An Enhanced ECC Remote Mutual Authentication with Key Agreement Scheme for Mobile Devices

Ming Luo
School of Software, Nanchang University, Nanchang, P. R. China
Email: lmhappy21@163.com

Donghua Huang and Jun Hu
School of Software, Nanchang University, Nanchang, P. R. China
Email: {eighteenth18, jx_hujun}@163.com

***Abstract*—Recently, Islam and Biswas proposed an efficient and secure ID-based remote mutual authentication with key agreement scheme. We have analyzed the security and performance of Islam and Biswas's scheme and showed that their scheme has some pitfalls. In order to solve these problems, we have constructed an enhanced ECC remote mutual authentication with key agreement scheme and proven that the proposed scheme is a secure authenticated key agreement protocol in the random oracle and can survive against the known session-specific temporary information attack, channel attack and replay attack, and the user can freely choose and change his password without any hassle of contacting the remote server. As compared with Islam and Biswas's scheme, our scheme has better performance in term of the computation cost, security, communication overhead and communication round. Thus, our scheme is suitable for resource-constrained wireless communication networks.**

***Index Terms*—elliptic curve; certificateless cryptography; authentication; key agreement; random oracle**

I. Introduction

To consider security services in wireless communications, mutual authentication and key agreement are very important mechanisms for preventing server impersonation attack, unauthorized network access and malicious attacks of the subsequent session message. The general approach to construct authentication and key agreement schemes is to adopt the Public Key Infrastructure (PKI). However, the PKI approach is costly to use since it involves certificate revocation, distribution, storage and verification. In order to eliminate the above problem, identity-based cryptography (IBC) was introduced. The main benefit of IBC is in greatly eliminating the need for the public key certificates. But the trusted authority called PKG in IBC can generate the private keys of all its users, so private key escrow becomes an inherent problem in IBC. Moreover, private keys must be sent over secure channels, and this makes secret key distribution a daunting task.

To avoid the problems of conventional PKIs and IBC, a new concept called certificateless public key cryptography (CL-PKC) was introduced by Al-Riyami and Paterson [1]. In CL-PKC, a trusted authority called Key Generation Centre (KGC) issues a partial private key for each user, and each user generates the other part of private key, so when the two parts of private keys are known some cryptographic operations can only be performed. Therefore, CL-PKC not only eliminates the use of certificates, but also solves the key escrow problem. Recently, several certificateless key agreement or authentication schemes were proposed [2,3].

Considering the traditional public-key systems require many expensive communication costs and the weak computing capability of mobile devices. In 2006, Das *et al*. [4] proposed an efficient ID-based remote user authentication scheme with smart cards using bilinear pairings. Goriparthi *et al*. [5] showed that their scheme is insecure against forgery attack resulting in an adversary can always pass the authentication. Subsequently, Fang *et al*. [6] proposed an improvement to withstand the mentioned forgery attack. However, Giri and Srivastava [7] pointed out that the Fang *et al.*'s scheme cannot overcome off-line attack and they proposed an improved scheme. Unfortunately it was shown by Tseng *et al*. [8] that the Giri and Srivastava's improvement has too expensive computational cost for smart cards with limited computing capability. In addition, they showed that both [4] and [7] do not provide mutual authentication and key exchange between the user and the server and proposed a solution. In 2009, Goriparthi *et al*. [9] proposed an improved bilinear pairing based remote client authentication protocol based on Das *et al*.'s schme. However, Wu and Tseng [10] showed that schemes [4,6,7,9] do not provide mutual authentication and key exchange between the user and the server, and they proposed a solution using bilinear pairings. As compared with the above client authentication schemes, Wu and Tseng's scheme provides both mutual authentication and key exchange. However, almost all of the above schemes have the bilinear pairings operations, which is not efficient in wireless communications system devices with limited computing capability.

doi:10.4304/jsw.8.12.3206-3213

To avoid the problems of bilinear pairings operations, some Identity-based authentication schemes without bilinear pairings on ECC are proposed [11,12]. However, Yang and Chang [13] point out some of these schemes do not provide the mutual authentication [11] or the key agreement [12] between the user and the server, also they proposed a more efficient ID-based scheme on ECC. Nevertheless, Yoon and Yoo [14] found Yang and Chang's protocol is vulnerable to an impersonation attack and does not provide perfect forward secrecy, and then they proposed an improved scheme which is claimed to offer more security attributes. In 2010, Chen et al. [15] found that the Yang and Chang's scheme is vulnerable to insider attack and impersonation attack, and an improvement was made to remove the above drawback. Recently, Islam and Biswas [16] pointed out that the schemes [13,14,15] suffer from replay attack/clock synchronization problem, known session-specific temporary information attack many logged-in users' attack, inability to protect user's anonymity, does not provide the session key forward secrecy and does not define how to revoke the authentication key with same identity. To resolve such problems, Islam and Biswas proposed a more efficient and secure ID-based remote mutual authentication with key agreement scheme for mobile devices on ECC, also they claimed that they scheme is against the known session-specific temporary information attack.

However, in this paper, firstly, we find that Islam and Biswas's scheme is vulnerable to the known session-specific temporary information attack and has some practical security and performance pitfalls including a secure channel needed between the user and the server, the somewhat inefficient three-way challenge-response handshake technique, the relatively inefficient replay attack detection mechanism, no choice of selecting the users' own password. To overcome the security and performance flaws of Islam and Biswas's scheme, we propose an enhanced ECC remote mutual authentication with key agreement scheme for mobile devices using certificateless public-key cryptography. Compared with Islam and Biswas's scheme, the proposed scheme is more secure, efficient, and practical for mobile devices because the proposed scheme not only eliminates the security flaws of Islam and Biswas's scheme but also reduces the computational costs and communication overheads between the user and the server.

The remainder of this paper is organized as follows. The preliminaries for elliptic curve group and security definitions are given in the next section. Islam and Biswas's scheme is described in Section 3. Section 4 points out the demerits of Islam and Biswas's scheme. The enhanced scheme is presented in Section 5. In Section 6, security and performance analysis of our scheme is presented. Section 7 gives our conclusions.

II. PRELIMINARIES

In this section, the mathematical preliminaries required to understand the following remote mutual authentication with key agreement scheme are introduced. Let the symbol $E_p(a, b)$ be an elliptic curve E over the prime finite field F_p, defined by an equation: $y^2 \bmod p=(x^3+ax+b) \bmod p$ with $a, b \in F_p$ and $4a^3+27b^2 \neq 0 \bmod p$. The points on $E_p(a, b)$ together with an point O known as "point at infinity" form a additive elliptic curve group defined as $G_p=\{(x, y) : x, y \in F_p, E(x, y)= 0\} \bigcup \{O\}$.

G_p is a cyclic group under the point addition defined as follows: Let $X, Y \in G_p$, l is the line containing X and Y (tangent line to $E_p(a, b)$ if $X=Y$), and Z be the third point of intersection of l with $E_p(a, b)$. Let l' be the line connecting Z and O. Then $X+Y$ is the point such that l' intersects $E_p(a, b)$ at Z and O. Given a point $P \in E_p(a, b)$ and an integer $s \in F_p$, the scalar multiplication over $E_p(a, b)$ can be defined as follows: $sP = P + P + \cdots + P$(s times). A point P has order n if $n \cdot P=O$ for smallest integer $n>0$.

The following computational problems defined over G_p are assumed to be intractable within polynomial time and those are frequently used to construct secure cryptographic schemes. So far, the probability of any polynomial-time algorithm to solve the following computational problems is negligible.

Elliptic Curve Discrete Logarithm Problem (ECDLP): Given two elements P, $Q \in G_p$, the ECDLP in G_p is to compute $x \in [1, n-1]$ given $(P , Q=xP)$

Computational Diffie–Hellman problem (CDHP): Given a generator P of G_p and (aP, bP) for unknown $a, b \in [1, n-1]$, the task of CDHP is to compute abP.

III. REVIEW OF ISLAM AND BISWAS'S SCHEME

In this section, we briefly review Islam and Biswas's scheme.

A. System Initialization Phase

1) The server S chooses a k-bit prime number p and a base point P with the order n over $E_p(a, b)$.

2) The server S chooses a random number q_S (the server's master secret) from $[1, n-1]$ and computes his public key $Q_S=q_S \cdot P$.

3) The server S selects two one-way hash functions $H_1:\{0, 1\}^* \times Z_p^* \rightarrow G_p$, $H_2:G_p \times G_p \rightarrow Z_p^*$ and a secure key derivation function kdf: $\{0, 1\}^* \times G_p \times G_p \rightarrow \{0, 1\}^k$.

4) The server S publishes the system parameters $\{E_p(a, b), P, Q_S, H_1, H_2, kdf\}$ and keeps his master key q_S in private.

B. User Registration Phase

1) The user U selects his identity ID_U and sends it to the remote server S with some personal authentication information through a secure channel.

2) The server S verifies the user U's identity ID_U. If the value ID_U already exists in the database of the server, S asks the user U for a new identity. Thereafter details of registration message will be checked by the remote server S and computes the authentication key $AID_U = q_S \cdot H_1(ID_U||X)$, where $X \in Z_p^*$ is randomly chosen by the server S. The remote server S stores the registration information $<ID_U, X, status\text{-}bit>$ about the user U to a

secure database. The remote server S sets the *status-bit* to "1" if the user is already logged in, otherwise sets to "0".

3) The server S sends the authentication key AID_U to the user U through a secure channel.

C. Mutual Authentication With Session Key Agreement Phase

1) Initially, the user U enters his identity ID_U and authentication key AID_U and chooses a random number r_U from [1, n-1], then computes $N=R+AID_U$, $M=r_U \cdot Q_S$ where $R= r_U \cdot P$ and computes the dynamic identity information $CID_U=ID_U \oplus H_2(R||AID_U)$ and submits the authentication message $<CID_U, N, M>$ to the remote server S.

2) Upon receiving the authentication message $<CID_U, N, M>$, the remote server S computes $R^* = q_s^{-1} \cdot M$ and $AID_U=N-R^*$. Then, S computes the user's identity $ID_U=CID_U \oplus H_2(R^* ||AID_U)$ and verifies the identity ID_U. If the value ID_U is invalid, the server S rejects U's login request, otherwise performs the next step.

3) Then, the remote server S computes $AID_U^* =q_S \cdot H_1(ID_U||X)$ and checks the equation $AID_U^* =?AID_U$, where ID_U and X are obtained from server's database. If it does not hold the remote server S rejects the user's login request, otherwise randomly selects a number r_S from [1, n-1], and computes $T=R^* +S$ ($S=r_S \cdot P$) and $H_S=H_2(S|| AID_U^*)$. Next, the server S sends the authentication message $<T, H_S>$ to the user.

4) When receiving the authentication message $<T, H_S>$, the user U computes $S^* =T-R$ and $H_S^* =H_2(S^* ||AID_U)$, then checks the equation $H_S^* =?H_S$. If it holds the user U sends the authentication message $<H_{RS}>$, where $H_{RS}=H_2(R||S^*)$. The user U obtains the session key by computing $SK=kdf(ID_U||AID_U||K)$, where $K=r_U \cdot S=r_U \cdot r_S \cdot P$.

5) When receiving the authentication message $<H_{RS}>$, the remote server S computes $H_{RS}^* =H(R^* ||S)$ and checks the equation $H_{RS}^* =?H_{RS}$. If it holds the server S obtains the session key by computing $SK=kdf\ (ID_U||AID_U||K)$, where $K= r_S \cdot R =r_S \cdot r_U \cdot P$.

D. Leaked Key Revocation Phase

Assume that an adversary illegally obtains the user U's authentication key AID_U, so the user U should send a request message to the server S for a new authentication key. The user U sends the identity ID_U, the old authentication key AID_U and some personal authentication information to the remote server S. Next, the server S first verifies the identity ID_U. After successfully validating user's credential, the remote server S randomly chooses another number $\overline{X} \in Z_p^*$ and computes the new authentication key $\overline{AID_U} =q_S \cdot H_1(ID_U|| \overline{X})$ with the same identity ID_U. It is to be pointed out that the authentication key revocation does not need a new identity of the user U, only the value X will be modified in each revocation phase. The server S sends the fresh authentication key $\overline{AID_U}$ through a secure channel to the user U. The remote server stores the database same identity ID_U except that X is exchanged by $\overline{X}$.

IV. Demerits of Islam and Biswas's Scheme

In this section, the security and performance of Islam and Biswas's scheme are analyzed carefully. Islam and Biswas state that the known session-specific temporary information (KSSTI) attack is infeasible in their scheme. However, in this section, we show that their scheme is insecure against the KSSTI attack. Also we find some security and performance pitfalls including a secure channel needed between the user and the server, the somewhat inefficient three-way challenge-response handshake technique, the relatively inefficient replay attack detection mechanism, no choice of selecting the users' own password.

A. Known Session-Specific Temporary Information Attack

Known session-specific temporary information attack means if the session short-lived secrets are leaked but from this disclosure, secrecy of generated session key should not be compromised. In the Islam and Biswas's scheme, the server and the user compute the same session key $SK=kdf(ID_U||AID_U||K)$, where $K=r_U \cdot S= r_S \cdot R=r_U \cdot r_S \cdot P$. Now if the session ephemeral secrets r_U and r_S are disclosed to an adversary by some means then he can compute the session key SK through the followings steps.

1) In the mutual authentication with key session agreement phase, we assume that the adversary, eavesdropping on the channel, has obtained all the messages exchanged in that session phase including $<CID_U, N=R+AID_U, M>$ and $<T=R^* +S, H_S>$, where $R= r_U \cdot P$ and $S=r_S \cdot P$.

2) The adversary can computes $R= r_U \cdot P$ and $AID_U=N-R$ with knowing the session ephemeral secrets r_U and N, or he can computes $S=r_S \cdot P$, $R =T-S$ and $AID_U=N-R$ with knowing the session ephemeral secrets r_S, T and N.

3) The adversary can computes $K=r_U \cdot r_S \cdot P$ and the session key $SK=kdf(ID_U||AID_U||K)$.

Hence, we conclude that the Islam and Biswas's scheme does prevent the known session-specific temporary information attack.

B. Secure Channel Needed Between The User And The Server

In the user registration phase of the Islam and Biswas's scheme, the user U chooses his identity ID_U and submits it to the server S with some personal secret information through a secure channel, and the server S returns the authentication key AID_U to U through secure channel. In the leaked key revocation phase of the Islam and Biswas's scheme, the server S returns the new authentication key $\overline{AID_U}$ to the user U through a secure channel. Thus, the authentication key escrow and distribution become a daunting task in they scheme. Moreover, using this secure channel will be facing some

security risks, such as if this secure channel is broken, the adversary can obtain the user's identity ID_U, authentication key AID_U and some personal secret information, which makes the Islam and Biswas's scheme does not preserve the anonymity of the user and the security of the user's authentication key.

C. Three-Way Challenge-Response Handshake Technique

The Islam and Biswas's scheme follows the three-way challenge-response handshake technique to provide the mutual authentication with session key agreement. This technique is not efficient, since the two-way challenge-response technique can be used to achieve this security target. Moreover, in their proposed scheme, the user authenticates the server first then the server authenticates the user, which is not a case in real-life applications. Usually, the server authenticates the user first then the user authenticates the server.

D. Inefficient Replay Attack Detection Mechanism

Assume that an adversary may replay the old message $<CID_U, N, M>$ in the step 1 of mutual authentication phase to impersonate a legal user U for the Islam and Biswas's scheme. However, only when the mutual authentication phase is performed to the fifth step, this attack can be detected. That is to say, the unwanted computational costs of four steps need to be performed in mutual authentication phase before the replay attack is detected.

E. No Choice Of Selecting The Users' Own Password

In Islam and Biswas's scheme, the user's authentication key is entirely generated by the remote server S and the user has no choice of selecting his own password, this situation is not a sound case in real-life applications, e.g. digital library, M-commerce, online banking, etc. Secondly, the user's authentication key AID_U chosen by the remote server could be random and long (for example, 512 or 1024 bits), which could be difficult for a user to key correctly this value into the mobile device and remember these numbers easily. Thirdly, when a user wants to change its secret key, he can only submit the key change request to the remote server S through a secure channel, which brings some inconveniences to the user and server.

V. Proposed Scheme

In this section, we propose an enhanced ECC remote mutual authentication with key agreement scheme for mobile devices using certificateless public-key cryptography to overcome the weaknesses of Islam and Biswas's scheme.

A. System Initialization Phase

1) The server S chooses a k-bit prime number p and a base point P with the order n over $E_p(a, b)$.

2) The server S chooses a random number q_S as its own private key from [1, n-1] and computes the corresponding public key $Q_S=q_S \cdot P$.

3) The server S selects four one-way secure hash functions $H_1: G_p \times G_p \rightarrow Z_p^*$, H_2: $\{0, 1\}^* \times \{0, 1\}^* \rightarrow G_p$, H_3: $\{0, 1\}^* \times G_p \times G_p \rightarrow Z_p^*$, H_4: $\{0, 1\}^* \times G_p \times G_p \times G_p \rightarrow Z_p^*$, a one-way secure key derivation function: kdf : $(\{0, 1\}^*)^3 \times (G_p)^4 \rightarrow \{0, 1\}^k$ and a one-way secure key confirmation function: $MAC_k(m)$: the secure message authentication code of m under the key k.

4) The server S publishes the system parameters $\{E_p(a, b), P, Q_S, H_1, H_2, H_3, H_4, kdf, MAC_k(m)\}$ and keeps his master key q_S in private.

B. Password Generation Phase

This phase is executed by the user U with the system parameters. The user U selects a random number s_U as his password from [1, n-1] and computes the corresponding public information $PK_U= s_U \cdot P$.

C. User Registration Phase

In order to avoid the authentication key escrow and distribution problem, and preserve the anonymity of the user and the security of the user's authentication key in an open channel, we design a more secure user registration phase by introducing some modifications to the Islam and Biswas's scheme. In the following, we explain the user registration phase in four steps.

1) The user U first selects his identity $ID_U=\{0, 1\}^p$, computes $K_1=s_U \cdot Q_S$ and $RG_U=ID_U \oplus H_1(K_1, PK_U)$, then submits the register information $<RG_U, PK_U>$ to the server S through an open channel.

2) The server S computes $K_1= q_S \cdot PK_U$ and $ID_U=RG_U \oplus H_1(K_1, PK_U)$, verifies the user U's identity ID_U. If the value ID_U already exists in the database of the server, S asks the user U for a new identity. Thereafter details of registration message will be checked by the remote server S and computes the user U's public key $Q_U=H_2(ID_U, X)$ and private key $D_U=q_S \cdot Q_U$, where $X \in Z_p^*$ is randomly chosen by the server S. The remote server S stores the registration information $<ID_U, X, status\text{-}bit>$ about the user U to a secure database. The remote server S sets the *status-bit* to "1" if the user is already logged in, otherwise sets to "0".

3) The server S computes $RQ_U=(X||Q_U||D_U) \oplus H_1(K_1, PK_U)$ and returns it to U through an open channel.

4) The user U computes $(X||Q_U||D_U)=RQ_U \oplus H_1(K_1, PK_U)$, and obtains his public/private key pair (Q_U, D_U).

D. Mutual Authentication With Session Key Agreement Phase

In order to achieve the mutual authentication with key session agreement, the Islam and Biswas's follows the three-way challenge-response handshake technique, but the proposed scheme follows the more efficient two-way challenge-response handshake technique. Assume that the message communication is over an open channel in this phase. Initially, the user U enters his identity ID_U and the password s_U into the mobile device, the device computes $Q'=H_2(ID_U, X)$ and $PK'=s_UP$, and then checks if $Q'=Q_U$ and $PK'=PK_U$. If they are incorrect, terminates the operation, otherwise, the mutual authentication with session key agreement phase is performed as follows.

1) The device acquires the current time stamp T_U, then randomly selects a number x from [1, n-1], and computes $R=(x+1)Q_U$, $S=xD_U$, $N=S+s_U \cdot Q_S=S+K_1$, $CID_U=(ID_U||N) \oplus H_3(T_U, R, D_U)$, $k=H_4(T_U, R, D_U, K_1)$ and sends the authentication message $<CID_U, T_U, R, S, MAC_k(CID_U, T_U, R, S)>$ to the server S.

2) As receives the authentication message $<CID_U, T_U, R, S, MAC_k(CID_U, T_U, R, S)>$ at time T_1, the server S first verifies the time interval between T_U and T_1. If $(T_1-T_U) \leqq \triangle t$, S continues to verify the authentication message. Otherwise, the authentication message is rejected. Here $\triangle t$ denotes the expected valid time interval for transmission delay. Then, S computes $D_U=q_S \cdot R-S$, $(ID_U||N)=CID_U \oplus H_3(T_U, R, D_U)$, $K_1=N-S$ and $k=H_4(T_U, R, D_U, K_1)$. Then, S checks the integrity of $MAC_k(CID_U, T_U, R, S)$ with the key k. S will quit the current session if the check produces a negative result. Otherwise, S chooses a random number y from [1, n-1], and computes $T=(y+1)Q_U$, $K_2=(y+1)R=(x+1)(y+1)Q_U$ and the session key $MK=kdf(ID_U, T_U, T_S, R, T, K_1, K_2)$, where T_S is a timestamp denoting the current time. Then S sends the authentication message $<T_S, T, MAC_k(ID_U, T_S, T)>$ to the user U.

3) Upon receiving the authentication message $<T_S, T, MAC_k(ID_U, T_S, T)>$ at time T_2. The user U verifies the validity of the time interval between T_S and T_2 for transmission delay. If T_S is valid, the user authenticates the service server S by checking the integrity of $MAC_k(ID_U, T_S, T)$ with the key k. U will quit the current session if the check produces a negative result. Otherwise, the user U computes $K_2=(x+1)T=(x+1)(y+1)Q_U$ and the session key $MK=kdf(ID_U, T_U, T_S, R, T, K_1, K_2)$.

E. Password Change Phase

The password change phase does not need any interaction with the remote server. This phase can be invoked whenever the user U wants to perform this operation and works as following steps:

1) The user U enters his identity ID_U and the password s_U into the mobile device. The device device computes $Q'=H_2(ID_U, X)$ and $PK'=s_UP$, and then checks if $Q'=Q_U$ and $PK'=PK_U$. If they are incorrect, it terminates the operation, otherwise, continues next step,.

2) The mobile device allows user U to submits a fresh password $s_U^{'}$, then the device computes $PK_U^{'} = s_U^{'}P$ and $K_1= s_U^{'}Q_S$. Finally, the device stores new key information $s_{U'}$, $PK_U^{'}$ and K_1.

F. Leaked Key Revocation Phase

Assume that an adversary illegally obtains the user U's private key D_U, so the user U should send a request message to the server S for a new private key. The user U computes $RG_U=(ID_U||D_U) \oplus H_1(K_1, PK_U)$ and submits the register information $<RG_U, PK_U>$ to the server S through an open channel. After receiving the register information, the server S computes $(ID_U||D_U)=RG_U \oplus H_1(K_1, PK_U)$ and verifies the identity ID_U. After successfully validating user's credential, the remote server S randomly chooses another number $\overline{X} \in Z_p^*$ and computes the new public key $\overline{Q_U} =H_1(ID_U, \overline{X})$ and private key $\overline{D_U} =q_S \cdot \overline{Q_U}$ with the same identity ID_U. It is to be pointed out that the public/private key pair (Q_U, D_U) revocation does not need a new identity of the user U, only the value X will be modified in each revocation phase. The server S computes $\overline{RQ_U} =(\overline{X} \| \overline{Q_U} \| \overline{D_U}) \oplus H_1(K_1, PK_U)$ and returns it to U through an open channel. The remote server stores the database same identity ID_U except that X is exchanged by $\overline{X}$.

VI. Security Analysis

A. Security Analysis

In the security model of certificateless public key cryptography defined by Al-Riyami and Paterson, there are two kinds of adversaries:

Type I Adversary: A_1 cannot obtain the master private key of server but can replace the public information PK_U of any entity with a value chosen by himself.

Type II Adversary: A_2 can obtain the master private key of server but cannot perform public information replacement.

Here, we show that the proposed scheme is a secure authenticated key agreement protocol under the random oracle model in Theorem 1.

Theorem 1. The proposed scheme is a secure authenticated key agreement protocol under the random oracle model, Specifically, suppose the adversary $A_{i(i=1,2)}$ against the scheme with non-negligible probability $Adv(A_i)$ and in the attack kdf has been queried q_h times at most and q_n oracles have been created. Then there exists a challenger C solve the CDH problem with an advantage $2Adv(A_i) / (q_h \cdot q_n \cdot (q_n - 1))$.

Proof. Given an instance of the Computational Diffie-Hellman problem (P, aP, bP). The goal of challenger C is to compute abP. At the beginning of the game, the challenger C sends the system parameters $\{E_p(a, b), P, Q_S, H_1, H_2, H_3, H_4, kdf, MAC_k(m)\}$ to A_i and sends the server's master secret q_S to A_2.

The challenger C chooses two random integers τ, υ (assuming $\tau < \upsilon$) from $\{1,2, ..., q_n\}$ and works by interacting with A_i as follows:

Create-User: C maintains a list L_u of tuples (ID_U, s_U, PK_U). On a new Create-User query for user U, C selects a random number $s_U \in Z_q^*$ as U's password and computes the corresponding public information $PK_U= s_UP$. Then, C adds (ID_U, s_U, PK_U) into the list L_u and returns PK_U to A_i.

Password-Extract queries: On a Password-Extract query of ID_U, We assume that Create-User query for ID_U has been asked. C searches a pair (ID_U, s_U, PK_U) corresponding to ID_U in the list L_u, then return s_U to A_i .

Public-Information-Replace queries: For the Type I adversary, A_1 can request to replace public information PK_U of a user U with new public information $PK_U^{'}$ chosen by A_1 itself. C replaces the original public information PK_U with $PK_U^{'}$ if ID_U has been created. Otherwise, C executes Create-User query to generate (ID_U, s_U, PK_U), then sets $PK_U= PK_U^{'}$ and adds $(ID_U, s_U, PK_U^{'})$ to the L_u.

Here, to replace public information, the password value corresponding to the new public information is not required. For the Type II adversary, he cannot perform this query**.**

Private-Key-Extract queries: On a Private-Key-Extract query for ID_U, We assume that H_2 query for ID_U has been asked. C searches an element (ID_U, X, w) corresponding to ID_U in the list L_2, then computes $D_U = wQ_S$ and returns D_U as the answer. For the Type II adversary, he doesn't need to perform this query.

***H_1* queries:** C maintains a list L_1 of tuples (K_1, PK_U, h_1).On a H_1 query of (K_1, PK_U), C searches an element (K_1, PK_U, h_1) in the list L_1. If such an element is found, C answers h_1, otherwise, C chooses a random number $h_1 \in Z_p^*$, then C will puts the element (K_1, PK_U, h_1) in list L_1 and answers h_1.

***H_2* queries:** C maintains a list L_2 of tuples (ID_U, X, w). Upon receiving a $H_2(ID_U, X)$ query, C first searches L_2 for the tuple with (ID_U, X, w). If the requested input is already on the list, then the corresponding $Q_U = wP$ is returned, otherwise C chooses a random number $w \in Z_p^*$ and sets $Q_U=wP$, then C will puts the tuple (ID_U, X, w) in list L_2 and answers Q_U.

***H_3* queries:** C maintains a list L_3 of tuples (T_U, R, D_U, h_3). Upon receiving a $H_3(T_U, R, D_U)$ query, C checks if there exists (T_U, R, D_U, h_3) in L_3. If such an element is found, C answers h_3, otherwise he answers A_i with a random binary sequence $h_3 \in Z_p^*$ and puts the (T_U, R, D_U, h_3) into L_3.

***H_4* queries:** C maintains a list L_4 of tuples (T_U, R, D_U, K_1, h_4). Upon receiving a $H_4(T_U, R, D_U, K_1)$ query, C checks if there exists (T_U, R, D_U, K_1, h_4) in L_4. If such an element is found, C answers h_4, otherwise he answers A_i with a random binary sequence $h_4 \in Z_p^*$ and puts the (T_U, R, D_U, K_1, h_4) into L_4.

***kdf* queries:** C maintains a list L_k of tuples $(ID_U, T_U, T_S, R, T, K_1, K_2, kdf)$. Upon receiving a $kdf(ID_U, T_U, T_S, R, T, K_1, K_2)$ query, C checks if there exists $(ID_U, T_U, T_S, R, T, K_1, K_2, kdf)$ in L_k. If such an element is found, C answers kdf, otherwise he answers A_i with a random binary sequence $kdf \in Z_p^*$ and puts the $(ID_U, T_U, T_S, R, T, K_1, K_2, kdf)$ into L_k.

Send queries: On a Send query, we have three cases to consider as follows.

Case 1: On a Send($\Pi_{U,S}^n$,M) query (M is an empty message), C performs the *step* 1 of mutual authentication with key session agreement algorithm and responses with the authentication message $M_0 = <CID_U, T_U, R, S, MAC_k(CID_U, T_U, R, S)>$. At the τ -th Send($\Pi_{U,S}^\tau$, M) query, C lets $R=aP$ and $S=q_S \cdot (R\text{-}Q_U)$ (For the Type I adversary's query, C can not compute $S=q_S \cdot (R\text{-}Q_U)$, he just consider $S=q_S \cdot (R\text{-}Q_U)$), then C obtains the password s_U corresponding to ID_i by running the Password-Extract query algorithm, computes $N=S+s_UQ_S$, $CID_U=(ID_U||N) \oplus H_3(T_U, R, D_U)$ and $k=H_4(T_U, R, D_U, s_UQ_S)$, finally C responses with the authentication message $M_1=<CID_U, T_U, R, S, MAC_k(CID_U, T_U, R, S)>$.

Case 2: On a Send($\Pi_{S,U}^n$,M_0) query, C performs the *step* 2 of mutual authentication with key session agreement algorithm and responses with the authentication message $M_2=<T_S, T, MAC_k(ID_U, T_S, T)>$. At the υ -th Send($\Pi_{S,U}^\upsilon$, M) query, if $M \neq M_1$, C stops and fails **(Event 1)**; Else, C checks the freshness T_U. Then, C obtains $S=q_S \cdot (R\text{-}Q_U)$ from Case 1, computes $(ID_U||N)=CID_U \oplus H_3(T_U, R, D_U)$, $K_1=N\text{-}S$, $k=H_4(T_U, R, D_U, K_1)$ and checks the integrity of $MAC_k(CID_U, T_U, R, S)$ with the key k. C stops if the check result is false. Otherwise, C lets $T=bP$ and $K_2=\delta$ (where δ is C candidate for the CDH problem), computes the session key $\eth=kdf(ID_U, T_U, T_S, R, T, K_1, \delta)$ and puts the $(ID_U, T_U, T_S, R, T, K_1, \delta, \eth)$ into L_k. Finally C responses with the authentication message $M_3=<T_S, T, MAC_k(ID_U, T_S, T)>$.

Case 3: When receiving Send($\Pi_{U,S}^n$,M_2) query, C performs the *step* 3 of mutual authentication with key session agreement algorithm and responses with the session key MK. When receiving Send($\Pi_{U,S}^n$,M_3) query, C checks the freshness T_S. Then, C checks the integrity of $MAC_k(ID_U, T_S, T)$ with the key k. C stops if the check result is false. Otherwise, C responses with the session key $\eth=kdf(ID_U, T_U, T_S, R, T, K_1, \delta)$.

Reveal queries: On a Reveal query, C responses with the appropriate session key, except if A_i asks the oracle $\Pi_{U,S}^\tau$ or $\Pi_{S,U}^\upsilon$ to ask the Test query, then C stops and fails (**Event 2**).

Test queries: On a Test query, If A_i does not select the guessed oracle $\Pi_{U,S}^\tau$ or $\Pi_{S,U}^\upsilon$ to ask the Test query, then C stops and fails (**Event 3**); otherwise, C chooses a random value $\beta=kdf$ from L_k and responds with β to A_i.

Output: Finally, A_i outputs its guess.

Solving the CDHP: C chooses a tuple $(ID_U, T_U, T_S, R, T, K_1, \delta, \eth)$ from L_k and returns δ as the answer to the CDHP challenge.

Now we estimate the probability that C does not fail, namely Event 1, 2 and 3 do not happen. As can be seen from the above game, if the test session is between the τ -th and υ -th oracle, then the game goes through. The probability that the game has chosen the right session is $1/(q_n \cdot (q_n - 1))$, since a randomly chosen oracle is $1/q_n$ and the other randomly chosen oracle in remaining oracles is $1/(q_n\text{-}1)$. We have: *Adv(C does not fail)* $> 1/(q_n \cdot (q_n - 1))$

Let $\hat{H}$ be the event that $\delta = abP$ has been queried to *kdf*. Because *kdf* is a random oracle, we have $\Pr[A_i \text{ wins}|\neg\hat{H}] = 1/2$. Then

$\Pr[A_i \text{ wins}]$

$= \Pr[A_i \text{ wins}|\neg\hat{H}]\Pr[\neg\hat{H}]+\Pr[A_i \text{ wins}|\hat{H}]\Pr[\hat{H}]$

$\leq \Pr[A_i \text{ wins}|\neg\hat{H}]\Pr[\neg\hat{H}]+\Pr[\hat{H}]$

$=1/2(\Pr[\neg\hat{H}])+\Pr[\hat{H}]=1/2+1/2(\Pr[\hat{H}])$

It follows that $\Pr[\hat{H}] \geq 2Adv(A_i)$.Combining all the above results, we have that C solves the CDHP with probability at least $2Adv(A_i)/(q_h \cdot q_n \cdot (q_n - 1))$.

B. Other Security Properties

Next, we will heuristically argue that enhanced ECC remote mutual authentication with key agreement scheme for mobile devices using certificateless public-key cryptography satisfies the following security properties.

1) **Known session-specific temporary information attack:** Compromising the ephemeral private keys of a session does not enable an attacker to compute the session key. Specifically, obtaining the keys x and y in any session between user ID_U and server S, allows the attacker to compute $K_2=xT=yR=xyQ_U$. However, in order to compute $MK=kdf(ID_U, T_U, T_S, R, T, K_1, K_2)$, the attacker needs to obtain $K_1= s_U \cdot Q_S=q_S \cdot PK_U$, hence he must know at least one long-term private key s_U or q_S. Given $PK_U=s_UP$ or $Q_S=q_SP$, it is hard to compute s_U or q_S under the assumption of DLP. Therefore, known session-specific temporary information attack is infeasible in our proposed scheme.

2) **Channel attack:** In our scheme, user and server do not need a secure channel to transmit the exchange messages in the user registration phase, password change phase and leaked key revocation phase. Moreover, our scheme eliminates the key escrow and distribution problem, also the anonymity of the user and the security of the user's authentication key can be achieved in the open channel. Therefore, the proposed scheme can survive against the channel attack.

3) **Replay attack:** Our scheme can withstand replay attack and this type of attack can be detected in the first step of the mutual authentication with key agreement phase between the user and server because the authenticity of two authentication messages $MAC_k(CID_U, T_U, R, S)$ and $MAC_k(ID_U, T_S, T)$ is firstly verified by checking the freshness of time stamps T_U and T_S, respectively.

4) **Secure password change:** In the presented scheme, the mobile device holder can freely choose and change his password without any hassle of contacting the remote server S. Also, the password chosen by user is easier to remember than the authentication key chosen by server. Any other person, even having stolen or get the mobile device cannot change or update the password without knowing the corresponding valid ID_U and s_U of the mobile device.

5) **Mutual authentication:** Suppose that an attacker wants to deceive server S into thinking he is the user ID_U, he needs to know $k=H_4(T_U, R, D_U, K_1)$ to compute $MAC=MAC_k(CID_U, T_U, R, S)$, but he can't compute the $K_1=s_U \cdot Q_S=q_S \cdot PK_U$ without user's password s_U or server's private key q_S under the assumption of CDHP, thus none other than server S and user ID_U can compute the value MAC and thus server S can authenticate user ID_U by verifying the value MAC, Similarly, user ID_U can authenticate server S in the same way. So, our scheme achieve mutual authentication between user and server with the two-way challenge-response handshake technique, where the server authenticates the user first then the user authenticates the server.

C. Performance Analysis

In this section, we compare the efficiency of our scheme with Islam and Biswas's scheme in terms of computation cost (not including precomputation cost), security, communication overhead and communication round in Table 1. We use the following notations to analyze the efficiency.

- PM, PA is the computation cost for point multiplication and point addition/subtraction respectively.
- KSSTIA, CAK, SPC is the abbreviation for known session-specific temporary information attack, channel attack and secure password change respectively.
- Y and N denote that whether satisfy this security property.

As shown in the Table 1, Islam and Biswas's scheme cannot survive against the known session-specific temporary information attack and channel attack, and cannot achieve secure password change. Our scheme eliminates these security shortcomings. Although Islam and Biswas's scheme has less computation cost in the registration phase, their scheme needs a secure channel to transmit the exchange messages and does not consider the anonymity of the user in this phase, and they scheme requires one additional point multiplication computation than ours in terms of the total computation cost. Compared with their scheme, our scheme not only enjoys less computation cost, communication overhead and communication round, but also has higher security level. Hence, consider the communication security and mobile devices with limited computing capability it may be that our enhanced ECC remote mutual authentication with key agreement scheme is more applicable.

Ⅶ. CONCLUSIONS

In this paper, we have analyzed the security and performance of Islam and Biswas's scheme and showed that their scheme has some security and performance pitfalls. In order to solve these problems, we have constructed an enhanced ECC remote mutual authentication with key agreement scheme for mobile devices using certificateless public-key cryptography and proven that the proposed scheme is a secure authenticated key agreement protocol in the random oracle and can

TABLE I.
COMPUTATION, COMMUNICATION AND SECURITY COMPARISON

Scheme	Computation cost		Communication		Security		
	Registration phase	Authentication phase	overhead	round	KSSTIA	CAK	SPC
Islam's scheme [16]	1PM	7PM+4PA	3P+2H	3	N	N	N
Our scheme	2PM	6PM+3PA	2P+2H	2	Y	Y	Y

survive against the known session-specific temporary information attack, channel attack and replay attack, also satisfies mutual authentication and the mobile device holder can freely choose and change his password without any hassle of contacting the remote server in our scheme. By exploiting the certificateless public key cryptography system, our scheme successfully eliminates the key escrow issue which is inherent in identity-based cryptography. As compared with Islam and Biswas's scheme, our scheme has better performance in term of the computation cost, security, communication overhead and communication round. Thus, our scheme is more suitable for resource-constrained wireless communication networks.

ACKNOWLEDGMENT

The authors would like to thank the reviewers for giving valuable suggestions and comments. This work is supported by the National Natural Science Foundation of China under grant no. 11226042, the Science and Technology Program of Jiangxi Province under grant no. 20132BAB211028 and 20132BBE50042, the Science and Technology Project of Jiangxi Provincial Department of Education under grant no. GJJ13084.

REFERENCES

[1] S. S. Al-Riyami, K. G. Paterson, "Certificateless Public Key Cryptography", in: *Proceedings of Cryptography-Asiacrypt 2003*, Taipei, Taiwan, pp. 452–473, 2003.

[2] M. Chen, K.G. Wu, J. Xu, et al, "A Certificateless and Across Administrative Domains Authenticated Key Exchange Scheme for E-payment", *Journal of Software*, vol. 6, no. 10, pp. 1985-1992, 2011.

[3] W. Gao, G. L. Wang, X. L. Wang, et al, "Controllable Ring Signatures and Its Application to E-Prosecution", *Journal of Computers*, vol. 8, no. 4, pp. 833-841, 2013.

[4] M. L. Das, A. Saxena, V. P. Gulati, et al, "A novel remote user authentication scheme using bilinear pairings", *Computers & Security*, vol. 25, no. 3, pp.184–189, 2006.

[5] T. Goriparthi, M. L. Das, A. Negi, et al, "Cryptanalysis of recently proposed Remote User Authentication Schemes", in: *Cryptology ePrint Archive*, Report 2006/028, pp. 1-6, 2006.

[6] G. Fang, G. Huang, "Improvement of recently proposed remote client authentication protocols", in: *Cryptology ePrint Archive*, Report 2006/200, pp. 1-6, 2006.

[7] D. Giri, P. D. Srivastava, "An improved remote user authentication scheme with smart cards using bilinear pairings", in: *Cryptology ePrint Archive*, Report 2006/274, pp. 1-11, 2006.

[8] Y. M. Tseng, T. Y. Wu, J. D. Wu, "A mutual authentication and key exchange scheme from bilinear pairings for low power computing devices", in: *Proceedings of COMPSAC 2007*, Beijing, China, pp. 700-710, 2007.

[9] T. Goriparthi, M. L. Das, A. Saxena, "An improved bilinear pairing based remote user authentication scheme", *Computer Standards & Interfaces*, vol.31, no.1, pp.181–185, 2009.

[10] T. Y. Wu, Y. M. Tseng, "An efficient user authentication and key exchange protocol for mobile client–server environment", *Computer Networks*, vol.54, no.9, pp. 1520–1530, 2010.

[11] P. E. Abichar, A. Mhamed, B. Elhassan, "A fast and secure elliptic curve based authenticated key agreement protocol for low power mobile communications", in: *Proceedings of NGMAST 2007*, Athens, Greece, pp. 235–240, 2007.

[12] X. Cao, W. Kou, Y. Yu, et al, "Identity-based authentication key agreement protocols without bilinear pairings", *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E91.A, no. 12, pp. 3833-3836, 2008.

[13] J. Yang, C. Chang, "An ID-based remote mutual authentication with key agreement protocol for mobile devices on elliptic curve cryptosystem", *Computers & Security*, vol.28, no.3, pp.138–143, 2009.

[14] E. Yoon, K. Yoo, "Robust ID-based remote mutual authentication with key agreement protocol for mobile devices on ECC", in: *Proceedings of CSE 2009*, Vancouver, Canada, pp. 633–640, 2009.

[15] T. H. Chen, Y. C. Chen, W. K. Shih, "An advanced ECC ID-based remote mutual authentication scheme for mobile devices", in: *Proceedings of ATC 2010*, Xian, China, pp.116–120, 2010.

[16] S. H. Islam, G. P. Biswas, "A more efficient and secure ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem", *Journal of Systems and Software*, vol.84, no.11, pp. 1892-1898, 2011.

Ming Luo received the B.E. and Ph.D degree from Northeastern University, Shenyang, China in 2004 and 2010, respectively. Now he is an associate professor in the School of Software, Nanchang University, Nanchang, China. He has won lots of scholarships in China and was supported by the National Natural Science Foundation of China under grant no. 60602061, 60803131 and 11226042, the National High-Tech Research and Development Plan of China under grant no. 2006AA01Z413 and the Science and Technology Supporting Program of Jiangxi Province under grant no. 2012ZBBE50036. His research interests are information security, networks security and cryptography.

Donghua Huang received the B.E. degree form the School of Software, Nanchang University in July 2012. He is currently pursuing his M.E degree from the School of Software, Nanchang University. His current research interests include networks security and cryptography.

Jun Hu received his PhD degree from Beijing Institute of Technology, Beijing, China in 2003. He is currently a professor in the School of Software, Nanchang University, Nanchang, China. He is a director of Chinese Association for Artificial Intelligence and a senior member of China Computer Federation. He has participated in a number of in the computer area, such as: the National High-Tech Research and Development Plan of China, National Natural Science Foundation of China, Ten Five-Year Plan of General Armament Department of China, and so on. His research interests include network security, electronic commerce and artificial intelligence.