An enhanced ID-based remote mutual authentication with key agreement protocol for mobile devices on elliptic curve cryptosystem

He Debiao*, Chen Jianhua, Hu Jin
Mathematics & Statistics School of Wuhan University, Wuhan, Hubei province, 430072, China

Abstract: Recently, Yoon et al. and Wu proposed two improved remote mutual authentication and key agreement scheme for mobile devices on elliptic curve cryptosystem. In this paper, we show that Yoon et al.'s protocol fails to provide explicit key perfect forward secrecy and fails to achieve explicit key confirmation. We also point out Wu's scheme decreases efficiency by using the double secret keys and is vulnerable to the password guessing attack and the forgery attack. In order to overcome the drawback, we proposed and improved scheme. Through the comparison with other protocol, we believe that our improved scheme is more suitable for real-life applications.
Key words: ID-based; Mutual authentication; Key agreement; Elliptic curve Cryptosystem; Perfect forward secrecy; Modular multiplication

1. Introduction

With the rapid development of the development of electronic technology, various mobile devices (e.g., cell phone, PDA, and notebook PC) are produced and people's life is made more convenient. More and more electronic transactions for mobile devices are implemented on Internet or wireless networks. In electronic transactions, remote user authentication in insecure channel is an important issue. For example, when one user wants to login a remote server and access its services, such as on-line shopping and pay-TV, both the user and the server must authenticate the identity with each other for the fair transaction.

Generally, the remote user authentication can be implemented by the traditional public-key cryptography (Rivest et al., 1978; ElGama, l985). The computation ability and battery capacity of mobile devices are limited, so traditional public-key cryptograph, in which the computation of modular exponentiation is needed, can't be used in mobile devices. Fortunately, Elliptic curve cryptosystem (ECC) (Miller, 1986; Koblitz, 1987) has significant advantages like smaller key sizes, faster computations compared with other public-key cryptography. Thus, ECC-based authentication protocols are more suitable for mobile devices than other cryptosystem. However, like other public-key cryptography, ECC also needs a key authentication center (KAC) to maintain the certificates for users' public keys. When the number of users is increased, KAC needs a large storage space to store users' public keys and certificates. In addition, users need additional

*Corresponding author
E-mail: hedebiao@163.com, *Tel*:+0086015307184927, *Fax*: +008602787817667

computations to verify the other's certificate in these protocols

To solve the above problems, several ID-based authentication protocols on ECC are proposed (Abichar et al., 2007; Choie et al., 2005; Cao et al., 2008; Chen and Song, 2007; Jiang C et al., 2007; Jia Z. et al., 2006; Tian et al., 2005; Wu et al.,2005). But there are some disadvantages in the previous user authentication protocols on ECC (Yang et al. 2009). That is, some of these protocols do not provide the mutual authentication (Chen and Song, 2007; Jiang et al., 2007; Jia et al., 2006; Wuet al., 2005) or the session key agreement (Cao et al., 2008; Chen and Song, 2007; Jia et al., 2006;Wuet al., 2005) between the user and the server. For some applications, the user and the server need a session key to encrypt the secret information for the subsequent communications after they authenticate with each other.

In 2009, Yang et al. propose the first ID-based remote mutual authentication with key agreement protocol on ECC (Yang et al., 2009). Based upon the ID-based concept, the protocol does not require public keys for users so that the additional computations for certificates can be reduced. Moreover, the protocol not only provides mutual authentication but also supports a session key agreement between the user and the server. Recently, Yoon et al. (Yoon et al., 2009) found Yang et al.'s protocol is vulnerable to an impersonation attack and does not provide perfect forward secrecy. At the same time, Wu (Wu, 2009) pointed out Yang et al.'s protocol depends solely on a long-term private key stored in the mobile device, does not provide perfect forward secrecy and does not consider personal privacy problem.

Nevertheless, we find Yang et al.'s protocol does not provide perfect forward secrecy and fails to achieve forward secrecy. We also find Wu's protocol is vulnerable to the password guessing attack and the forgery attack. In addition, Wu's protocol decreases efficiency by using the double secret keys. In this paper, we propose an efficient ID-based remote mutual authentication with key agreement protocol for mobile devices on elliptic curve cryptosystem. Compared with that of Yang et al., Yoon et al. and Wu, the proposed protocol is more secure, efficient, and more suitable for mobile devices.

The rest of our paper is organized as follows. Section 2 gives the some basic concept. Section 3 reviews the protocols of Yoon et al. and Wu. Section 4 analyzes the security of the protocols of Yoon et al. and Wu. Section 5 and Section 6 propose our protocol and the security of the proposed protocol. Finally, Section 7 concludes the paper.

2. Preliminaries

2.1 Notations

We first introduce common notations used in this paper as follows.

- F_p : a finite field;
- E : an elliptic curve defined on finite field F_p with large order;
- G : the group of elliptic curve points on E ;
- P : a point on elliptic curve E with order n , where n is a large prime number;

- $H_1(\cdot)$: a secure one-way hash function, where $H_1:\{0,1\}^* \to G$;
- $H_2(\cdot)$: a secure one-way hash function, where $H_2:\{0,1\}^* \to Z_p^*$;
- $H_3(\cdot)$: a secure one-way hash function, where $H_3:\{0,1\}^* \to Z_p^*$;
- $H_4(\cdot)$: a secure one-way hash function, where $H_4:\{0,1\}^* \to Z_p^*$;
- U : the user;
- S : the server;
- ID_U : the identity of the user U ;
- ID_S : the identity of the server S ;
- (q_S, Q_S): the server S 's private/public key pair, where $Q_S = q_S \cdot P$.

2.2 Background of elliptic curve cryptograph

We will just give a simple introduction of elliptic curve defined on prime field F_p. The knowledge of elliptic curve defined on binary field can be found in (Miller, 1986; Koblitz, 1987).

Let the symbol E/F_p denote an elliptic curve E over a prime finite field F_p, defined by an equation

$$y^2 = x^3 + ax + b\ ,\quad a,b \in F_p \qquad (1)$$

and with the discriminant

$$\Delta = 4a^3 + 27b^2 \neq 0\ . \qquad (2)$$

The points on E/F_p together with an extra point O called the point at infinity form a group

$$G = \{(x,y): x,y \in F_p, E(x,y) = 0\} \cup \{O\}\ . \qquad (3)$$

Let the order of G is n, G is a cyclic additive group under the point addition "+" defined as follows: Let $P,Q \in G$, l be the line containing P and Q (tangent line to E/F_p if P = Q), and R , the third point of intersection of l with E/F_p . Let l' be the line connecting R and O . Then P "+" Q is the point such that l' intersects E/F_p at R and O and

P “+” Q. Scalar multiplication over E / F_p can be computed as follows:

$$tP = P + P + \cdots + P(t \ times) \quad (4).$$

3. Review of Two Protocols

3.1 Yoon et al.’s Protocol

Yoon et al.’s protocol consists of three phases: system initialization phase, user registration phase, and mutual authentication with key agreement phase.

- **System initializing phase**

In this phase, S generates parameter of the system.

1). S chooses an elliptic curve E over a finite field F_p. Let $E(F_p)$ denote the set of all the point on E.

2). S chooses a point $P \in E(F_p)$, such that the subgroup generated by P has a large order n.

3). S chooses three hash functions $H_1(\cdot), H_2(\cdot), H_3(\cdot)$ described in section 2.1.

4). S publishes the parameter $(p, E, G, n, H_1(\cdot), H_2(\cdot), H_3(\cdot))$.

- **User registration phase**

In this phase, everyone who wants to register at the server should obtain a smart card. The user U begins his registration at the server S as shown in Fig 1.

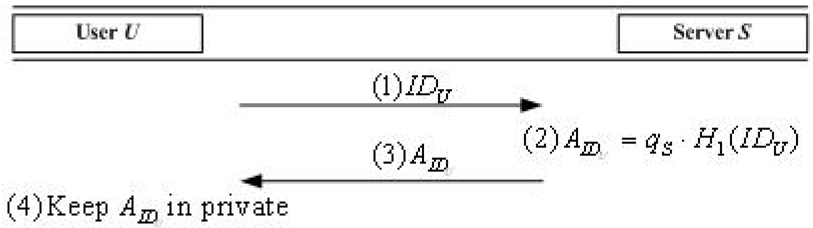

Fig. 1. User registration phase of Yoon et al.’s protocol

- **Mutual authentication with key agreement phase**

In this phase, the user U sends a login request message to the server S whenever U wants to access some resources upon S. Then the server S verifies the authenticity of the login message requested by the user U. At the same time, a session generation between U and S is generated. The detailed of the phase is illustrated in Fig 2.

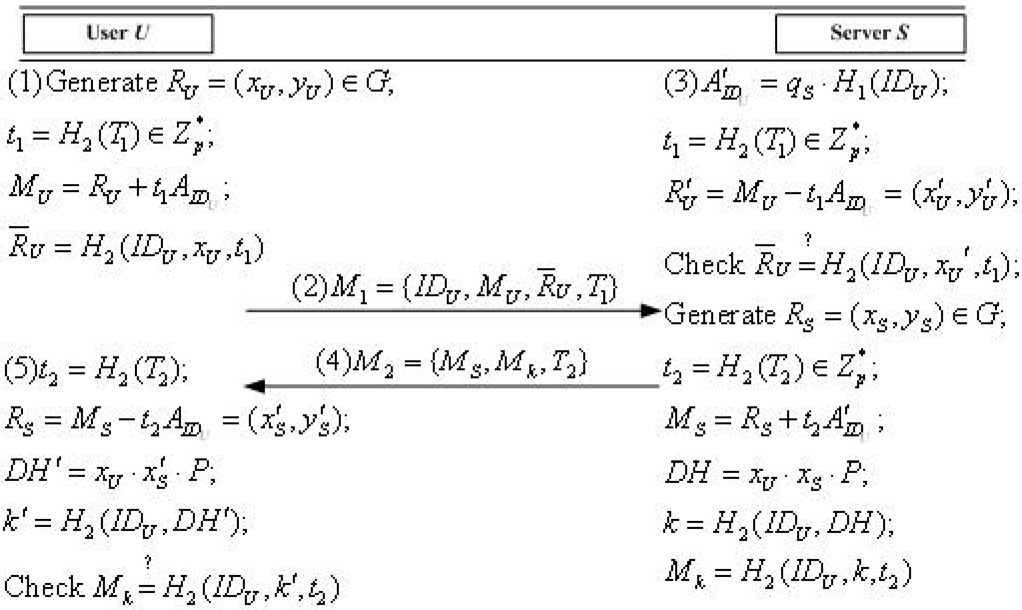

Fig. 2. Mutual authentication with key agreement phase of Yoon et al.'s protocol

3.2 Wu's Protocol

Wu's protocol also consists of three phases: system initialization phase, user registration phase, and mutual authentication with key agreement phase.

- **System initializing phase**

In this phase, S generates parameter of the system.

1). S chooses an elliptic curve E over a finite field F_p. Let $E(F_p)$ denote the set of all the point on E.

2). S chooses a point $P \in E(F_p)$, such that the subgroup generated by P has a large order n.

3). S chooses three hash functions $H_2(\cdot), H_3(\cdot), H_4(\cdot)$ described in section 2.1.

4). S computes private/public key pair (q_S, Q_S) and selects a private key d_S.

5). S publishes the parameter $(p, E, G, n, H_2(\cdot), H_3(\cdot), H_4(\cdot))$.

- **User registration phase**

In this phase, everyone who wants to register at the server should obtain a smart card. The user U begins his registration at the server S as shown in Fig 3.

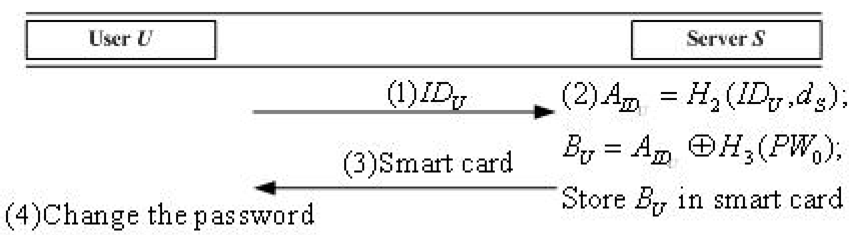

Fig. 3. User registration phase of Wu's protocol

- **Mutual authentication with key agreement phase**

In this phase, the user U sends a login request message to the server S whenever U wants to access some resources upon S. Then the server S verifies the authenticity of the login message requested by the user U. At the same time, a session generation between U and S is generated. The detailed of the phase is illustrated in Fig 2.

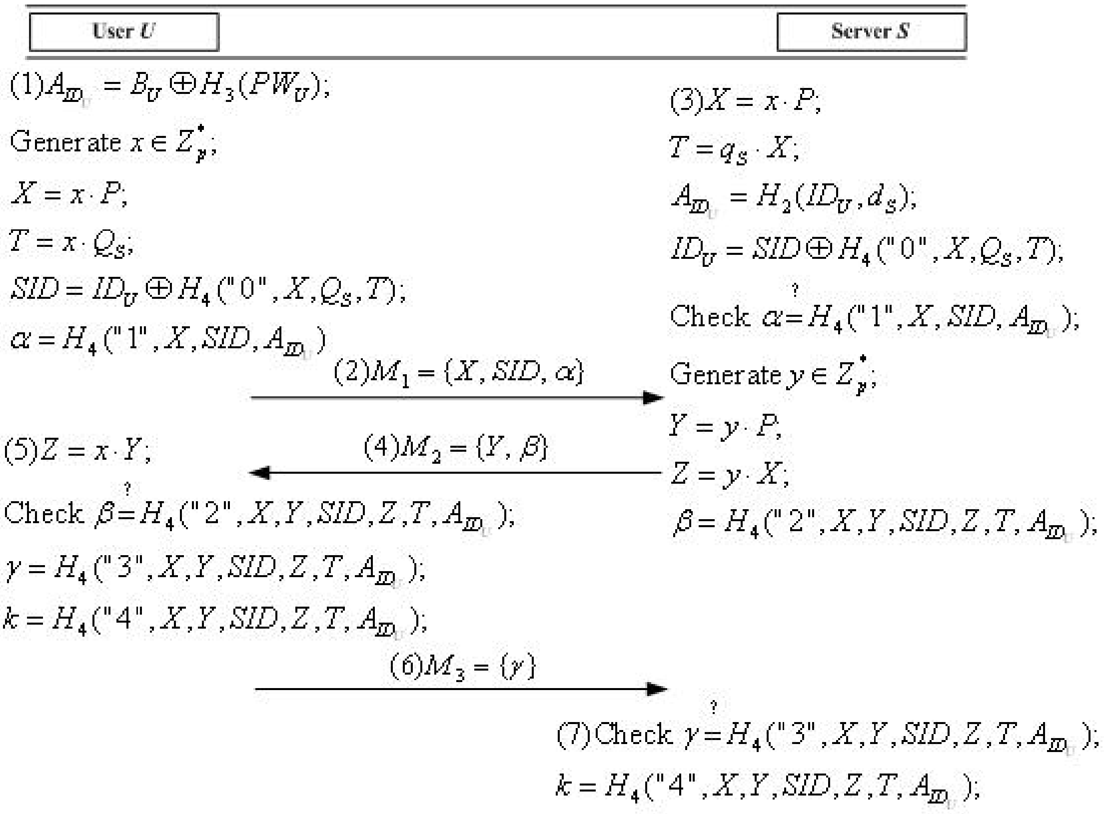

Fig. 2. Mutual authentication with key agreement phase of Yoon et al.'s protocol

4. Analysis of Two Protocols

4.1 Analysis of Yoon et al.'s Protocol

This section shows that Yoon et al.'s protocol does not provide perfect forward secrecy and does not achieve explicit key confirmation.

- **Failure to provide explicit key perfect forward secrecy**

Perfect forward secrecy is one of desirable attributes of key agreement protocols, it means that if the long-term private keys of one or more entities are compromised, the secrecy of previous session keys, which was established by honest entities, is not affected (Blake-Wilson S. et al., 1997).

We find that Yoon et al.'s protocol could not provide perfect forward secrecy. The following is our reasons.

In Yoon et al.'s protocol, since all transcripts are transmitted over an open network, a benign (passive) adversary can easily obtain a valid information pair $\{ID_U, M_U, \overline{R}_U, T_1\}$ and $\{M_S, M_k, T_2\}$.

If the long-term private key A_{ID_U} of the user is compromised, and the key is derived by the attacker A, then A can compute all the session key generated between the user and the server as follow.

1) A computes $t_1 = H_2(T_1)$, and $R_U = (x_U, y_U) = M_U - t_1 A_{ID_U}$.

2) A computes $t_2 = H_2(T_2)$, and $R_S = (x_S, y_S) = M_S - t_2 A_{ID_U}$.

3) A computes $DH = x_U \cdot x_S \cdot P$.

4) The attacker A get the session key $k = H_2(ID_U, DH);$.

If the long-term private key q_s of the server S is compromised, the attacker A can compute the session key at the same way, because the attacker A can get the long-term private key A_{ID_U} of any user U by computing $A_{ID_U} = q_s \cdot H_1(ID_U)$.

Since the attacker can get the session key through the method described above, then we can conclude that Yoon et al.'s protocol does not provide perfect forward.

- **Failure to achieve explicit key confirmation**

A key agreement scheme is said to provide the explicit key confirmation if one entity is assured that the second entity has actually computed the session key (Blake-Wilson S. et al., 1997). In many applications, it is highly desirable for a key agreement scheme to provide the explicit key confirmation. We can see that the scheme of Yoon et al. merely provides the implicit key confirmation, because S cannot confirm U has correctly computed the session key after the Mutual authentication with key agreement phase. However, in general, key agreement scheme can provide the explicit key confirmation. Hence, the scheme of Yoon et al. is not practical for application.

4.2 Analysis of Wu's Protocol

- **Inefficiency of double secret keys**

We can see that the scheme of Wu requires S to keep two keys secret, i.e., the secret key d_S and the private key q_S for the elliptic curve algorithm. In common sense, it is possible to only use one secret key for achieving the user authentication and key agreement service. Therefore, two secret keys mean more overheads without the security enhancement for the whole authentication system. Furthermore, we need to point out the drawback of using the elliptic-curve algorithm in the scheme of Wu. Since S uses the private/public key pair (q_S, Q_S), this elliptic-curve algorithm is a public key algorithm, which may involve the certificate mechanism, e.g., X.509 (ITU-T, 2005). To maintain the certificate framework, the public key infrastructure incurs a nontrivial level of system complexity and implementation costs.

- **Vulnerable to password guessing attack and forgery attack**

We assume that an attacker A has total control over the communication channel between the user U and the remote server S, which means that he can insert, delete, or alter any messages in the channel. According to the researches in (Kocher et al.'s, 1999; Messerges et al.'s 2002), all existing smart cards are vulnerable since the secret values stored in a smart card could be extracted by monitoring its power consumption. Therefore, we further assume that the attacker A can steal the user's smart card and extract the values stored in the smart card. Under these two assumptions, we will examine some security flaws of Wu's remote user authentication method.

The server S stores B_U into the smart card of the user U in the registration phase. If the attacker A steals the smart card and extracts the secret values from the smart card as in (Kocher et al.'s, 1999; Messerges et al.'s 2002), he can then easily figure out U's password as follow.

1) A get a message $M_1 = \{X, SID, \alpha\}$ transmitted between U and S.
2) A selects a password PW' s from a uniformly distributed dictionary D.
3) A computes $A'_{ID_U} = B_U \oplus H_3(PW')$.
4) A computes $\alpha' = H_4(\text{"1"}, X, SID, A'_{ID_U})$
5) A then verify the correctness of PW' s by checking that α is equal to α'.
6) A repeats steps 1, 2, and 3 of this phase until the correct password if found.

After the adversary has obtained the password PW_U (using the above method), since she has also B_U, A can compute $A_{ID_U} = B_U \oplus H_3(PW_U)$. In this way he can impersonate U by forging her login message $\{X, SID, \alpha\}$ and $\{\gamma\}$. Therefore, Wu's scheme is vulnerable to forgery attacks. Please observe that the results of a successful guessing attack can be used to forge a valid login message and carry out a forgery attacks.

5. Our improved protocol

In this section, we propose an improved scheme to overcome those disadvantages existing in Yoon et al.'s protocol and Wu's protocol while the merits of the original scheme are left unchanged. The proposed scheme is divided into three phases: system initialization phase, user registration phase, and mutual authentication with key agreement phase.

- **System initializing phase**

In this phase, S generates parameter of the system.

1). S chooses an elliptic curve equation E .
2). S selects a base point P with the order n over E.

3). S selects its master key q_s.

4). The server chooses three secure one-way hash functions $H_1(\cdot), H_2(\cdot), H_3(\cdot)$ described in section 2.1.

5). The server keeps q_s in private and publishes (F_p, E, n, P, H_1, H_2, H_3).

- **User registration phase**

In this phase, everyone who wants to register at the server should obtain a smart card. The user U begins his registration at the server S, shown in Fig 5, as follows.

1). The user U sends his identity ID_U to the server S.

2). S computes $A_{ID_U} = \frac{1}{q_s + H_2(ID_U)} P \in G$ and $B_U = A_{ID_U} \oplus H_1(PW_0)$, where PW_0 is the initial password. Then, S store ID_U and B_U in a smart card. At last, S issues U the smart card.

3). Upon receiving the smart card, U change his password at once.

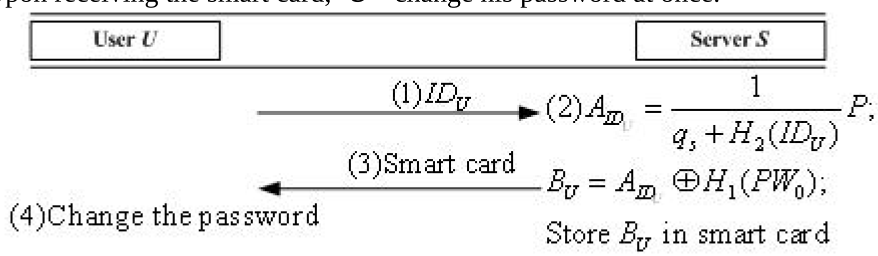

Fig. 5. User registration phase of our protocol

- **Mutual authentication with key agreement phase**

In this phase, the user U sends a login request message to the server S whenever U wants to access some resources upon S. Then the server S verifies the authenticity of the login message requested by the user U. At the same time, a session generation between U and S is generated. The detailed of the phase, shown in Fig. 6, is illustrated as follows.

1). The user U insert his smart card and input his password PW_U. U's smart card computes $A_{ID_U} = B_U \oplus H_1(PW_0)$.

2). U's smart card chooses a random number $x \in Z_n^*$, and computer $X = x \cdot P$, $\overline{X} = x \cdot A_{ID_U}$. Then U's smart card computes $\alpha = H_2("1", ID_U, X, \overline{X}, T_1)$, where T_1 is a timestamp denotes the current time. Finally, U's smart card sends $M_1 = \{\overline{X}, ID_U, \alpha, T_1\}$ to the server.

3). After receiving $M_1 = \{\overline{X}, ID_U, \alpha, T_1\}$, S checks the validity of ID_U and the freshness of T_1. The freshness of T_1 is checked by performing $T' - T_1 \le \Delta T$, where T' is the time that S receives the above message and ΔT is a valid time interval. If ID_U is not valid or T_1 is not fresh, S aborts the current session. S computes $X' = (q_s + H_2(ID_U)) \cdot \overline{X}$ and $\alpha' = H_2("1", ID_U, X', \overline{X}, T_1)$. Then, S checks if $\alpha = \alpha'$ holds. If the equation does not holds, the server aborts the current session. S chooses a random number $y \in Z_n^*$, and computes $Y = y \cdot P$, $Z = y \cdot X$ and $\beta = H_2("2", ID_U, X', \overline{X}, Y, Z, T_2)$ where T_2 is a timestamp denotes the current time. At last, S sends $M_2 = \{Y, \beta, T_2\}$ to the server.

4). Upon receiving $M_2 = \{Y, \beta, T_2\}$, U 's smart card checks the freshness of T_2. The freshness of T_2 is checked by performing $T'' - T_2 \le \Delta T$, where T'' is the time U 's smart card receives the above message and ΔT is a valid time interval. If T_2 is not fresh, U 's smart card aborts the current session. U 's smart card computes $Z' = x \cdot Y$ and $\beta' = H_2("2", ID_U, X, \overline{X}, Y, Z', T_2)$. Then, U 's smart card checks if $\beta = \beta'$ holds. If the equation does not holds, U 's smart card aborts the current session. Then U 's smart card computes $\gamma = H_2("3", ID_U, X, \overline{X}, Y, Z', T_1, T_2)$, $k = H_3("4", ID_U, X, \overline{X}, Y, Z', T_1, T_2)$ and sends $M_3 = \{\gamma\}$ to the server.

5). After receiving $M_3 = \{\gamma\}$, S computes $\gamma' = H_2("3", ID_U, X', \overline{X}, Y, Z, T_1, T_2)$ checks if $\gamma = \gamma'$ holds. If the equation does not holds, the server aborts the current session. At last, S computes the session key $k = H_3("4", ID_U, X', \overline{X}, Y, Z, T_1, T_2)$ and accepts the request.

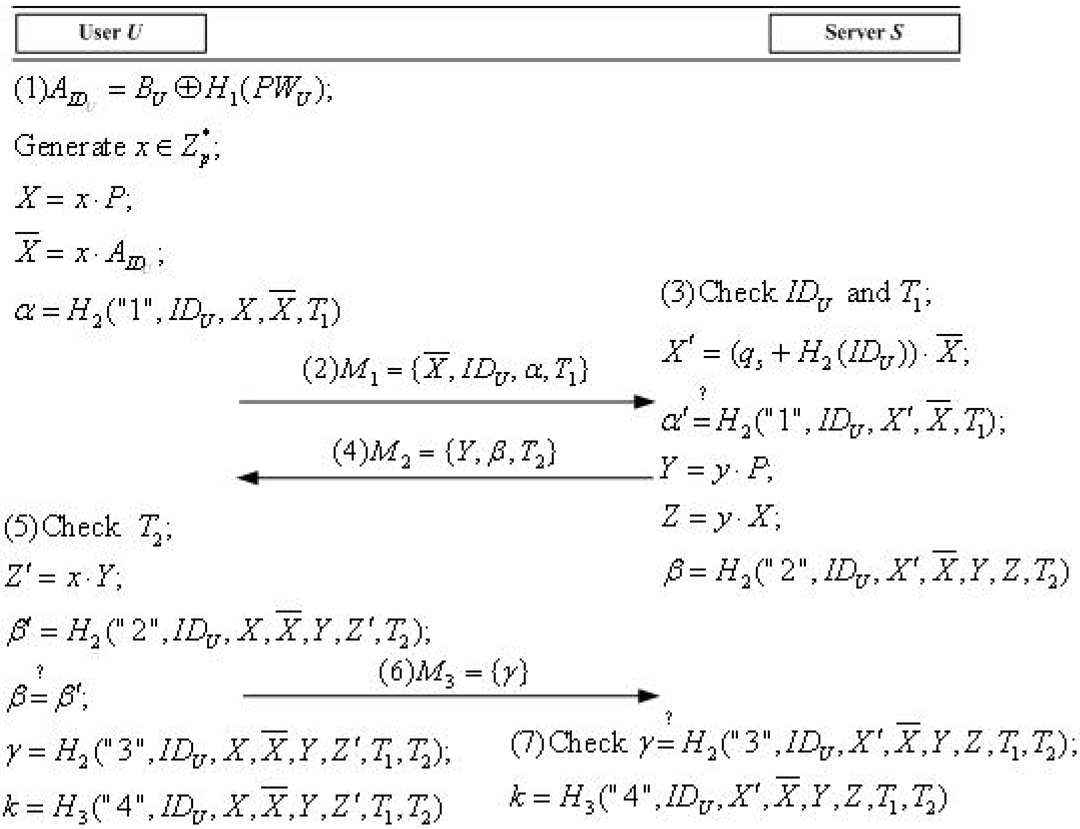

Fig. 6. Mutual authentication with key agreement phase of our protocol

6. Security and Efficiency Discussion

6.1 Security Discussion

In this section, we analyze the security of our improved scheme by discussing resistance to characteristic attacks on schemes of this type.

- **Mutual authentication**

Mutual authentication means that both the server and the user can authenticate each other before generating the common session key.

In the authentication phase of our scheme, S has to verify the validity of α, and U's smart card has to verify the validity of β in order to authenticates S. If the attack wants to forge the message, he will face *ECDLP*. When both the validity of α and β are confirmed by S and U respectively, the mutual authentication between them is achieved.

- **Perfect forward and backward secrecy**

Perfect forward and backward secrecy means that if an intruder gets the session key, he can't reconstruct any previous or subsequent session keys. In our improvements, the compromised password or the master key q_s can't be used to reconstruct any previous or subsequent session keys for that we use the Diffie-Hellman key agreement scheme.

If an attacker A gets the password in our scheme, he may get A_{ID_U}. But he can't compute X if he does not know the master key q_s.

If the master key q_s is compromised, then A can computes $X = x \cdot P$ and gets $Y = y \cdot P$, but he can't deduce $Z = x \cdot y \cdot P$, without the knowledge of the two random numbers, x and y. Therefore, our scheme can provide perfect forward and backward secrecy.

- **Key freshness**

Key freshness means that the key used in each session is different from the ones used in other sessions. Since each party picks his random nonce secretly when computing the session key in our protocol, it can be easily seen that the freshness of the used session keys in our scheme is guaranteed.

- **Preventing the replay attack**

Replay attack means that a legal peer's transmission message is intercepted and replayed by an adversary for fooling another legal peer to regard him as authentic. However, the fresh nonces chosen at each protocol run are used to avoid such replay attacks in our improvements.

- **Preventing the off-line password guessing attack**

Off-line password guessing attack means that a passive attacker intercepts the communication line between a legal client and the server, and tries to guess the client's password off line. In the following, we prove why our scheme can resist against such an off-line password guessing attack.

The attack A may intercepts $M_1 = \{\overline{X}, ID_U, \alpha, T_1\}$, $M_2 = \{Y, \beta, T_2\}$ and $M_3 = \{\gamma\}$. A may get B_U stored the smart card. Then A could guess a password PW'. But A can't verify the correctness of the PW', since he will face the *ECDLP*.

- **Preventing the insider attack**

Insider attack means that a legal client D can impersonate another legal client U to gain the service of server S.

Assume that D wants to impersonate U to login to S. However, without the knowledge A_{ID_U}, D can not construct a valid message. Therefore, our scheme can withstand the insider attack.

- **Preventing man-in-the-middle attack**

Man-in-the-middle attack means that an active attacker intercepts the communication line between a legal user and the server and uses some means to successfully masquerade as both the server to the user and the user to the server. Then, the user will believe that he is talking to the intended server and vice versa.

In our scheme, the attack A can't generate the valid M_1 and M_3 without the value of A_{ID_U} and A can not generate the valid M_2 without the value of q_S. S and U will find the attack through check the correctness of α or β separately.

- **Preventing the on-line password guessing attack**

Suffering on-line password guessing attack means that an attacker can successfully guess a legal user's password on line. Since our scheme has the mutual authentication function. Only the user with the right password can pass the authentication of the server. Therefore, any attempt to launch a password guessing attack will be detected by the server. Moreover, we can set both improvements to tolerate some times of wrong password logins, e.g., three time. If the number of wrong login times is reached, the system would reject the login request. Under such a setting, our scheme can resist the on-line password guessing attack.

- **Preventing smart-card-lost attack**

Smart-card-lost attack means an attacker can launch various attacks when he gets a legal user's smart card. In the following, we discuss two of the most common attacks launched under such a situation, off-line password guessing attack and impersonation attack.

1) Suppose U 's smart card is lost and obtained by A. Through, A can read B_U in U 's smart card. Then A could guess a password PW'. But A can't verify the correctness of the PW', since he will face the *ECDLP*..
2) If A impersonates U to login in the server. He can not construct the valid message M_1, since he doesn't the value A_{ID_U}. Then the impersonation attack will be found by the server.

6.2 Efficiency Discussion

We let PM, PA, H and MM denote elliptic curve multiplication, elliptic curve addition, hash operation and modular multiplication separately. The operations which have to be performed in the mutual authentication with key agreement phase are given in Table 1, whereas the operation that have to be performed in the registration phase are not taken into consideration since they are computed just for the first time and thus have little influence on the efficiency of the scheme.

Table 1. Efficiency of the schemes

Operation	Yang et al.'s protocol		Yoon et al.'s protocol		Wu's protocol		Our protocol	
	U's smart card	Server	*U*'s smart card	Server	*U*'s smart card	Server	*U*'s smart card	Server
PM	4	3	3	4	3	3	3	3
H	4	4	5	6	6	6	4	4
PA	2	2	2	2	0	0	0	0
MM	0	1	1	1	0	0	0	0

It can be observed that our scheme is simpler and efficient than other scheme. Although the improved scheme can not provide anonymity for the user's identity like Wu's protocol does, it resolves the security issues and is therefore more secure than that of Yang et al., Yoon et al. and

Wu. Moreover, compared with Wu's protocol, the server in our protocol uses only on private key and does not uses the private/public key pair (q_S, Q_S) (this is elliptic-curve algorithm is a public key algorithm, which may involve the certificate mechanism, e.g., X.509 (ITU-T, 2005)). Then our protocol is more efficient than Wu's protocol.

7. Conclusion

In this paper, we review Yang et al.'s protocol and Wu's protocol then point out the security vulnerability of the two protocols. In order to overcome the weakness of the two protocols, we propose an improved protocol. In addition, our protocols increases the efficiency by letting the server use only one private key and not use the private/public key pair. Therefore, we believe that our improved scheme is more suitable for real-life applications than that of Yang et al., Yoon et al. and Wu.

Reference

[1] Abichar PE, Mhamed A, Elhassan B. A fast and secure elliptic curve based authenticated key agreement protocol for low power mobile communications. In: Proceedings of the 2007 international conference on next generation mobile applications, services and technologies; 2007:235–240.

[2] Blake-Wilson S., Johnson D. and. Menezes A, "Key Agreement Protocols and Their Security Analysis", P roceedings of Sixth IMA International Conference on Cryptography and Coding, Cirencester, UK, 1997:30-45.

[3] Cao X, Kou W, Dang L, Zhao B. IMBAS: identity-based multi-user broadcast authentication in wireless sensor networks. Computer Communications 2008, 31:659–671.

[4] Chen ZG, Song XX. A distributed electronic authentication protocol based on elliptic curve. In: Proceedings of the sixth international on machine learning and cybernetics; 2007: 2179–2182.

[5] ElGamal T. A public key cryptosystem and a signature protocol based on discrete logarithms. IEEE Transactions on Information 1985;IT-31:469–72.

[6] Institute of Electrical and Electronics Engineers, IEEE Standard Specifications for Public-Key Cryptography, http://grouper.ieee.org/groups/1363/.

[7] ITU-T, Recommendation X.509-the Directory: Public-Key and Attribute Certificate Frameworks, ITU-T Study Group 17, Aug. 2005. (equivalent to ISO/IEC 9594-8).

[8] Jia Z, Zhang Y, Shao H, Lin Y, Wang J. A remote user authentication protocol using bilinear pairings and ECC. In:Proceedings of the sixth international conference on intelligent system design and applications; 2006:1091–1094.

[9] Jiang C, Li B, Xu H. An efficient protocol for user authentication in wireless sensor networks. In: Proceedings of 21st international conference on advanced information networking and applications workshops; 2007:438–442.

[10] Kocher P., Jaffe J., Jun B., Differential power analysis, Proceedings of Advances in Cryptology (1999) 388 – 397.
[11] Koblitz N. Elliptic curve cryptosystem. Mathematics of Computation 1987, 48:203–209.
[12] Messerges T.S., Dabbish E.A., R.H. Sloan, Examining smart-card security under the threat of power analysis attacks, IEEE Transactions on Computers 51 (5) (2002) 541 – 552.
[13] Miller V.S. Use of elliptic curves in cryptography. In: Advances in cryptology, proceedings of CRYPTO'85, vol. 218. LNCS, Springer-Verlag; 1986: 417–426.
[14] Rivest RL, Shamir A, Adleman L. A method for obtaining digital signatures and public key cryptosystems. Communications of the ACM 1978;21(2):120–126.
[15] Tian X, Wong DS, Zhu RW. Analysis and improvement of authenticated key exchange protocol for sensor networks. IEEE Communications Letters 2005;9(11):970–972.
[16] Wu S., Practical remote mutual authentication with key agreement protocol for mobile devices on elliptic curve cryptosystem, 2009, http://eprint.iacr.org/2009/536.pdf.
[17] Wu ST, Chiu JH, Chieu BC. ID-based remote authentication with smart cards on open distributed system from elliptic curve cryptography. In: Proceedings of IEEE international conference on electro information technology; 2005.
[18] Yang J.H., Chang C.C., An ID-based remote mutual authentication with key agreement protocol for mobile devices on elliptic curve cryptosystem, computers & security 28(2009) 138 – 143.
[19] Yoon E.-J., Yoo K.-Y., Robust ID-based Remote Mutual Authentication with Key Agreement Protocol for Mobile Devices on ECC, 2009 International Conference on Computational Science and Engineering, 2009, pp. 633-640.