KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS VOL. 9, NO. 12, Dec. 2015

Multi-party Password-Authenticated Key Exchange Scheme with Privacy Preservation for Mobile Environment

Chung-Fu Lu
Department of Information Management, Chihlee University of Technology
New Taipei City, Taiwan, R.O.C.
[e-mail: peter61@mail.chihlee.edu.tw]

Received April 17, 2014; revised April 5, 2015; revised October 17, 2015; accepted October 19, 2015; published December 31, 2015

Abstract

Communications among multi-party must be fast, cost effective and secure. Today's computing environments such as internet conference, multi-user games and many more applications involve multi-party. All participants together establish a common session key to enable multi-party and secure exchange of messages. Multi-party password-based authenticated key exchange scheme allows users to communicate securely over an insecure network by using easy-to-remember password. Kwon *et al*. proposed a practical three-party password-based authenticated key exchange (3-PAKE) scheme to allow two users to establish a session key through a server without pre-sharing a password between users. However, Kwon *et al*.'s scheme cannot meet the security requirements of key authentication, key confirmation and anonymity. In this paper, we present a novel, simple and efficient multi-party password-based authenticated key exchange (*M*-PAKE) scheme based on the elliptic curve cryptography for mobile environment. Our proposed scheme only requires two round-messages. Furthermore, the proposed scheme not only satisfies security requirements for PAKE scheme but also achieves efficient computation and communication.

Keywords: Authenticated key exchange, password-based, multi-party, cryptography

http://dx.doi.org/10.3837/tiis.2015.12.022 **ISSN : 1976-7277**

1. Introduction

Password authenticated key exchange (PAKE) studies how to establish secure communications between two or more parties solely based on their password. The key challenge with password-based schemes is that the memorable password, associated with each user, has low entropy. It is not easy to protect the password information against dictionary attacks whereby an adversary ends up with the correct password after exhaustively testing all possible passwords against known password verifiers. Therefore, the intrinsic problem in designing PAKE schemes is to preserve password security against dictionary attacks.

In 1992, Bellovin and Merritt first proposed the two-party PAKE protocol (2-PAKE) [1], where two entities A and B share a human-memorable password to establish a common session key. Because 2-PAKE protocol is not suitable for the large peer-to-peer architecture, many researchers on the topic have concentrated on proposing schemes that either extend Bellovin and Merritt's scheme into three-party applications or have better performance. Three-party password-based authenticated key exchange protocol (3-PAKE) is a simple and an important mechanism that allows each user to choose his own password and to share with the server. In a 3-PAKE scheme, it requires a trusted server which shares an easy-to-remember password with each user. However, as a result of limited ability of memory of human, people prefer natural language phrases as their own secret passwords. This will make 3-PAKE scheme becomes vulnerable to password guessing attacks [2]. Furthermore, the number of transmission rounds and computational complexities are two important criteria of 3-PAKE for describing the system performance [3-5].

Based on Diffie-Hellman key exchange concept, Steiner *et al*. proposed a 3-PAKE protocol [6] in 1995. Thereafter, Ding *et al*. [2] and Sun *et al*. [7] pointed out that Steiner *et al*.'s scheme is vulnerable to undetectable on-line password guessing attacks. Moreover, Lin *et al*. [8] further showed that Steiner *et al*.'s scheme suffers not only undetectable on-line password guessing attacks but also off-line password guessing attacks. To eliminate these flaws, Sun *et al*. and Lin *et al*. separately utilized public key cryptographic technology to prevent undetectable on-line password guessing attacks and off-line password guessing attacks. However, the public key technologies need to take more computational complexities in current 3-PAKE protocol.

Chang and Chang proposed a robust and efficient 3-PAKE protocol by using trapdoor one-way function [9] in 2004. Later, Chen *et al*. [10] and Yoon *et al*. [11] pointed out that Chang and Chang's scheme cannot resist undetectable on-line password guessing attacks and proposed an enhancement schemes to solve the security problem separately. However, Lo and Yeh [12] pointed out that both of these two schemes proposed by Chen *et al*. and Yoon *et al*. are still vulnerable against the undetectable on-line password guessing attacks.

In 2005, Abdalla *et al*. proposed a formal security model of 3-PAKE with different passwords [13]. From the viewpoint of the rounds/computational complexities, Abdalla *et al*.'s scheme requires six rounds and more than 17 modular exponentiations per user in the standard model. To improve the efficiency of the above scheme, Abdalla *et al*. presented a tailor-made protocol [14]. But they fail to resist to undetectable on-line dictionary attack. The authors count this attack in the number of queries for message modifications which are limited to certain numbers.

In 2008, Kwon *et al*. proposed a password-based 3-PAKE scheme with different passwords that achieves forward secrecy in the standard model [15]. Their scheme requires four rounds to achieve authentication between users and the server. Besides, their scheme does not provide key authentication, key confirmation and user anonymity. In 2012, we proposed a PAKE scheme for multi-party setting to meet the above security requirements and the efficiency is greatly [16]. The latest survey of 3-PAKE issues is presented in [17-23].

With the emergence of mobile environment, conventional 3-PAKE protocols face two common problems. The first problem is that the server and users are not in the same domain, and therefore, the shared authenticated keys may be unknowingly compromised. In addition, conventional 3-PAKE protocols require higher on-line communication cost and computational cost during session key agreement, which can create excessive overheads for user using device with low computational capacity.

Despite recent researches aimed at reducing the computation and energy costs of public key operations/protocols, which are successfully applied in traditional wired networks, are not suitable in low - power devices, such as mobile networks/WSNs [24, 25]. Although RSA is well established, the elliptic curve cryptography (ECC) is still more commercial importance and has attracted attention because of a smaller key size, reducing storage, low on CPU consumption, and transmission requirements [26].

In this paper, we will propose a multi-party PAKE (M-PAKE) scheme based on the ECC for mobile environment. Our proposed scheme achieves better performance by requiring only two round-messages and meets security requirements. The proposed scheme is more efficient than previously proposed schemes in terms of the computational complexities and the communication costs. Furthermore, our proposed scheme provides security from entity authentication, confidentiality of private/session key, forward secrecy, user anonymity, key authentication, and key confirmation.

Organization of this paper is sketched as follows. Section 2, we revisit the password-based 3-PAKE scheme of Kwon *et al*. We then present our proposed scheme in Section 3. The security analysis and the performance evaluation will in Section 4. Finally, a conclusion is given in Section 5.

2. Revisiting Kwon *et al*.'s 3-PAKE scheme

In this section, we show that the 3-PAKE scheme [15] of Kwon *et al*. Their scheme requires four rounds to achieve authentication between users and the server.

Initialization. Each user $U_i \in U$ for $i \in \{1,2\}$ obtains pw_i in the beginning of the scheme by using a password generation algorithm $PG(1^k)$. Based on the decisional Diffie-Hellman assumption, let p' and q' be safe primes such that $p' = 2q' + 1$. Let g_1 and g_2 be generators of a finite cyclic group G having order q'. Let $H()$ be a hash function, $F()$ be a secure pseudorandom function family, and $MAC_K(m)$ be a message authentication code function, where m is a message and K is a key. Assume that each user U_i and server S have shared $PW_i = g_2^{H(U_i \| S \| pw_i)} \bmod p'$, the public information $(G, p', q', g_1, g_2, H(), F())$, and the identities of users exchanging a session key.

Round 1. Each user U_i chooses a random number $x_i \in Z_{q'}^*$, computes $X_{iS} = g_1^{x_i} \cdot PW_i \bmod p'$, and sends (I_i, X_{iS}) to the Server S, where I_i is the identity information of the user U_i. Then, S chooses random number $y_i \in Z_{q'}^*$, computes $X_{Si} = g_1^{y_i} \cdot PW_i \bmod p'$ for $i \in \{1,2\}$, and broadcasts (I_S, X_{S1}, X_{S2}), where I_S is the identity information of the server S.

Round 2. Upon receiving (I_S, X_{S1}, X_{S2}) from the server S, the user U_i computes $K_{iS} = (X_{Si} / PW_i)^{x_i} \bmod p'$ and $a_{iS} = MAC_{K_{iS}}(I_i \| I_S \| X_{iS} \| X_{Si})$. Then, U_i sends (I_i, a_{iS}) to S.

Round 3. Upon receiving (I_i, a_{iS}), S compares $a_{Si} = MAC_{K_{Si}}(I_i \| I_S \| X_{iS} \| X_{Si})$, where $K_{Si} = (X_{iS} / PW_i)^{y_i} \bmod p'$ for $i \in \{1,2\}$. If a_{Si} and a_{iS} are identical, a_{iS} is verified. If both a_{1S} and a_{2S} are verified, S chooses a random number $s \in Z_{q'}^*$, computes $Y_{Si} = (g_1^{x_{i+1}})^s \bmod p'$ and $a_{Si} = MAC_{K_{Si}}(I_i \| I_{i+1} \| Y_{Si})$, and sends $(I_S \| Y_{Si} \| a_{Si})$ to each user U_i.

Round 4. Upon receiving $(I_S \| Y_{Si} \| a_{Si})$, each user U_i compares a_{Si} with $MAC_{K_{iS}}(I_i \| I_S \| X_{iS} \| X_{Si})$. If a_{Si} and a_{iS} are identical, U_i computes $K_i = (Y_{Si})^{x_i} \bmod p'$ and the session key $sk_i = F_{K_i}(I_1 \| I_S \| I_2)$, where $F_{K_i}()$ is a secure pseudorandom function and $I_1 < I_2$. Both users U_1 and U_2 compute an identical session key $sk = sk_1 = sk_2$.

Kwon *et al.*'s scheme does not provide key authentication, key confirmation and user's anonymity. The identity I_i of user U_i is transmitted in plaintext. Accordingly, the user privacy can be intruded upon easily, especially in mobile environment. In terms of key confirmation, after the session key sk is distributed to each user U_i, Kwon *et al.*'s scheme is not convinced that U_i actually possesses the session key sk. In addition, for mobile environment the efficiency of authenticated key exchange should be one of the core considerations. Nevertheless, the modulus operation used in Kwon *et al.*'s scheme is expensive.

3. The Proposed Scheme

In this section, we present the proposed *M*-PAKE scheme with privacy preservation for mobile environment. The logical architecture for proposed *M*-PAKE scheme is shown in **Fig. 1**. Without loss of generality, let $U = \{U_1, U_2, ..., U_n\}$ be a set of *n* users, *S* be a trusted server, and $M = n + 1$ be the total amount of the communication parties. Using users' password $PW_1, PW_2, ..., PW_n$ secretly shared with server *S*, the users in the set *U* can cooperate to generate a valid session key. The notations used in the proposed *M*-PAKE scheme are listed in **Table 1.** The proposed *M*-PAKE scheme consists of three phases: the system setup, the user registration, and the multi-party PAKE. We outline these phases shown in the proposed scheme, and detailed descriptions of these phases are given below sub-sessions.

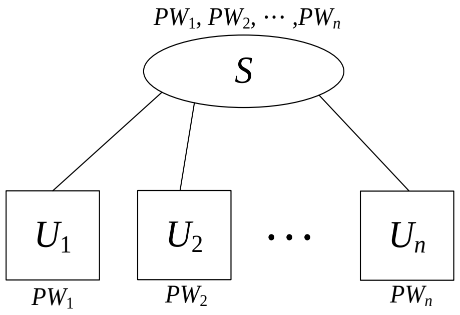

Fig. 1. Logical architecture for proposed *M*-PAKE scheme

Phase 1. System setup phase: The trusted server defines system parameters and generates his private/public key-pair. Finally, the trusted server publishes the system parameters and keeps private key secret.

Phase 2. User registration phase: Each user must register in trusted server before multi-party PAKE. The trusted server cooperates with the registering user to generate the shared password between the registering user and the trusted server.

Phase 3. Multi-party PAKE phase: Using only two round-messages, all participating users will cooperative with the trusted server to generate the secret session key.

- Each participating user sends his authenticator and session key contribution to trusted server**.** The trusted server can authenticate the legitimacy of all participating users and generate the session key derivation information.
- The trusted server sends his authenticator and session key information to each participating user. All participating users can authenticate the legitimacy of the server and explicitly verify the authenticity of the established session key.

Table 1. Notations

U_i	*i*th User
S	trusted server
I_i	identity information of user U_i
I_S	identity information of server S
pw_i	U_i 's password
PW_i	U_i 's password secretly shared with S
$H_j(\cdot)$	one-way hash functions, $j = \{1,2,\ldots,4\}$
$E_{pk}(\cdot)/D_{pk}(\cdot)$	symmetric encryption/decryption function with key *pk*

3.1 System setup phase

Initially, the server S determines a large prime p and a non-supersingular elliptic curve $EC_p(a,b)$ as $y^2 = x^3 + ax + b \pmod{p}$, where $a,b \in_R Z_p^*$ and $4a^3 + 27b^2 \bmod p \neq 0$. The server S further determines a large prime q and a base point G of order q over $EC_p(a,b)$, where q is a divisor of the number of points on the elliptic curve $EC_p(a,b)$. Let O be a point at infinity over $EC_p(a,b)$, $Q_{i.x}/Q_{i.y}$ be the x-coordinate/y-coordinate of the point Q_i, and H_1, H_2, H_3, H_4 be secure one-way hash functions that accepts a variable length input and produces a fixed length output which is over GF(q). The private and public keys for the server S are respectively defined as x_s and Y_S, where $x_S \in_R Z_q$ and $Y_S = x_s G$. Let E/D be the secure symmetric encryption/decryption function. Finally, the server S publishes $(p,q,EC_p(a,b),O,H_1,H_2,H_3,H_4,G,Y_S,E,D)$ while keeps x_s secret.

3.2 User registration phase

When a user U_i wants to use the multi-party PAKE service, he has to register beforehand to the trusted server S. The user U_i obtains pw_i at the start of the scheme by using a password generation algorithm $PG(1^l)$, where l is the bit length of password pw_i. When subscribing to the multi-party PAKE service, the user U_i will receive the $PW_i = H_1(I_i \| I_S \| pw_i)G$ secretly shared between the user and the server, the identity I_i and the public information $(p,q,EC_p(a,b),O,H_1,H_2,H_3,H_4,G,Y_S,E,D)$.

3.3 Multi-party PAKE phase

The multi-party PAKE phase requires only two round-messages. Without loss of generality, let $U = \{U_1, U_2, ..., U_n\}$ be the set of n users that want to agree on a secret session key shared among them. All the users will cooperative with a trusted server S to generate the secret session key. The procedure for the M-PAKE phase is stated as follows (as depicted in **Fig. 2**).

Step 1. Each user U_i chooses a random number $r_i \in Z_q^*$ and computes $R_i = r_i G$, $A_i = r_i Y_S$, $mac_i = H_2(A_{i.x} \| PW_{i,x} \| I_i \| t_i)$, $m_i = mac_i \| I_i$, $C_i = E_{A_{i.x}}(m_i)$. Finally, U_i sends his authenticator/session key contribution (R_i, C_i, t_i) to trusted server S, where t_i is the current timestamp.

Step 2. The trusted server S authenticates the legitimacy of all participating users and generates the session key derivation information by performing the following sub-steps.

Step 2-1. Upon receiving (R_i, C_i, t_i) from U_i at the time T_i, (for $i = 1,2,...,n$) S verifies the validity of the time interval between t_i and T_i. If $(T_i - t_i) \geq \Delta T$ then S rejects the request, where ΔT denotes the expected valid time interval for transmission delay.

Step 2-2. The server S computes $A_i = x_s R_i$, $m_i = D_{A_{i,x}}(C_i)$ and verifies the legitimacy of the user U_i. If $mac_i = H_2(A_{i.x} \| PW_{i.x} \| I_i \| t_i)$ does not hold, S rejects the request.

Step 2-3. The server S chooses a random number $r_S \in Z_q^*$ and computes $Y_{S,i} = r_s R_i$, $K = H_3(R_S \| Y_{(S,1).x} \| Y_{(S,2).x} \| \ldots \| Y_{(S,n).x} \| t_S)$, $\delta_i = H_4(A_{i.x} \| K \| I_S)$. Finally, S sends his authenticator/session key related information $t_S, (Y_{S,i}, \delta_i)|_{i=1,2,\ldots,n}$ to each user U_i, where t_S is the current timestamp.

Step 3. Upon receiving $t_S, (Y_{S,i}, \delta_i)|_{i=1,2,\ldots,n}$ at the time $T_i^{'}$, each user U_i verifies the validity of the time interval between t_S and $T_i^{'}$. If $(T_i^{'} - t_S) \geq \Delta T$, where ΔT denotes the expected valid time interval for transmission delay, then U_i rejects the request. If it holds, user U_i computes $R_S = r_i^{-1}(Y_{S,i})$, $K = H_3(R_S \| Y_{(S,1).x} \| Y_{(S,2).x} \| \ldots \| Y_{(S,n).x} \| t_S)$, and verifies $\delta_i \overset{?}{=} H_4(A_{i.x} \| K \| I_S)$. If it holds, U_i accepts the session key K. Otherwise, U_i rejects the request.

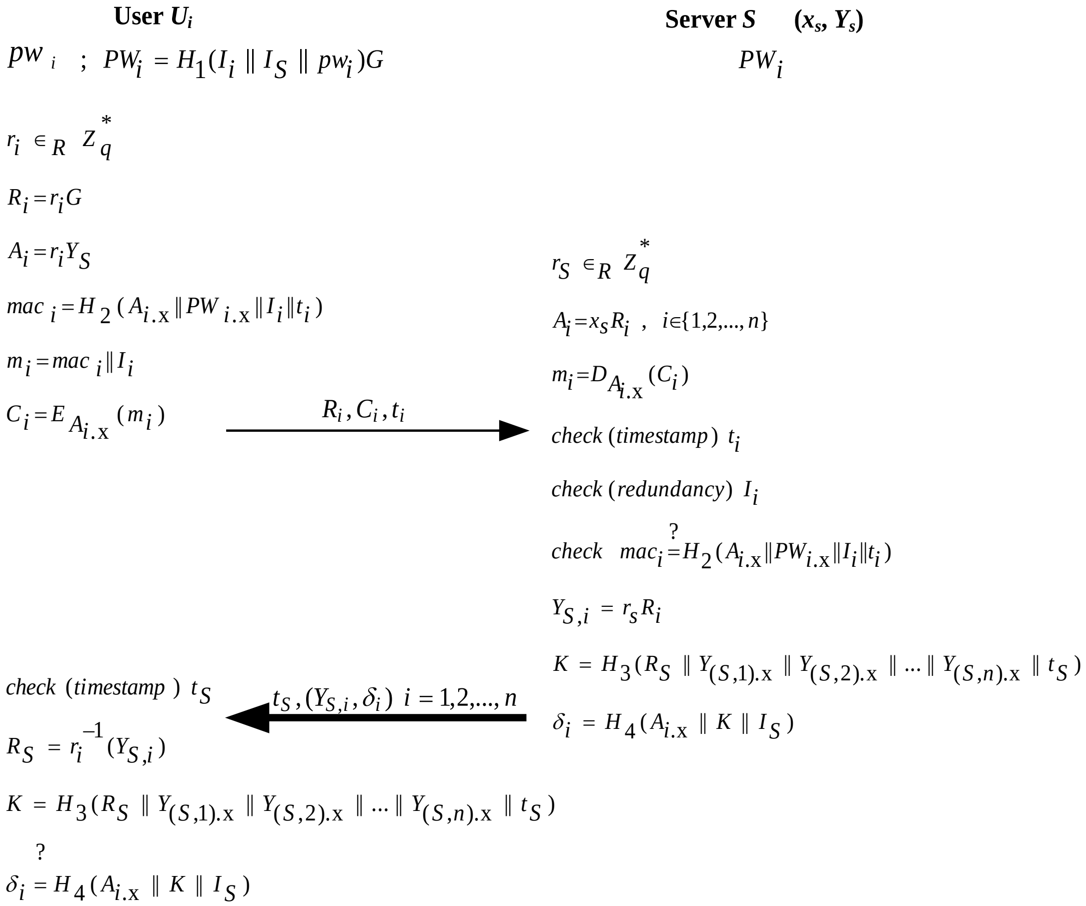

Fig. 2. The multi-party PAKE phase

4. Security Analysis and Performance Evaluation

4.1 Security analysis

The security of the proposed scheme is based on the elliptic curve discrete logarithm problem (ECDLP) [27-29] and the one-way hash function (OWHF) assumption [30, 31].

***Elliptic curve discrete logarithm problem (ECDLP)*:**

We assume that the elliptic curve contains a large prime subgroup of order p (>=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 $Q = xP$, where x is an integer. It is computationally infeasible to find an integer x from $Q = xP$.

***One way hash function (OWHF) assumption*:**

If a hash function h is one-way, it must satisfy the following conditions:

- It is computationally infeasible to find a message m from its hash value $h(m)$.
- For any message m_1, it is computationally infeasible to find another message m_2 such that $h(m_2) = h(m_1)$.
- 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 the analysis on the security of our proposed scheme. The proposed scheme can withstand possible attacks and satisfies the following security requirements:

(1) Entity authentication

The proposed scheme provides mutual authentication for verifying the server S and user U_i with each other. To authenticate the legitimacy of user U_i, the server can check its legitimacy by $mac_i \overset{?}{=} H_2(A_{i.x} \| PW_{i,x} \| I_i \| t_i)$. The adversary can successfully generate a valid mac_i for cheating the server only if he knows the user's password PW_i. Security of PW_i is based on the OWHF assumptions as analyzed above.

On the other hand, each user U_i can authenticate the legitimacy of the server by $\delta_i \overset{?}{=} H_4(A_{i.x} \| K \| I_S)$. The adversary can successfully masquerade as the server for cheating any user U_i if he can correctly derive A_i and PW_i. Security of A_i and PW_i is protected under the ECDLP and the OWHF assumption as discussed above.

(2) Confidentiality of private key

Consider the scenario of a compromising attack that an adversary attempts to derive server's private key x_S. With the knowledge of server's public key $Y_S = x_S G$, the adversary will face the ECDLP to derive x_S.

(3) Confidentiality of the established session key

In the proposed scheme, the session key K is generated by $K = H_3(R_S \| Y_{(S,1).x} \| Y_{(S,2).x} \| \ldots \| Y_{(S,n).x} \| t_S)$. Only one secret variable R_S is contributed to key generation. The adversary can successfully compromise R_S for deriving K only if he knows r_i or r_S due to $R_S = r_i^{-1}(Y_{S,i}) = r_i^{-1}(r_S R_i) = r_i^{-1}(r_S r_i G) = r_S G$. Compromising r_i from R_i or r_S from $Y_{S,i}$ is an ECDLP. On the other hand, if the adversary attempts to derive K from the intercepted message $\delta_i = H_4(A_{i.x} \| K \| I_S)$, he will face the intractability of reversing the one-way hash function (i.e. OWHF problem). Hence, the confidentiality of the session key is protected under the ECDLP or OWHF assumption.

(4) Confirmation of the established session key

In addition, the proposed scheme provides explicit key authentication (also called key confirmation) in such a way that all users can explicitly verify the authenticity of the established session key. It can see that the message δ_i is regarded as an authenticator by $\delta_i = H_4(A_{i.x} \| K \| I_S)$ for this purpose. If the session key K is not correctly computed by $K = H_3(R_S \| Y_{(S,1).x} \| Y_{(S,2).x} \| \cdots \| Y_{(S,n).x} \| t_S)$, it will fail to the verification of δ_i by $\delta_i \overset{?}{=} H_4(A_{i.x} \| K \| I_S)$. And if it holds, K is the session key shared among all participating users. All participating users can explicitly verify the authenticity of the established session key.

(5) Session key contribution

We will show that the proposed scheme is a contributory key agreement one which allows every participating users to contribute their shares to the session key generation. It can be seen that the session key is computed by $K = H_3(R_S \| Y_{(S,1).x} \| Y_{(S,2).x} \| \cdots \| Y_{(S,n).x} \| t_S)$. The secret random number r_i is secretly determined by user U_i, and hence contributed to the session key generation. This means that each user equally contributes to the session key and guarantees its freshness in each session key construction, that is to say, no participant user can predetermine the session key. Hence, the proposed scheme is a contributory key agreement one.

(6) Forward secrecy

The forward secrecy guarantees that an adversary who compromises a private key or one session key must not reveal previously established session keys. As mentioned of the proposed scheme, the session key K is generated by $K = H_3(R_S \| Y_{(S,1).x} \| Y_{(S,2).x} \| \cdots \| Y_{(S,n).x} \| t_S)$. The session key is protected by the secret R_S . It is easy to see that compromising r_S from $Y_{S,i} = r_s R_i$ is an ECDLP. Although the server’s private key x_s is disclosed for some reason, the proposed scheme can withstand the attack that any adversary with the knowledge of x_s attempts to derive one current session key. The adversary cannot compute K without knowing R_S . Hence, the adversary cannot derive any one session key with the compromised private key x_s .

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

(7) User anonymity

The user sends the request (R_i, C_i, t_i) to the server in each login. The adversary may analyze the login message. It is infeasible to derive the identity of the user from the login message, where $mac_i = H_2(A_{i.x} \| PW_{i,x} \| I_i \| t_i)$. Since the timestamp t_i is different for sessions and the identity I_i is protected by the one-way hash function. Therefore, the adversary cannot identify the person who wants to login.

The identity information I_i of the user U_i is encrypted with C_i. In encrypted message C_i of the proposed scheme, the identity I_i is encrypted so that no identity-related information is leaked. The server can decrypt I_i on the receipt of message C_i and then recognize the identity of the participating user U_i. Any adversary who eavesdrops on the communication channel and wants to recover the identity of the user U_i faces the intractability of the OWHF assumption. Therefore, user anonymity is achieved through using an encrypted message C_i.

(8) Replay attack and impersonation attack

This kind of replay attack, the attacker listens to communication between the sender and the receiver and then replays the same message of the user or the server. Our proposed scheme uses the timestamp to withstand replay attacks. Since the timestamp t_i or t_S is included in mac_i or K, the adversary cannot replay the intercepted messages to masquerade as a valid user or server. The attacker cannot work because he will fail the validity of the time interval $(T_i - t_i) \geq \Delta T$ or $(T_i^{'} - t_S) \geq \Delta T$. This also implies the proposed scheme can withstand the impersonation attacks.

On the other hand, the adversary impersonates as the legitimate user and forges the message using the information obtained from the scheme. The adversary needs to guess (A_i, mac_i, m_i) to masquerades as a legitimate user to forge a valid login. The adversary cannot obtain (A_i, mac_i, m_i) from intercepted communication information R_i, C_i and t_i. Therefore, our proposed scheme is secure against impersonation attack.

(9) Off-line dictionary attack

It is hard for any adversary to derive the user password pw_i or server private key x_s from recorded messages, because the adversary will face the OWHF assumption and the ECDLP.

4.2 Performance Evaluation

In this subsection, we will evaluate the performance of the proposed scheme and make comparison with related researches in **Table 2**. The computational complexities represent how many (or how heavy) cryptographic operations such as symmetric encryption or one-way hash function are adopted in the communication protocol. For simplicity, we denote the following notation to evaluate the performance of our proposed scheme and related researches:

T_{Mac}: the time for performing a strongly unforgeable MAC algorithm computation,

T_F: the time for performing a secure pseudorandom function computation,

T_H: the time for performing a one-way hash function computation ($T_H \approx 4\ T_{MUL}$),

$T_{EM/EA}$: the time for computing a point multiplication/addition operation over an elliptic curve

($T_{EM} \approx 29T_{MUL}$, $T_{EA} \approx 0.12T_{MUL}$);

$T_{MUL/EXP/INV}$: the time for computing a modular multiplication/exponentiation/inversion ($T_{EXP} \approx 240\ T_{MUL}$, $T_{INV} \approx 10\ T_{MUL}$);

$T_{SE/SD}$: the time for performing a symmetric encryption (SE)/decryption (SD) algorithm computation ($T_{SE} \approx T_H \approx 4\ T_{MUL}$, $T_{SD} \approx T_H \approx 4\ T_{MUL}$);

n: the number of participating users that want to agree on a secret session key shared among them;

$|a|$: the bit-length of a variable a.

Table 2. Performance comparisons of 3-PAKE scheme

		Proposed scheme (*M*-PAKE, *n* = 2)	**Kwon *et al.* [15]**	**Lu *et al.* [16] (*M*-PAKE, *n* = 2)**	**Farash *et al.* [19]**	**Wei *et al.* [23]**
computational complexities	user *i*	$4T_{EM} + T_{INV} + 4T_H + T_{SE}$	$4T_{EXP} + T_{MUL} + T_{INV} + 2T_{Mac} + T_H + T_F$	$4T_{EXP} + 5T_H + T_{INV}$	$3T_{EXP} + 7T_H$	$3T_{EXP} + 5T_H$
	server *S*	$2n\ T_{EM} + nT_{SD} + (2n+1)T_H$	$6T_{EXP} + 4T_{MUL} + 2T_{INV} + 4T_{Mac}$	$2n\ T_{EXP} + (3n+1)T_H$	$3T_{EXP} + 8T_H$	$5T_{EXP} + 8T_H$
communication overheads	user *i*	$2\|p\| + \|SE\| + \|t\|$	$2\|I\| + \|p'\| + \|Mac\|$	$\|p'\| + 2\|H\| + \|t\|$	$2\|I\| + \|p'\| + \|H\|$	$2\|I\| + \|p'\| + \|H\|$
	server *S*	$2n\|p\| + n\|H\| + \|t\|$	$2\|I\| + 3\|p'\| + 2\|Mac\|$	$n\|p'\| + n\|H\| + \|t\|$	$4\|p'\| + 2\|H\|$	$4\|p'\| + 2\|H\|$

Table 2 compares the total computation costs required by user and the server in the proposed protocol and that proposed by related researches. 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 [32-35], the time complexities can be respectively regarded as $T_{\mathrm{EM}} \approx 29T_{MUL}$, $T_{\mathrm{EA}} \approx 0.12T_{MUL}$, $T_{\mathrm{EXP}} \approx 240T_{MUL}$, $T_{\mathrm{INV}} \approx 10T_{MUL}$, and $T_{\mathrm{H}} \approx 4T_{MUL}$. To facilitate the comparisons in **Fig. 3**, we converted the costs of all operations into cost of T_{MUL}. The results of the comparisons indicate that the proposed scheme imposes significantly lower computational costs than previously proposed schemes.

Considering the communication overheads, we let the adopted one-way hash function be SHA-1 [36] (the bit length of the output is 160 bits), $|p'| = 1024$ bits, $|q'| = 160$ bits, $|p| = |q| = 163$ bits, respectively. The timestamp t, the identity, and the Mac value are all assumed to be 160 bits. We thus compared the size of messages transmitted using the proposed scheme and that proposed by related researches. **Fig. 4** presents the results. In the communication overheads of user i, the cost of the proposed scheme is 2*163+2*160+160 bits, whereas in the communication overheads of server S, the cost is 4*163+2*160+160 bits. The results of the comparisons indicate that the proposed scheme imposes significantly lower communication costs than previously proposed schemes.

From **Table 2**, **Fig. 3** and **Fig. 4**, they obviously show that our proposed scheme is more efficient than previously proposed schemes in term of computational complexities and communication overheads.

We also summarize the functionalities of the proposed scheme and make comparison with related researches in **Table 3**. It demonstrates that our scheme can achieve key authentication, key confirmation and user anonymity. The transmission rounds include all independent steps that can be sent and received in parallel. Moreover, our proposed scheme rearranges all independent messages as a round. Our proposed scheme only requires two round-messages, which is less than required by previously proposed schemes.

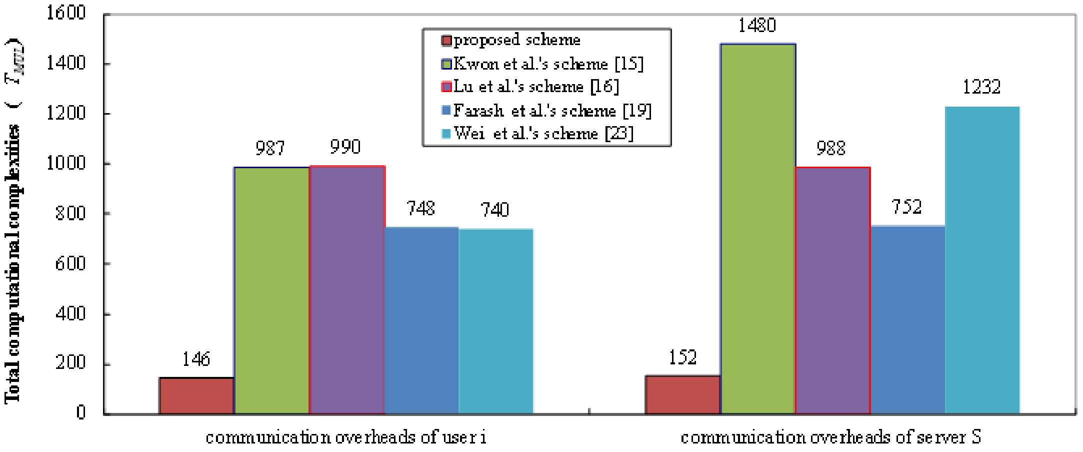

Fig. 3. Comparison of computational costs

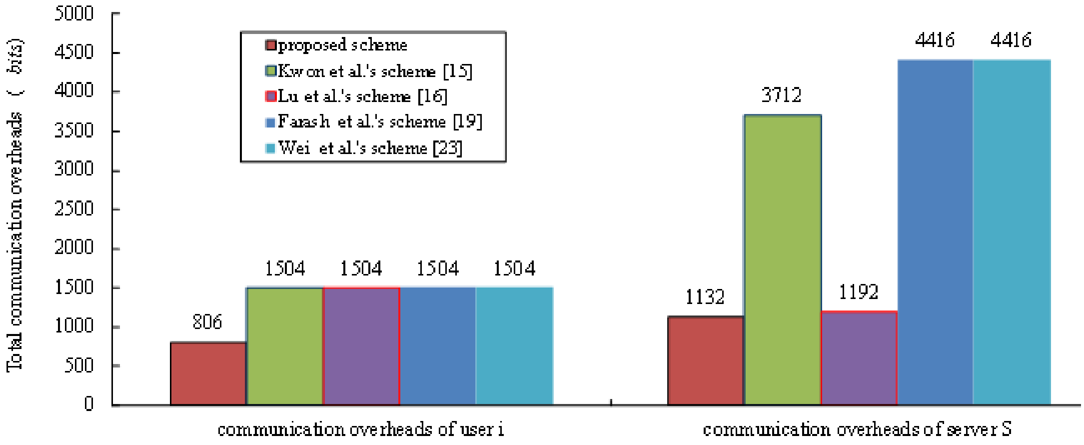

Fig. 4. Size comparison of messages transmitted

Table 3. Comparisons of main functionalities

	Proposed Scheme (*M*-PAKE)	**Kwon *et al.* [15] (3-PAKE)**	**Lu *et al.* [16] (*M*-PAKE)**	**Farash *et al.* [19] (3-PAKE)**	**Wei *et al.* [23] (3-PAKE)**
Round-messages	2	4	2	5	3
User's anonymity	O	X	O	X	X
Key authentication	O	X	O	O	O
Key confirmation	O	X	O	O	O

5. Conclusion

Recently, several researchers have proposed many 3-PAKE protocols. However, we have scrutinized carefully recently published Kwon *et al*.'s protocol, and it has been observed that the same protocol suffers from several security weaknesses such as key authentication, key confirmation and anonymity. To improve the efficiency and solve the security problem of the above 3-PAKE scheme, we proposed a multi-party PAKE scheme with privacy preservation based on the ECC.

The ECC is more commercial importance and has attracted attention because of a smaller key size, reducing storage, low on CPU consumption, and transmission requirements. The proposed scheme is to use ECC which provides striking advantage of shorter key size compared to conventional algorithm (e.g., RSA algorithm), while preserving the equivalent security level. Additionally, the proposed scheme requires only two round-messages and achieves better performance efficiency. Accordingly, the proposed scheme is suitable for applied in mobile environment.

Furthermore, our proposed scheme provides security from entity authentication, confidentiality of private/session key, forward secrecy, user anonymity, key authentication, and key confirmation. The proposed scheme is more efficient than previously proposed schemes and meets security requirements.

The proposed scheme assumes that the server is honest and follows the required security service agreement. However, malicious servers are still possible, and we therefore plan to develop a *M*-PAKE scheme for multi-server mobile networks capable of withstanding malicious attacks even from the servers themselves.

References

[1] S. M. Bellovin and M. Merritt, "Encrypted key exchange: Password-based protocols secure against dictionary attacks," in *Proc. of 1992 IEEE Computer Society Conference on Research in Security and Privacy*, pp. 72-84, May 4-6, 1992. Article (CrossRef Link).

[2] Y. Ding and P. Horster, "Undetectable on-line password guessing attack," *ACM SIGOPS Operating Systems Review*, vol. 29, no. 4, pp. 77-86, October, 1995. Article (CrossRef Link).

[3] H. B. Chen, T. H. Chen, W. B. Lee, and C. C. Chang, "Security enhancement for a three-party encrypted key exchange protocol against undetectable online password guessing attacks," *Computer Standards & Interfaces*, vol. 30, no. 1-2, pp. 95-99, January, 2008. Article (CrossRef Link).

[4] T. F. Lee, T. Hwang, and C. L. Lin, "Enhanced three-party encrypted key exchange without server public keys," *Computers and Security*, vol. 23, no. 7, pp. 571-577, October, 2004. Article (CrossRef Link).

[5] B. W. Simon and M. Alfred, "Authenticated Diffie-Hellman key agreement protocols," in *Proc. of the 5th Annual Workshop on Selected Areas in Cryptography (SAC'98)*, pp. 339-361, August 17-18, 1998. Article (CrossRef Link).

[6] M. Steiner, G. Tsudik and M. Waidner, "Refinement and extension of encrypted key exchange," *ACM SIGOPS Operating Systems Review*, vol. 29, no. 3, pp. 22-30, July, 1995. Article (CrossRef Link).

[7] H. M. Sun, B. C. Chen and T. Hwang, "Secure key agreement protocols for three-party against guessing attacks," *Journal of Systems and Software*, vol. 75, no. 1-2, pp. 63-68, February, 2005. Article (CrossRef Link).

[8] C. L. Lin, H. M. Sun and T. Hwang, “Three-party encrypted key exchange: attacks and a solution,” *ACM SIGOPS Operating Systems Review*, vol. 34, no. 4, pp. 12-20, October, 2000. Article (CrossRef Link).

[9] C. C. Chang and Y. F. Chang, “A novel three-party encrypted key exchange protocol,” *Computer Standards & Interfaces*, vol. 26, no. 5, pp. 471-476, September, 2004. Article (CrossRef Link).

[10] T. H. Chen, W. B. Lee and H. B. Chen, “A round- and computation- efficient three-party authenticated key exchange protocol,” *Journal of Systems and Software*, vol. 81, no. 9, pp. 1581-1590, September, 2008. Article (CrossRef Link).

[11] E. J. Yoon and K. Y. Yoo, “Improving the novel three-party encrypted key exchange protocol,” *Computer Standards & Interfaces*, vol. 30, no. 5, pp. 309-314, July, 2008. Article (CrossRef Link).

[12] N. W. Lo and K. H. Yeh, “Cryptanalysis of two three-party encrypted key exchange protocols,” *Computer Standards & Interfaces*, vol. 31, no. 6, pp. 1167-1174, November, 2009. Article (CrossRef Link).

[13] M. Abdalla, P. A. Fouque and D. Pointcheval, “Password-based Authenticated Key Exchange in the Three-Party Setting,” *Public Key Cryptography - PKC 2005*, pp. 65-84, January 23-26, 2005. Article (CrossRef Link).

[14] M. Abdalla and D. Pointcheval, “Interactive Diffie-Hellman assumptions with applications to password-based authentication,” in *Proc. of 9th International Conference on Financial Cryptography - FC 2005*, pp. 341-356, February 28-March 3, 2005. Article (CrossRef Link).

[15] J. O. Kwon, I. R. Jeong and D. H. Lee, “Practical Password-Authenticated Three-Party Key Exchange,” *KSII Transactions on Internet and Information Systems*, vol. 2, no. 6, pp. 312-332, December, 2008. Article (CrossRef Link).

[16] C. F. Lu, Y. L. Lin and C. L. Hsu, “Password-based Authenticated Multi-party Key Exchange Scheme with Privacy Preservation,” in *Proc. of 2012 International Conference on e-Commerce, e-Administration, e-Society,e-Education, and e-Technology (e-CASE & e-Tech 2012)*, March 30-April 1, 2012.

[17] S. Wu, K. Chen and Y. Zhu, “Enhancements of a three-party password-based authenticated key exchange protocol,” *The International Arab Journal of Information Technology*, vol. 10, no. 3, pp. 215-221, May, 2013. Article (CrossRef Link).

[18] C. L. Hsu and T. W. Lin, “Password authenticated key exchange protocol for multi-server mobile networks based on chebyshev chaotic map,” in *Proc. of 2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pp. 90-95, March 18-22, 2013. Article (CrossRef Link).

[19] M. S. Farash and M. A. Attari, “An enhanced and secure three-party password-based authenticated key exchange protocol without using server's public-keys and symmetric cryptosystems,” *Information Technology And Control*, vol. 43, no. 2, pp. 143-150, June, 2014. Article (CrossRef Link).

[20] Y. Lee, “Cryptanalysis and Improvement of a Password-Based Authenticated Three-Party Key Exchange Protocol,” *International Journal of Security and Its Applications*, vol. 8, no. 4, pp. 151-160, July, 2014. Article (CrossRef Link).

[21] R. Amin and G. P. Biswas, “Cryptanalysis and Design of a Three-Party Authenticated Key Exchange Protocol Using Smart Card,” *Arabian Journal for Science and Engineering*, pp. 1-15, June, 2015. Article (CrossRef Link).

[22] J. Nam, K. -K. R. Choo, S. Han, J. Paik and D. Won, “Two-round password-only authenticated key exchange in the three-party setting,” *Symmetry*, vol. 7, no. 1, pp. 105-124, January, 2015. Article (CrossRef Link).

[23] F. Wei, J. Ma, A. Ge, G. Li and C. Ma, “A Provably Secure Three-Party Password Authenticated Key Exchange Protocol without Using Server's Public-Keys and Symmetric Cryptosystems,” *Information Technology And Control*, vol. 44, no. 2, pp. 195-205, June, 2015. Article (CrossRef Link).

[24] H. T. T. Nguyen, M. Guizani, J. Minho and E. N. Huh, “An Efficient Signal-Range-Based Probabilistic Key Predistribution Scheme in a Wireless Sensor Network,” *IEEE Transactions on Vehicular Technology*, vol. 58, no. 5, pp. 2482-2497, October, 2009. Article (CrossRef Link).

[25] H. T. T. Nguyen, J. Minho, T. D. Nguyen and E. N. Huh, “A beneficial analysis of deployment knowledge for key distribution in wireless sensor networks,” *Security and Communication Networks*, vol. 5, no. 5, pp. 485-495, May, 2012. Article (CrossRef Link).

[26] L. C. Huang and M. S. Hwang, “Two-party authenticated multiple-key agreement based on elliptic curve discrete logarithm problem,” *International Journal of Smart Home*, vol. 7, no. 1, pp. 9-18, January, 2013. Article (CrossRef Link).

[27] A. J. Menezes, P. C. Oorschot and S. A. Vanstone, “Handbook of Applied Cryptography,” *CRC* Press *Inc*., Boca Raton, Florida, 1997. Article (CrossRef Link).

[28] IEEE Std 1363-2000 Working Group, “IEEE Standard Specifications for Public Key Cryptography,” *The Institute of Electrical and Electronics Engineers, Inc.*, New York, August, 2000. Article (CrossRef Link).

[29] A. Menezes, “Elliptic curve public key cryptosystems,” *Kluwer Academic Publishers*, Norwell, Massachusetts, 1993. Article (CrossRef Link).

[30] I. Blake, G. Seroussi and N. Smart, “Elliptic curves in cryptography,” *Cambridge University Press*, Cambridge, United Kingdom, August, 1999. Article (CrossRef Link).

[31] W. Diffie and M. Hellman, “New directions in cryptography,” *IEEE Transactions on Information Theory*, vol.22, no.6, pp.644-654, November, 1976. Article (CrossRef Link).

[32] N. Koblitz, A. Menezes and S. Vanstone, “The state of elliptic curve cryptography,” *Designs, Codes and Cryptography*, vol. 19, no. 2, pp. 173-193, March, 2000. Article (CrossRef Link).

[33] T. S. Chen, E. T. Hsu, and Y. L. Yu, “A New Elliptic Curve Undeniable Signature Scheme,” *International mathematical forum*, vol. 1, no. 31, pp. 1529-1536, 2006. Article (CrossRef Link).

[34] D. Hankerson, J. L. Hernandez and A. Menezes, “Software Implementation of Elliptic Curve Cryptography over Binary Fields,” in *Proc. of Workshop on Cryptographic Hardware and Embedded Systems - CHES 2000*, pp. 1-24, August 17-18, 2000. Article (CrossRef Link).

[35] S. Contini, A. K. Lenstra and R. Steinfeld, “VSH, an Efficient and Provable Collision-Resistant Hash Function,” in *Proc. of 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques: Advances in Cryptology - EUROCRYPT 2006*, pp. 165-182, May 28-June 1, 2006. Article (CrossRef Link).

[36] FIPS PUB 180-4, “Secure Hash Standard (SHS),” *Information Technology Laboratory, National Institute of Standards and Technology (NIST)*, Gaithersburg, Maryland, August, 2015. Article (CrossRef Link).

Chung-Fu Lu received the B.S. and M.S. degree in Electrical Engineering from National Taiwan University of Science and Technology in 1991 and 1993 respectively. He received the Ph.D degree in information management from the National Taiwan University of Science and Technology, Taiwan in 2011. Since August 2011, he has been the Associate Professor in the Department of Information Management, Chihlee University of Technology, Taiwan. His current research includes cryptography, information security, network security, and mobile commerce.