against colluding attack and could not satisfy the requirements for forward secrecy. They also proposed a new forward secure ID-based group key agreement scheme with anonymity [24]. Their group key agreement scheme supposes that the session initiator has a list of pseudonyms and real names of all communication entities who are involved in the group key agreement. The list is shared among all communication entities in the group key agreement. However, if the list is disclosed by a legal communication entity, the user anonymity cannot be achieved. Meanwhile, the communication entities know the real IDs of other members, so the complete user anonymity cannot be obtained.

Based on the elliptic curve cryptosystem and self-certified public key cryptosystem, this paper will propose a certificateless authenticated group key agreement scheme with privacy-preservation for resource-limited mobile devices. In 1991, the self-certified public

key cryptosystem has been introduced by Girault [25]. A self-certified public key system has three features: First, the secret key can be determined by the user himself/herself or together by the user and system authority (SA), and cannot be known to SA. Second, the user can use his/her own secret key to verify the authenticity of the self-certified public key issued by SA, and thus no extra certificate is required. Third, the task of public key verification can be further accomplished with subsequent cryptographic application (e.g., key distribution or signature scheme) in a logically single step [26,27]. Therefore, public key verification of the self-certified approach earns more efficiency in saving the communicational cost and the computational effort as compared with other approaches, such as ID-based public key cryptosystem. Elliptic curves were first proposed for use in cryptography by Koblitz [28] and Miller [29]. Elliptic curve cryptosystem is based on the hardness of solving the elliptic curve discrete logarithm problem. Based on the elliptic curve cryptosystem and self-certified public key cryptosystem, the proposed scheme provides entities with not only a secure channel but also defense of privacy. In our scheme, the identities of group members, who are involved in the authenticated group key agreement, will not be traced by any adversary.

The rest of this paper is organized as follows: In Section 2, we propose a novel authenticated group key agreement scheme with user anonymity. We discuss the security analysis and performance evaluation of the proposed scheme in Section 3 and Section 4, respectively. Finally, some concluding remarks are presented in Section 5.

2. The Proposed Scheme.

There are three roles involved in the proposed scheme: system authority (SA), low-power nodes and a powerful node as mentioned above. The SA is responsible to generate all necessary system parameters and cooperates with each node to generate valid node's private and public key pair. The powerful node will authenticate the legitimacy of the participant low-power nodes and determine a group secret key shared among them.

According to the Tseng's scheme, we also assume that the SA and powerful node are trusted. The proposed scheme consists of five phases: the system setup, the mobile node registration, the authenticated group key agreement, the node leaving and the node joining phases. The proposed scheme is suitable to the dynamic group applications, small group especially. It allows a cluster of low-power nodes and one powerful node (e.g., wireless gateway) to dynamically agree on a group secret key shared among them for securing communications. Detailed descriptions of these phases are given below.

2.1. System setup phase.

Initially, the SA determines a large prime p and a non-supersingular elliptic curve $E_p(a,b)$ as $y^2 = x^3 + ax + b (\mathrm{mod}\, p)$, where $a, b \in_R Z_p^*$ and $4a^3 + 27b^2 \bmod p \neq 0$. The SA further determines a large prime q and a base point G of order q over $E_p(a,b)$, where q is a divisor of the number of points on the elliptic curve $E_p(a,b)$. Let O be a point at infinity over $E_p(a,b)$, $Q_{i.x}/Q_{i.y}$ be the x-coordinate/y-coordinate of the point Q_i [30], and h be a collision resistant hash function which accept variable-length inputs and produce a fixed-length output. The private and public keys for the SA are respectively defined as s_{SA} and P_{SA}, where $s_{SA} \in_R Z_q$ and

$$P_{SA} = s_{SA}G. \tag{1}$$

The SA publishes $(p, q, E_p(a,b), h, G, P_{SA})$ while keeps s_{SA} secret.

2.2. Mobile node registration phase.

When a mobile node N_i associated with a distinguished identifier I_i wants to join the system, he will cooperate with the SA to perform the following steps to generate a valid private and public key pair.

Step 1: The mobile node N_i randomly chooses an integer $v_i \in_R Z_q$, computes

$$V_i = v_i G, \tag{2}$$

and then sends $\{I_i, V_i\}$ to the SA.

Step 2: On receiving $\{I_i, V_i\}$ sent from the node N_i, the SA checks whether the identifier I_i is unregistered. If it holds, the SA computes and returns $\{P_i, s_i\}$ to the node N_i, where

$$P_i = V_i + h(r_i \| I_i)G = (P_{i.x}, P_{i.y}), \tag{3}$$

$$s_i = h(r_i \| I_i) + (P_{i.x} + I_i) \cdot s_{SA} \bmod q, \tag{4}$$

$r_i \in_R Z_q$, and $\|$ is the concatenation symbol. Note that P_i is regarded as N_i's public key issued by the SA.

Step 3: The node N_i computes a private key

$$x_i = s_i + v_i \bmod q. \tag{5}$$

Further, N_i can verify the validity of the private key x_i by checking whether

$$x_i G = P_i + (P_{i.x} + I_i)P_{SA}. \tag{6}$$

If Equation (6) holds, (s_i, P_i) is a valid key pair of N_i.

2.3. **Authenticated group key agreement phase.** Without loss of generality, let $\boldsymbol{N} = \{N_1, N_2, \ldots, N_n\}$ be the set of n low-power nodes that want to agree on a group secret key shared among them. The group $\boldsymbol{N}$ is a dynamic group and its schematic diagram is shown in Figure 1. All low-power nodes of the group $\boldsymbol{N}$ must belong to the same high-power node domain. Each low-power nodes $N_i \in \boldsymbol{N}$ such as cell phone, personal assistant device (PDA) is a device with very-restricted computing power and some required memory capacity. The powerful node N_A is a device with high computing power capabilities and large memory capacity, such as the base stations of cellular mobile networks, the access points of wireless local area networks or the cluster-heads of mobile ad hoc networks. All the low-power nodes will cooperative with a powerful node N_A (i.e., group manager) to generate the group secret key. The responsibility of the N_A is to authenticate the identity of all low-power nodes and determines the group secret key k_G for them. The procedure for the authenticated group key agreement phase is shown in Figure 2 and stated as follows.

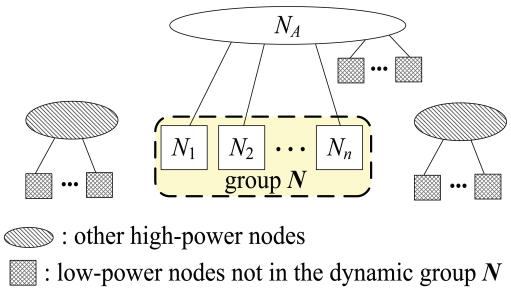

FIGURE 1. Schematic diagram of dynamic group

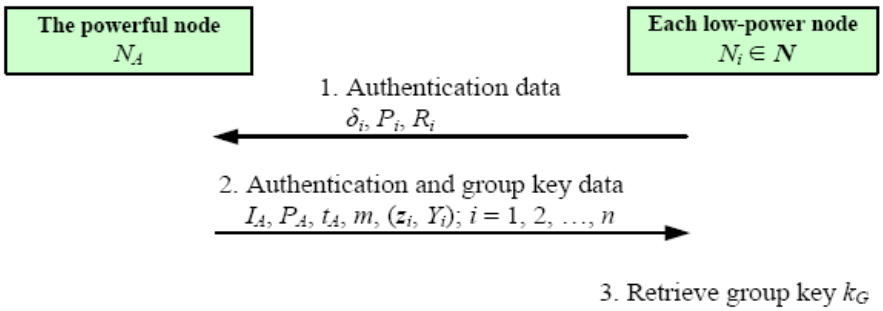

FIGURE 2. Authenticated group key agreement phase

Step 1: Each low-power node N_i computes r_i^{-1},

$$R_i = r_i G, \tag{7}$$
$$B_i = x_i(P_A + (P_{A.x} + I_A)P_{SA}), \tag{8}$$
$$C_i = r_i(P_A + (P_{A.x} + I_A)P_{SA}), \tag{9}$$
$$a_i = h(B_{i.x} \| C_{i.x} \| I_A \| I_i \| t_i), \tag{10}$$
$$\delta_i = (a_i \| I_i \| t_i) \oplus h(C_{i.x}), \tag{11}$$

and where $r_i \in_R Z_q$ and t_i is the current timestamp. Finally, N_i sends $\{\delta_i, P_i, R_i\}$ to N_A.

Step 2: The powerful node N_A verifies the legitimacy of N_i's identity and generates group secret key by performing the following sub-steps.

Step 2-1: On receiving $\{\delta_i, P_i, R_i\}$ from N_i (for $i = 1, 2, \ldots, n$), the powerful node N_A computes

$$B_i' = x_A(P_i + (P_{i.x} + I_i)P_{SA}), \tag{12}$$
$$C_i' = x_A R_i, \tag{13}$$
$$(a_i \| I_i \| t_i) = \delta_i \oplus h(C_{i.x}), \tag{14}$$

and checks whether $t_i' - t_i \leq \Delta t$, where t_i' is the timestamp of receiving $\{\delta_i, P_i, R_i\}$ and Δt is the preset acceptable delay threshold. If it holds, the powerful node verifies the legitimacy of the low-power node N_i by checking whether

$$h(B_{i.x}' \| C_{i.x}' \| I_A \| I_i \| t_i) = a_i. \tag{15}$$

If Equation (15) does not hold, N_A requests N_i to re-send valid $\{\delta_i, P_i, R_i\}$. Otherwise, both of the identity authentication and the authenticity public key P_i for N_i are verified.

Step 2-2: The powerful node N_A computes

$$R_A = r_A G, \tag{16}$$
$$z_i = h(R_{A.x} \| B_{i.x}' \| C_{i.x}' \| t_A), \tag{17}$$
$$k_G = h(R_{A.x} \| z_1 \| z_2 \| \ldots \| z_n \| t_A), \tag{18}$$
$$m = h(k_G \| I_A), \tag{19}$$
$$Y_i = r_A R_i, \tag{20}$$

where t_A is the current timestamp of powerful node N_A, $r_A \in_R Z_q$ and $i = 1, 2, \ldots, n$. Finally, N_A broadcasts $\{I_A, P_A, t_A, m, (z_i, Y_i); i = 1, 2, \ldots, n\}$ to all low-power nodes. Note that k_G is the shared group secret key. If the sent message is lost in transmission, the nodes are not able to join the group until those nodes

send another join request. The power node should send the above message to those nodes which did not get the group key.

Step 3: On receiving $\{I_A, P_A, t_A, m, (z_i, Y_i); i = 1, 2, \ldots, n\}$ from the powerful node N_A, each low-power node N_i checks whether $t_i'' - t_i \leq \Delta t'$, where t'' is the timestamp of receiving $\{I_A, P_A, t_A, m, (z_i, Y_i); i = 1, 2, \ldots, n\}$ and $\Delta t'$ is the preset acceptable delay threshold. If it holds, the node N_i can compute

$$R'_A = r_i^{-1} Y_i \tag{21}$$

and verify the legitimacy of N_A's identity and the authenticity of N_A's public key P_A by checking whether

$$h(R'_{A.x} \| B_{i.x} \| C_{i.x} \| t_A) = z_i. \tag{22}$$

If Equation (22) holds, the low-power node N_i can further derive the group secret key

$$k'_G = h(R'_{A.x} \| z_1 \| z_2 \| \ldots \| z_n \| t_A) \tag{23}$$

and verify the validity of the group secret key by checking whether

$$h(k'_G \| I_A) = m \tag{24}$$

If Equation (24) holds, k'_G is the group secret key shared among the powerful node and all participating low-power nodes.

In Figure 3, we give a simple example to demonstrate our authenticated group key agreement phase as follows. Suppose there are three low-power nodes N_1, N_2 and N_3 that want to agree on a group secret key shared among them, where $N = \{N_1, N_2, N_3\}$. After performing this phase, each low-power node can be given a group key k_G.

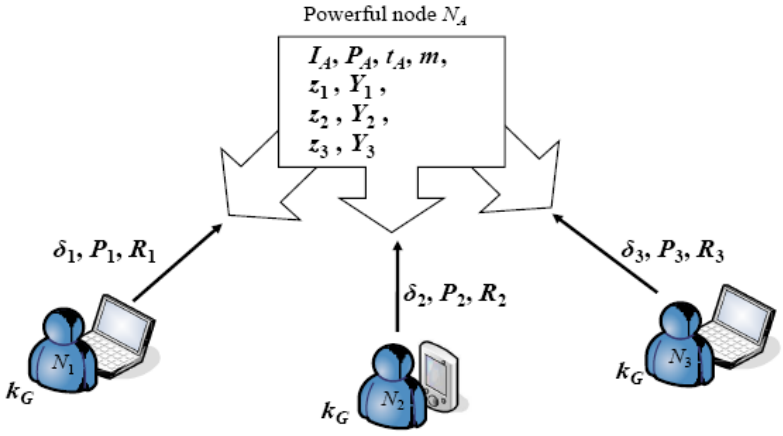

FIGURE 3. Example of the proposed authenticated group key agreement phase

2.4. **Node leaving phase.** When a low-power node N_i wants to leave the group, the remaining nodes must update the group secret key for ensuring the confidentiality of the future communications. The procedure of the group secret key updating is described below (as depicted in Figure 4).

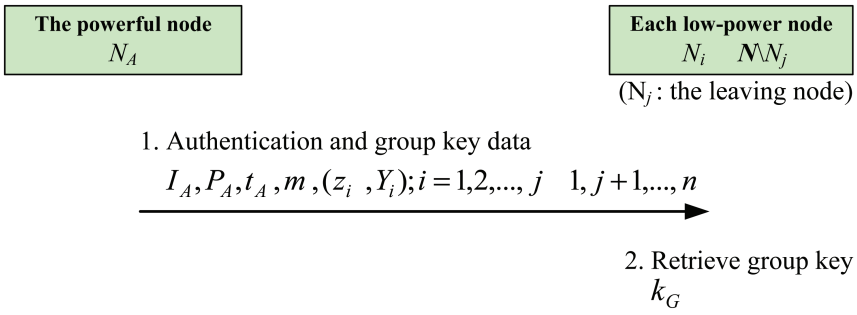

FIGURE 4. Node leaving phase

Step 1: The powerful node N_A computes

$$R'_A = r'_A G, \tag{25}$$
$$z'_i = h(R'_{A.x} \| B'_{i.x} \| C'_{i.x} \| t'_A), \tag{26}$$
$$Y'_i = r'_A R_i, \tag{27}$$
$$k'_G = h(R'_{A.x} \| z'_i \| z'_2 \| \ldots \| z'_{j-1} \| z'_{j+1} \| \ldots \| z'_n \| t'_A), \tag{28}$$
$$m' = h(k'_G \| I_A), \tag{29}$$

where t'_A is the current timestamp of N_A, $r'_A \in_R Z_q$, and $i = 1, 2, \ldots, j-1, j+1, \ldots, n$. The node N_A then broadcasts $\{I_A, P_A, t'_A, m', (Z'_i, Y'_i); i = 1, 2, \ldots, j-1, j+1, \ldots, n\}$ to the remaining low-power nodes N_i's, where $N_i \in N\backslash N_j$. Note that k'_G is the shared group secret key.

Step 2: On receiving $\{I_A, P_A, t'_A, m', (Z'_i, Y'_i); i = 1, 2, \ldots, j-1, j+1, \ldots, n\}$ from the powerful node N_A, each low-power node $N_i \in N\backslash N_j$ checks whether $t''_i - t'_A \leq \Delta t''$, where t''_i is the timestamp of receiving $\{I_A, P_A, t'_A, m', (Z'_i, Y'_i); i = 1, 2, \ldots, j-1, j+1, \ldots, n\}$ and $\Delta t''$ is the preset acceptable delay threshold. If it holds, the low-power node $N_i \in N\backslash N_j$ computes

$$R''_A = r_i^{-1} Y'_i \tag{30}$$

and verify the legitimacy of N_A's identity and the authenticity of N_A's public key P_A by checking whether

$$h(R''_{A.x} \| B_{i.x} \| C_{i.x} \| t'_A) = z'_i. \tag{31}$$

If Equation (31) holds, the low-power node N_i can further derive the group secret key

$$k''_G = h(R''_{A.x} \| z'_1 \| z'_2 \| \ldots \| z'_{j-1} \| z'_{j+1} \| \ldots \| z'_n \| t'_A) \tag{32}$$

and verify the validity of the group secret key by checking whether

$$h(k''_G \| I_A) = m'. \tag{33}$$

If Equation (33) holds, k''_G is the group secret key shared among the powerful node and all participating low-power nodes.

2.5. **Node joining phase.** When a low-power node N_{n+1} wants to join the group in progress, he needs to obtain the group secret key. All participant nodes and the new node N_{n+1} cooperates with each to perform the following steps (see also Figure 5) to generate a new group secret key shared among them.

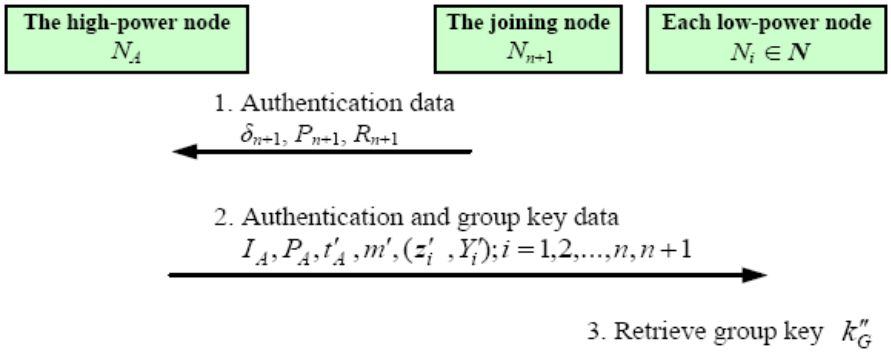

FIGURE 5. Node joining phase

Step 1: The N_{n+1} chooses $r_{n+1} \in_R Z_q$ to compute r_{n+1}^{-1},

$$R_{n+1} = r_{n+1}G, \tag{34}$$
$$B_{n+1} = x_{n+1}(P_A + P_{A.x} + I_A)P_{SA}), \tag{35}$$
$$C_{n+1} = r_{n+1}(P_A + (P_{A.x} + I_A)P_{SA}), \tag{36}$$
$$a_{n+1} = h(B_{(n+1).x} \| C_{(n+1).x} \| I_A \| I_{n+1} \| t_{n+1}), \tag{37}$$
$$\delta_{n+1} = (a_{n+1} \| I_{n+1} \| t_{n+1}) \oplus h(C_{(n+1).x}), \tag{38}$$

where t_{n+1} is the current timestamp of low-power node N_{n+1}. Finally, N_{n+1} sends $\{\delta_{n+1}, P_{n+1}, R_{n+1}\}$ to N_A.

Step 2: The powerful node N_A verifies legitimacy of N_{n+1}'s identity and generates group secret key by performing the following sub-steps.

Step 2-1: On receiving $\{\delta_{n+1}, P_{n+1}, R_{n+1}\}$ from N_{n+1}, the powerful node N_A computes

$$B'_{n+1} = x_A(P_{n+1} + (P_{(n+1).x} + I_{n+1})P_{SA}), \tag{39}$$
$$C'_{n+1} = x_A R_{n+1}, \tag{40}$$
$$(a_{n+1} \| I_{n+1} \| t_{n+1}) = \delta_{n+1} \oplus h(C'_{(n+1).x}), \tag{41}$$

and checks whether $t'_{n+1} - t_{n+1} \leq \Delta t$, where t'_{n+1} is the timestamp of receiving $\{\delta_{n+1}, P_{n+1}, R_{n+1}\}$ and Δt is the preset acceptable delay threshold. If it holds, the powerful node N_A node verifies the legitimacy of the low-power node N_{n+1} by checking whether

$$h(B'_{(n+1).x} \| C'_{(n+1).x} \| I_A \| I_{n+1} \| t_{n+1}) = a_{n+1}. \tag{42}$$

If Equation (42) does not hold, N_A requests N_{n+1} to re-send valid $\{\delta_{n+1}, P_{n+1}, R_{n+1}\}$. Otherwise, both of the identity authentication and the authenticity public key P_{n+1} for N_{n+1} are verified.

Step 2-2: The powerful node N_A computes

$$R'_A = r'_A G, \tag{43}$$
$$z'_i = h(R'_{A.x} \| B'_{i.x} \| C'_{i.x} \| t'_A), \tag{44}$$
$$k'_G = h(R'_{A.x} \| z'_1 \| z'_2 \| \ldots \| z'_{n+1} \| t'_A), \tag{45}$$
$$m' = h(k'_G \| I_A) \tag{46}$$
$$Y'_i = r'_A R_i, \tag{47}$$

where t'_A is the current timestamp of N_A, $r'_A \in_R Z_q$, and $i = 1, 2, \ldots, n, n+1$. Furthermore, N_A broadcasts $\{I_A, P_A, t'_A, m', (z'_i, Y'_i); i = 1, 2, \ldots, n, n+1\}$ to all low-power nodes. Note that k'_G is the shared group secret key.

Step 3: On receiving $\{I_A, P_A, t'_A, m', (z'_i, Y'_i); i = 1, 2, \ldots, n, n+1\}$ from the powerful node N_A, each low-power node $N_i \in \boldsymbol{N} \cup N_{n+1}$ checks whether $t''_i - t'_A \leq \Delta t''$, where t''_i is the timestamp of receiving $\{I_A, P_A, t'_A, m', (z'_i, Y'_i); i = 1, 2, \ldots, n, n+1\}$ and $\Delta t''$ is the preset acceptable delay threshold. If it holds, the low-power node $N_i \in \boldsymbol{N} \cup N_{n+1}$ computes

$$R''_A = r_i^{-1} Y'_i \tag{48}$$

and verify the legitimacy of N_A's identity and the authenticity of N_A's public key P_A by checking whether

$$h(R''_{A.x} \| B_{i.x} \| C_{i.x} \| t'_A) = z'_i. \tag{49}$$

If Equation (48) holds, the low-power node N_i can further derive the group secret key

$$k''_G = h(R''_{A.x} \| z'_1 \| z'_2 \| \ldots \| z'_n \| z'_{n+1} \| t'_A) \tag{50}$$

and verify the validity of the group secret key by checking whether

$$h(k''_G \| I_A) = m' \tag{51}$$

If Equation (51) holds, k''_G is the group secret key shared among the powerful node and all participating low-power nodes.

2.6. **Correctness of the proposed scheme.** The correctness of the proposed scheme is shown in the following theorems.

Theorem 2.1. *In the mobile node registration phase, the mobile node N_i can verify the validity of the key pair by Equation (6).*

Proof: From Equations (1)-(5), the left-hand side of Equation (6) can be rewritten as:

$$\begin{aligned} x_i G &= (s_i + v_i) G \\ &= (h(r_i \| I_i) + (P_{i.x} + I_i) \cdot s_{SA}) G + V_i \\ &= V_i + h(r_i \| I_i) G + (P_{i.x} + I_i) \cdot s_{SA} G \\ &= P_i + (P_{i.x} + I_i) P_{SA} \end{aligned}$$

Theorem 2.2. *In the authenticated group key agreement phase, the powerful node N_A can verify the legitimacy of the low-power node N_i by Equation (15).*

Proof: From Equation (6), the right-hand side of Equation (12) can be rewritten as

$$B'_i = x_A(P_i + (P_{i.x} + I_i) P_{SA}) = x_A x_i G = x_i x_A G = x_i(P_A + (P_{A.x} + I_A) P_{SA}) = B_i. \tag{52}$$

From Equations (6) and (7), the right-hand side of Equation (13) can be rewritten as:

$$C'_i = x_A R_i = x_A r_i G = r_i x_A G = r_i(P_A + (P_{A.x} + I_A) P_{SA}) = C_i. \tag{53}$$

Therefore, the left-hand side of Equation (15) can be rewritten as:

$$h(B'_{i.x} \| C'_{i.x} \| I_A \| I_i \| t_i) = h(B_{i.x} \| C_{i.x} \| I_A \| I_i \| t_i) = a_i$$

Theorem 2.3. *In the authenticated group key agreement phase, the low-power node N_i can verify the legitimacy of N_A's identity and the authenticity of N_A's public key by Equation (22).*

Proof: From Equations (7) and (20), the left-hand side of Equation (21) can be rewritten as:

$$R'_A = r_i^{-1}Y_i = r_i^{-1}r_A R_i = r_i^{-1}r_A r_i G = r_A r_i G = R_A \tag{54}$$

Therefore, the left-hand side of Equation (22) can be rewritten as:

$$h(R'_{A.x}\|B_{i.x}\|C_{i.x}\|t_A) = h(R_{A.x}\|B'_{i.x}\|C'_{i.x}\|t_A) = z_i.$$

Theorem 2.4. *In the authenticated group key agreement phase, the low-power node N_i can derive the group secret key and verify the validity of the group secret key by Equation (24).*

Proof: From Equation (54), the right-hand side of Equation (23) can be rewritten as:

$$k'_G = h(R'_{A.x}\|z_1\|z_2\|\ldots\|z_n\|t_A) = h(R_{A.x}\|z_1\|z_2\|\ldots\|z_n\|t_A) = k_G.$$

Therefore, the left-hand side of Equation (24) can be rewritten as:

$$h(k'_G\|I_A) = h(k_G\|I_A) = m$$

3. **Security Analysis.** The security of the proposed scheme is based on the elliptic curve discrete logarithm problem (ECDLP) [30-32] and the one-way hash function (OWHF) assumption [33,34].

Elliptic curve discrete logarithm problem (ECDLP): We assume that the elliptic curve contains a large prime subgroup of order p ($\geq$ 160 bits) which is large enough to make solving discrete logarithms in the finite field GF(p) infeasible. Suppose we have two points P, Q of an elliptic curve and let $Q = aP$, where a is an integer. It is computationally infeasible to find an integer a from $Q = aP$.

One way hash function (OWHF) assumption: If a hash function h is one-way, it must satisfy the following conditions:

(1) It is computationally infeasible to find a message M from its hash value $h(M)$.
(2) For any message M_1, it is computationally infeasible to find another message M_2 such that $h(M_2) = h(M_1)$.
(3) It is computationally infeasible to find a pair of different messages M_1 and M_2 such that $h(M_1) = h(M_2)$.

In the following, we present a detailed analysis on the security of the proposed scheme. The proposed scheme satisfies the following security requirements:

(1) **Confidentiality of Private Keys:** Consider the scenario of a compromising attack that an adversary or a malicious registered node attempts to derive SA's private key s_{SA}. With the knowledge of SA's public key $P_{SA} = s_{SA}G$, the adversary will face the ECDLP to derive s_{SA}. Only if he can derive r_i first, a malicious registered node can successfully compromise SA's private key s_{SA} with the SA's sending s_i by Equation (4). However, it is computationally infeasible to derive r_i from Equation (3) under the ECDLP and the OWHF assumption.

Similarly, consider the scenario of a compromising attack that a malicious adversary (including any registered node) attempts to derive node's private key x_i. Since the node's private key x_i is computed by Equation (5), the adversary will face the ECDLP to derive v_i from Equation (2). The private key satisfies the verification of Equation (6). With knowledge of $\{P_i, I_i, P_{SA}\}$, it is computationally infeasible to derive x_i under the ECDLP.

Consider the scenario of an attack that an adversary attempts to derive nodes' private keys x_i by the intercepted messages $\{I_i, P_i, R_i, t_i, a_i\}$'s and $\{I_A, P_A, t_A, m, (z, Y_i, I_i);\ i = 1, 2, \ldots, n\}$. From Equation (8) and Equation (10), the adversary will face the ECDLP and OWHF assumption to compromise the private key x_i. Similarly,

the adversary cannot derive the private key x_A from Equation (12), Equation (13) and Equation (17).

(2) **Entity Authentication:** The proposed scheme provides mutual authentication for verifying the legitimacies of the powerful node and the low-power nodes with each other. To authenticate the legitimacy of the participating low-power node N_i, the powerful node can check its legitimacy by Equation (15). Since $B_i = x_i(P_A + (P_{A.x} + I_A)P_{SA}) = x_A(P_i + (P_{i.x} + I_i)P_{SA}) = B'_i$ and $C_i = r_i(P_A + (P_{A.x} + I_A)P_{SA}) = x_A R_i = C'_i$, the adversary can successfully generate a valid a_i for cheating the powerful node only if he knows x_i or x_A. Security of the private keys is based on the ECDLP and the OWHF assumptions as analyzed above. Since the timestamp t_i is included in a_i, the adversary cannot replay the intercepted messages to masquerade as a valid low-power node. This also implies the proposed scheme can withstand the impersonation attacks.

On the other hand, each low-power node N_i can authenticate the legitimacy of the powerful node by Equation (22). The adversary can successfully masquerade as the powerful-node for cheating any low-power node N_i if he can correctly derive B_i, C_i and R'_A. Security of B_i and C_i is protected under the ECDLP and the OWHF assumption as discussed above. It can be seen that the security of $R'_A = r_i^{-1}Y_i$ is also protected based on the ECDLP since the adversary will face the ECDLP to derive r_i from R_i and then use r_i to compute R'_A.

(3) **Authenticity of Public Keys:** Seeing that a valid public key P_i with respect to x_i and I_i has to satisfy the check of Equation (6), a malicious adversary N_{adv} (including any registered node) may attempt to forge a valid pair $(I_{adv}, x_{adv}, P_{adv})$ to satisfy this verification equality. Consider the scenario that an adversary N_{adv} attempts to choose an identity information I_{adv} and try to generate a valid certificateless private and public key pair (x_{adv}, P_{adv}) without the assistant of SA. The adversary can first arbitrarily choose his identifier I_{adv} and private key x_{adv}, and then tries to compute the corresponding public key P_{adv} such that $P_{adv} + P_{adv.x}P_{SA} = x_{adv}G - I_{adv}P_{SA}$. It can be seen that the adversary will face the intractability of the ECDLP to derive $P_{adv.x}$ and P_{adv} from this equation. Similarly, the adversary might first determine (I_{adv}, P_{adv}), and then try to derive x_{adv} to satisfy above verification equality. It is obvious to see that x_{adv} is protected under the ECDLP. What is more, to generate a valid I_{adv} with the arbitrarily chosen x_{adv} and P_{adv}, the adversary will be confronted with the difficulty of the ECDLP.

(4) **Confidentiality of the Established Group Secret Key:** In the proposed scheme, the group secret key k_G is generated by Equation (18). Only one secret variable $R_{A.x}$ is contributed to key generation. The adversary can successfully compromise R_A for deriving k_G only if he knows r_i or r_A due to $R_A = r_i^{-1}Y_i = r_i^{-1}(r_A r_i G) = r_A G$. Compromising r_i from R_i or r_A from Y_i is an ECDLP. On the other hand, if the adversary attempts to derive k_G from the intercepted message m by Equation (19), he will face the intractability of reversing the one-way hash function (i.e., OWHF problem). Hence, the confidentiality of the group secret key is protected under the ECDLP or OWHF assumption.

(5) **Confirmation of the Established Group Secret Key:** In addition, the proposed scheme provides explicit key authentication (also called key confirmation) in such a way that all participating low-power nodes can explicitly verify the authenticity of the established group secret key. It can see that the message m is regarded as an authenticator by Equation (19) for this purpose. If the group secret key k_G is not correctly computed by Equation (18), it will fail to the verification of m by Equation

(24). And if it holds, k'_G is the group secret key shared among all participating low-power nodes. All participating low-power nodes can explicitly verify the authenticity of the established group secret key.

(6) **Group Key Contribution:** We will show that the proposed scheme is a contributory key agreement one which allows every participating low-power nodes to contribute their shares to the group key generation. It can be seen that the group secret key is computed by Equation (18). The secret random number r_i is secretly determined by a low-power node N_i, and hence contributed to the group key generation. This means that each low-power node equally contributes to the group secret key and guarantees its freshness in each group secret key construction, that is to say, no participant node can predetermine the group secret key. Hence, the proposed scheme is a contributory group key agreement one.

(7) **Forward Secrecy:** The forward secrecy guarantees that an adversary who compromises a private key(s) or one group secret key must not reveal previously established group secret keys. For example, when a low-power node wants to join a group, forward secrecy must be achieved to prevent the new member from accessing the previous group communications. As mentioned of the proposed scheme, the group secret key k_G is generated by Equation (18). Compromising the private key x_i (or x_A) only help to derive B_i and C_i. The group secret key is still protected by the secret R_A. It is easy to see that compromising r_i from Equation (7) or r_A from Equation (20) is an ECDLP. Hence, the adversary cannot derive any one group secret key with the compromised private keys.

Consider the scenario that the adversary compromised one group secret key attempts to derive any one previously established group secret key. Since the proposed scheme is a contributory one as mentioned above, the group secret key for distinct session will be refreshed by the random secret values. The group secret keys can be regarded as a random number generated by all participating nodes. Hence, the adversary knowing one group secret key cannot derive previously established one, which implies the forward secrecy is achieved.

(8) **User Anonymity:** The real identity information I_i of the participating low-power node N_i is encrypted with C_i by Equation (8). In message δ_i of the proposed scheme, identities are encrypted so that no identity-related information is leaked. The powerful node can decrypt I_i on the receipt of message δ_i and then recognize the identity of the participating low-power node N_i. Any adversary who eavesdrops on the communication channel and wants to recover the identity of the participating low-power node N_i faces the intractability of the ECDLP and OWHF assumption. Without the powerful node private key x_A or the low-power node private key x_i, the adversary cannot rederive I_i from Equation (14). Therefore, user anonymity is achieved through using an encrypted message δ_i.

We use Table 1 to show that our functionality comparison with the related scheme. From Table 1, we can see that the proposed scheme is a certificateless contributory key agreement one, while the other two schemes are certificate-based ones. The proposed scheme will have the merits of the certificateless public keys. Bresson et al.'s and Tseng's schemes [12,16] are all insecure against the impersonation attack, since their transmitted messages can be replayed by the adversary. Hence, they cannot achieve the mutual authentication. As we analyzed above, the proposed scheme are secure against the impersonation attack and achieves the mutual authentication. Furthermore, the proposed scheme satisfies the security requirement of user anonymity. Considering the security of the established group secret key, the proposed scheme and Tseng's scheme can achieve

contributory group key agreement, forward secrecy and key confirmation, while Bresson et al.'s scheme cannot. Key confirmation of Tseng's scheme can be implicitly achieved by checking the correctness of all variables contributed to group key generation, while that of the proposed scheme are explicitly achieved by a key authenticator. We also considered and proposed the group key updating mechanisms for node leaving or joining in our proposed scheme. Moreover, the underlying cryptographic assumption of the proposed scheme is elliptic curve discrete logarithm problem, while that of Bresson et al.'s [12] and Tseng's schemes is discrete logarithm (DL) problem. The proposed scheme is hence more secure than the other two schemes under the same key size.

Table 1. Comparisons of security properties

	Bresson et al.'s scheme [12]	Tseng's scheme [16]	The Proposed
Public keys	Certificate-based	Certificate-based	Certificateless
Mutual authentication	No	No	Yes
User anonymity	No	No	Yes
Impersonation attack resistance	No	No	Yes
Contributory group key agreement	No	Yes	Yes
Forward secrecy	No	Yes	Yes
Key confirmation	No	Implicit	Explicit
Group key updating (when member joins or leaves)	Yes	No	Yes
Certificateless	No	No	Yes
Underlying cryptographic assumption	DLP	DLP	ECDLP

4. **Performance Evaluations.** In this section, we evaluate the performance of our proposed scheme in terms of the computational complexities and the communication overheads. For convenience, we first define the following notations:

T_{EM}: the time for computing a point multiplication operation over an elliptic curve;
T_{EA}: the time for computing a point addition operation over an elliptic curve;
T_{MM}: the time for computing a modular multiplication;
T_{EXP}: the time for computing a modular exponentiation;
T_{INV}: the time for computing a modular inversion;
T_H: the time for computing a secure one-way hash function h;
T_{SIG}: the time for generating a signature;
T_{VER}: the time for verifying a signature;
n: the number of low-power nodes that want to agree on a group secret key shared among them;
$|a|$: the bit-length of a variable a.

Note that the time for computing a modular addition and that for XOR function are ignored here for that they are negligible as compared to the other complexities measures. From [35-38], the time complexities can be respectively regarded as $T_{EM} \approx 29T_{MM}$, $T_{EA} \approx 0.12T_{MM}$, $T_{EXP} \approx 240T_{MM}$, $T_{INV} \approx 10T_{MM}$ and $T_H \approx 4T_{MM}$.

First, we discuss the computational complexities of low-power nodes in our proposed scheme. In Step 1, each low-power node N_i computes r_i^{-1} and Equation (7) to Equation (11). The computational complexities for Step 1 are $4T_{EM} + T_{EA} + 2T_H + T_{INV}$. In Step 3, the low-power node N_i computes and verifies Equation (21) to Equation (24). Step 3 requires $T_{EM} + 3T_H$. Therefore, the computational complexities for each low-power

node are $5T_{EM} + T_{EA} + 5T_H + T_{INV}$. In the following, we consider the computational complexities of the powerful node in our proposed scheme. In Step 2, the powerful node computes and verifies Equation (12) to Equation (20). Thus, computational complexities for the powerful node are $(4n + 1)T_{EM} + nT_{EA} + (3n + 2)T_H$.

Comparisons of the computational complexities among the proposed, Bresson et al.'s, and Tseng's schemes are given in Table 2. In Bresson et al.'s and Tseng's schemes, all public keys are certificate-based ones. This means that the authenticity of all public keys will be verified by checking the validity of extra public key certificates issued by a certification authority CA. The proposed scheme uses certificateless public keys and hence gains performance efficiency in computational complexities due to no certificate verification. For simplicity, we assume that the public key certificates are implemented by ElGamal signature scheme [39] in Bresson et al.'s and Tseng's schemes. The digital signature used in both schemes is also assumed to be implemented by ElGamal signature [39]. From Table 2, it can see that the computational complexities for each low-power node are independent on the number of the low-power nodes in Bresson et al.'s and the proposed schemes, but not in Tseng's scheme. Computational complexities for the powerful node in these three schemes are all dependent on the number of the low-power nodes, but the proposed scheme requires lower computational complexities. In summary, the proposed scheme is more efficient than Bresson et al.'s and Tseng's schemes in computational complexities.

Table 2. Comparisons of computational complexities

	Bresson et al.'s scheme [12]	Tseng's scheme [16]	The Proposed scheme
Each low-power node	$2T_{EXP} + 2T_H$ $+T^{\ddagger}_{VER} + T^{\dagger}_{SIG}$ $\approx 1461T_{MM}$	$3T_{EXP} + nT_{MM} + T_H +$ $T_{INV} + T^{\ddagger}_{VER} + T^{\dagger}_{SIG}$ $\approx (n + 1707)T_{MM}$	$5T_{EM} + T_{EA}$ $+5T_H + TINV$ $\approx 166.12T_{MM}$
Powerful node	$nT_{EXP} + (n + 1)T_H$ $+2nT^{\ddagger}_{VER}$ $\approx (1686n + 4)T_{MM}$	$(2n + 1)T_{EXP} + nT_{MM}$ $+T_H + 2nT^{\ddagger}_{VER}$ $\approx (1923n + 244)T_{MM}$	$(4n + 1)T_{EM} + nT_{EA}$ $+(3n + 2)T_H$ $\approx (128.12n + 37)T_{MM}$

Remark

†The computational complexity for generating a signature is $T_{SIG} = T_{EXP} + 2T_{MM} + T_{INV} + T_{MA} \approx 252T_{MM}$ according to ElGamal signature scheme [39].

‡The computational complexity for verifying a signature/public key certificate is $T_{VER} = 3T_{EXP} + T_{MM} \approx 721T_{MM}$ according to ElGamal signature scheme [39].

Considering the communication overheads in the three schemes, we let the adopted one-way hash function h be SHA-1 [40] (the bit length of the output is 160 bits), $|p'| = 1024$ bits, $|q'| = 160$ bits, $|p| = |q| = 163$ bits respectively. For simplicity, the counter c (used in Bresson et al.'s scheme [12]), the timestamp t (used in the proposed scheme), and the identity I are all assumed to be 160 bits.

In our proposed scheme, each low-power node N_i sends $\{\delta_i, P_i, R_i\}$ to the powerful node N_A via uni-cast communication. Thus, the communication overheads sent by each low-power node are $|I| + 4|p| + |h| + |t|$. In Step 2, the powerful node N_A broadcasts $\{I_A, P_A, t_A, m, (z_i, Y_i); i = 1, 2, \ldots, n\}$ to low-power nodes. The communication overheads sent by the powerful node are $|I|+2(n+1)|p|+(n+1)|h|+|t|$, where $|h|$ is the bit-length of the adopted hash function. Table 3 shows comparisons of the communication overheads. Since Bresson et al.'s and Tseng's schemes are certificate-based ones, they require 2048 bits (i.e., $2|p'|$ bits) for transmitting an extra public key certificate which is assumed to be implemented by ElGamal signature scheme [39]. As seen from Table 3, the communication overheads sent by each low-power node in Bresson et al.'s and Tseng's schemes are larger

than those in the proposed scheme. The communication overheads sent by the powerful node in the proposed scheme are larger than those in Bresson et al.'s scheme, since it requires extra communication overheads for achieving contributory group key agreement, key confirmation and mutual authentication. If the proposed scheme provides the same security requirements as mentioned in Bresson et al.'s scheme [12], the communication overheads of the powerful node are only $(326n + 486)$ bits (i.e., the powerful node broadcasts $\{(P_A, t_A, Y_i); i = 1, 2, \ldots, n\}$ instead of $\{I_A, P_A, t_A, m, (z_i, Y_i); i = 1, 2, \ldots, n\}$. All the schemes require the same number of rounds in the group key agreement. The Tseng's scheme does not consider and propose the group key updating mechanisms for node leaving or joining process. For the joining and leaving process of the proposed and Bresson et al.'s schemes, the communication overhead depends on how many mobile nodes join/leave in a given time period.

TABLE 3. Comparisons of communication overheads

	Bresson et al.'s scheme [12]	Tseng's scheme [16]	The Proposed scheme
Communication overheads sent by each low-power node	$\|I\| + 5\|p'\|^{\ddagger}$ ≈5280 bits	$\|I\| + 5\|p'\|^{\ddagger}$ ≈5280 bits	$\|I\| + 4\|p\| + \|h\| + \|t\|$ ≈1132 bits
Communication overheads sent by the powerful node	$(n+1)\|I\| + \|c\| + n\|h\| + 2\|p'\|^{\ddagger}$ $\approx 320n + 2368$ bits	$(n+1)\|I\|$ $+\|h\| + 2(n+1)\|p'\|^{\ddagger}$ $\approx 2208n + 2368$ bits	$\|I\| + 2(n+1)\|p\|$ $+(n+1)\|h\| + \|t\|$ $\approx 486n + 806$ bits
Number of rounds	2	2	2

Remark ‡The costs for transmitting each public key certificate/signature of the Bresson et al.'s scheme and the Tseng's scheme are assumed to be implemented with ElGamal signature [39], i.e., $2|p'| = 2048$ bits.

In summary, based on the elliptic curve cryptosystem and self-certified public key cryptosystem, the proposed scheme can achieve much more performance efficiency in saving both the computational complexity and the communicational cost.

5. **Conclusions.** We have proposed a certificateless authenticated group key agreement scheme with privacy-preservation based on elliptic curve discrete logarithms for resource-limited mobile devices. The proposed scheme not only keeps the fundamental security requirements for authenticated group key agreement scheme but also highlights privacy protection for group communication on an open network. To protect privacy, our scheme provides the property of user anonymity, the identities of group users, who are involved in the authenticated group key agreement, will not be traced by any adversary. Furthermore, the elliptic curve cryptosystem and self-certified public key cryptosystem are integrated into the proposed scheme to achieve performance efficiency in practices, so it is quite suitable to be used for resource-limited mobile devices.

Acknowledgment. We would like to thank anonymous referees for their valuable suggestions. This work was supported in part by the Chang Gung University Grant UARPD390121, Chang Gung Memorial Hospital Grant CMRPD390031, and in part by National Science Council under the grants NSC 100-2628-H-182-001-MY3 and NSC 100-2321-B-182A-003, and Taiwan Information Security Center (TWISC), NSC 100-2219-E-011-002.

REFERENCES

[1] D. Jing, A. Kurt and N. R. Cristina, Secure group communication in wireless mesh networks, *Ad Hoc Networks*, vol.7, no.8, pp.1563-1576, 2009.

[2] K. Matsumoto, A. Utani and H. Yamamoto, Adaptive and efficient routing algorithm for mobile Ad-Hoc sensor networks, *ICIC Express Letters*, vol.3, no.3(B), pp.825-832, 2009.

[3] M. Manulis and A. R. Sadeghi, Key agreement for heterogeneous mobile Ad-Hoc groups, *International Journal of Wireless and Mobile Computing*, vol.4, no.1, pp.17-30, 2010.
[4] C.-C. Chang, Y.-W. Lai and J.-H. Yang, An efficient authenticated encryption scheme based on elliptic curve cryptosystem for broadcast environments, *ICIC Express Letters*, vol.4, no.1, pp.95-99, 2010.
[5] H.-Y. Chien and T.-C. Wu, Efficient and scalable key agreement schemes for mobile RFID readers and servers, *International Journal of Innovative Computing, Information and Control*, vol.6, no.12, pp.5463-5472, 2010.
[6] E. Bresson, O. Chevassut and D. Pointcheval, Provably authenticated group Diffie-Hellman key exchange – The dynamic case, *Proc. of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*, pp.290-309, 2001.
[7] D. Seo and P. Sweeny, Simple authenticated key agreement algorithm, *IEE Electronics Letters*, vol.35, no.13, pp.1073-1074, 1999.
[8] G. Ateniese, M. Steiner and G. Tsudik, Authenticated group key agreement and friends, *Proc. of the 5th ACM Conference on Computer and Communications Security*, pp.17-26, 1998.
[9] G. Ateniese, M. Steniner and G. Tsudik, New multiparty authentication services and key agreement protocols, *IEEE Journal on Selected Areas in Communications*, vol.18, no.4, pp.628-639, 2000.
[10] E. Bresson, O. Chevassut, A. Essiari and D. Pointcheval, Mutual authentication and group key agreement for low-power mobile devices, *Computer Communications*, vol.27, no.17, pp.1730-1737, 2004.
[11] T. S. Wu, H. Y. Lin, C. L. Hsu and K. Y. Chang, Efficient verifier-based authenticated key agreement protocol for three parties, *International Journal of Innovative Computing, Information and Control*, vol.6, no.2, pp.755-762, 2010.
[12] E. Bresson, O. Chevassut, A. Essiari and D. Pointcheval, Multual authentication and group key agreement for low-power mobile devices, *Computer Communications*, vol.27, no.17, pp.1730-1737, 2004.
[13] J. Nam, S. Kim and D. Won, A weakness in the Bresson-Chevassut-Essiari-Pointcheval's group key agreement scheme for low-power mobile devices, *IEEE Communications Letters*, vol.9, no.5, pp.429-431, 2005.
[14] J. Nam, S. Kim and D. Won, DDH-based group key agreement in a mobile environment, *Journal of Systems and Software*, vol.78, no.1, pp.73-83, 2005.
[15] Y. M. Tseng, On the security of two group key agreement protocols for mobile devices, *Proc. of the International Workshop on Future Mobile and Ubiquitous Information Technologies*, pp.59-62, 2006.
[16] Y. M. Tseng, A secure authenticated group key agreement protocol for resource-limited mobile devices, *The Computer Journal*, vol.50, pp.41-52, 2007.
[17] S. S. M. Chow and K. K. R. Choo, Strongly-secure identity-based key agreement and anonymous extension, *Proc. of the 10th International Conference on the Information Security*, pp.203-220, 2007.
[18] C. F. Lu, T. C. Wu and C. L. Hsu, Certificateless authenticated group key agreement protocol for unbalanced wireless mobile networks, *WSEAS Transactions on Communications*, vol.11, no.8, pp.1145-1159, 2009.
[19] H. Y. Chien, ID-based key agreement with anonymity for Ad Hoc networks, *Proc. of the 2007 International Conference on Embedded and Ubiquitous Computing*, pp.333-345, 2007.
[20] W. H. Kim, E. K. Ryu, J. Y. Im and K. Y. Yoo, New conference key agreement protocol with user anonymity, *Computer Standards and Interfaces*, vol.27, no.2, pp.185-190, 2005.
[21] K. Mangipudi and R. Katti, A secure identification and key agreement protocol with user anonymity (SIKA), *Computers and Security*, vol.25, no.6, pp.420-425, 2006.
[22] R. C. Wang, W. S. Juang, C. C. Wu and C. L. Lei, A lightweight key agreement protocol with user anonymity in ubiquitous computing environments, *Proc. of the 2007 International Conference on Multimedia and Ubiquitous Engineering*, pp.313-318, 2007.
[23] Z. Z. Wan, K. Ren, W. Lou and B. Preneel, Anonymous ID-based group key agreement for wireless networks, *Proc. of the 2008 IEEE Wireless Communications and Networking Conference*, pp.2615-2620, 2008.
[24] H. Park, Z. Kim and K. Kim, Forward secure ID-based group key agreement protocol with anonymity, *Proc. of the 3rd International Conference on Emerging Security Information, Systems and Technologies*, pp.274-279, 2009.
[25] M. Girault, Self-certified public keys, *Proc. of Advances in Cryptology – EUROCRYPT'91*, pp.490-497, 1991.

[26] H. Petersen and P. Horster, Self-certified keys concepts and applications, *Proc. of Communications and Multimedia Security'97*, pp.102-116, 1997.
[27] W. J. Tsaur, Several security schemes constructed using ECC-based self-certified public key cryptosystems, *Applied Mathematics and Computation*, vol.168, no.1, pp.447-464, 2005.
[28] N. Koblitz, Elliptic curve cryptosystems, *Mathematics of Computation*, vol.48, no.177, pp.203-209, 1985.
[29] V. Miller, Use of elliptic curves in cryptography, *Proc. of Advances in Cryptology – CRYPTO'85*, pp.417-426, 1985.
[30] A. J. Menezes, P. C. Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
[31] IEEE P1363, *Standard Specifications for Public Key Cryptography*, IEEE Working Group, 2009.
[32] A. Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, 1993.
[33] I. Blake, G. Seroussi and N. Smart, *Elliptic Curves in Cryptography*, Cambridge University Press, 1999.
[34] W. Diffie and M. Hellman, New directions in cryptography, *IEEE Transactions on Information Theory*, vol.22, no.6, pp.644-654, 1976.
[35] N. Koblitz, A. Menezes and S. Vanstone, The state of elliptic curve cryptography, *Designs, Codes and Cryptography*, vol.19, no.2, pp.173-193, 2000.
[36] T. S. Chen, E. T. Hsu and Y. L. Yu, A new elliptic curve undeniable signature scheme, *International mathematical Journal*, vol.1, no.31, pp.1529-1536, 2006.
[37] D. D. Hankerson, J. L. Hernandez and A. Menezes, Software implementation of elliptic curve cryptography over binary fields, *Proc. of Cryptographic Hardware and Embedded Systems – CHES'00*, pp.1-24, 2000.
[38] S. Contini, A. K. Lenstra and R. Steinfeld, VSH, an efficient and provable collision-resistant hash function, *Proc. of Advances in Cryptology – EUROCRYPT'06*, pp.165-182, 2006.
[39] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Transactions on Information Theory*, vol.31, no.4, pp.469-472, 1985.
[40] NIST FIPS 180-3, *Secure Hash Standard (SHS)*, NIST, 2007.