A key agreement scheme for identity authentication based on multiple security factors

Ying Wang[1*], Xinguang Peng[1], Jing Bian[1,2]

[1]*College of Computer Science and Technology, Taiyuan University of Technology, Shanxi, Taiyuan China, 030024*

[2]*The Center of Information and Network, Shanxi Medical College of Continuing Education, Shanxi, Taiyuan China, 030012*

Received 1 October 2014, www.cmnt.lv

Abstract

By the study of Li-Hwang's three-factor authenticated key agreement protocol, we find there exists multiple attack in it. Therefore, this paper proposes a remote user authentication protocol based on ECC to improve the protocol. The new protocol only needs one point multiplication of elliptic curve cryptography in login and authentication protocol, so its computation efficiency is high. The protocol can resist all the attack mentioned in this paper, smart cards loss attack included. It is also suited to three-factor authentication protocol including password, mobile devices and biometrics. The comparisons with related schemes in computation price and efficiency show that though it costs a little more computation consumption than relative protocols, it has better performance in robustness.

Keywords: authentication, attack, biometric, smart card, ECC

1 Introduction

Authentication is a verification process for the authenticity of some object. It is the most important part in computer security, since most security services are based on authentication. The authentication protocol just based on password has been widely applied due to its memorable feature. However, such single-factor protocol is not suited to applications with higher security. The reason is that the password may be stolen or guessed easily. The smart card is also used by people in multiple fields by its excellent features. While it may be forged, stolen or cracked, losing the expectation of the users. Thus, most of recent protocols design combines the password with smart card [1-3]. To further improve the security of password-based authentication protocols, many scholars propose multi-factor authentication protocol based on biometrics. Such protocols have more advantages than single-factor and double-factor authentication protocols. The biometric are hard to be forgotten or lost, and they are difficult to be copied or shared [4, 5].

Tan, et al. believed that there was offline password guessing attack and server compromised impersonation attack in initial protocol based on HTTP digest authentication [6]. They proposed a protocol based on discrete logarithm problem and Diffie-Hellman key exchange algorithm. But this protocol is not suited to the environment with limit resources and low computation capability. Cao et al. proposed a SIP authentication protocol based on elliptic curve cryptosystem (ECC) [7]. But the problems of password guessing attacks and Denning-Sacco attacks still exist. In 2008 in order to improve the execution efficiency of protocol, Tsai et al. proposed a SIP protocol based on random nonce [8]. Tang also proposed an ECC-based SIP authentication protocol in 2009 [9]. In contrast, Tsai's protocol has higher efficiency than Tang and Cao, so it is more appropriate to be applied to some special environments. However, Yoon pointed that the protocol of CAO, Tang and Tsai would suffer off-line password guessing attack, Denning-Sacco attack and stolen validation data attack. Then they proposed a three-factor SIP authentication protocol based on elliptic curve discrete logarithm and proved its performances in anti-attacks [11]. Yoo et al. proposed a multi-factor remote user authentication protocol based on smart card, biometric and password [12]. Their protocol only depends on Hash operation or XOR operation so it has higher efficiency. Multi-factor authentication protocol offers mutual authentication among users and remote servers, to provide stronger security than single-factor protocol or double-factor protocol. In 2011, Li and Hwang [13] pointed Yoo's protocol had some problems in password misuse and denial of service (DoS). Meanwhile, Li-Hwang's protocol is also based on Hash computation and XOR operation, which shows higher efficiency. However, we find some attacks will play effect in this protocol such as privileged attack, server compromised impersonation attack, parallel attack, oracle attack and DoS attack, so it is not suited to the implementation in real environment.

We propose an improved remote user authentication scheme in this paper to solve the security problems in Li-Hwang's scheme. The new scheme keeps high efficiency by just one point multiplication of elliptic curve in the process of authentication. It can resist the attacks mentioned above so it is suited to the three-factor authentication protocol. Our scheme is verified to be much safer than Li-Hwang's protocol, keeping lower

[*] ***Corresponding author's*** e-mail: wangying@tyu.edu.cn

computation consumption and communication cost simultaneously.

2 SOFC model

2.1 LI-HWANG'S THREE-FACTOR AUTHENTICATION PROTOCOL

Li-Hwang's scheme consists of register protocol, login protocol, authentication protocol and password modification protocol. The symbols in protocol are defined as Table 1.

2.1.1 Registration protocol

U_A must register at registration centre first to get the sources on remote servers. The procedures are:

1) U_A inputs its biometric BIO_A to the devices of registration centre. Password PWD_A and identity ID_A are also input at the same time.

2) Registration centre R will compute the following parameters for users:

$$f_A = h(BIO_A),$$

$$r_A = h(PWD_A) \oplus f_A,$$

$$e_A = h(ID_A \| x) \oplus r_A.$$

3) At last, R saves parameter (ID_A, $h(.)$, f_A, e_A, r_A) in the smart card and distributes the smart card to U_A through the security channel.

TABLE 1 Symbol definition

Symbol	Meaning
U_A	User A
ID_A	Identity of U_A
PWD_A	Password shared by U_A and server
S	Remote server
BIO_A	Biometric template value of U_A
R	Credible registry center
P_s	Public key of server
x	Private key of server
$f(.,.)$	Symmetrical parameters function
$h()$	One-way Hash function
$\oplus$	XOR operator
$\|$:	String concatenation operator
τ	Biometric verification threshold

2.1.2 Login protocol

In login protocol, if U_A wants to login S, it must follow the rules below:

1) U_A Inserts the smart card into card reader terminal. Then its feature template value BIO_A is input. The smart card will check whether $h(B_A)$ is equal to f_A that is saved in smart card.

2) If they are not equal, the authentication process is terminated; otherwise, the user will input PWD_A and go to the next step.

3) The smart card computes $r_A^{'} = h(PWD_A) \oplus f_A$. If $r_A^{'} \neq r_A$, the password checking fails and the client software will terminate this session; otherwise, the smart card goes to the next step.

4) The related parameters are computed as follows:

$$M_1 = e_A \oplus r_A^{'} = h(ID_A \| x),$$

$$M_2 = M_1 \text{Å} R_c = h(ID_A \| x) \text{Å} R_c,$$

$$M_3 = h(R_c).$$

5) U_A sends message $< ID_A, M_2, M_3 >$ to S.

2.1.3 Authentication protocol

When receiving the login request message from U_A:

1) S Checks the validity of ID_A first. If the format is correct, S computes $M_4 = h(ID_A \| x)$, $M_5 = M_2 \oplus M_4 = R_c$. Then it compares $h(M_5)$ to M_3. If they are not equal S will refuse the user login request; otherwise, S computes $M_6 = M_4 \oplus R_s$, $M_7 = h(M_2 \| M_5)$ and $M_8 = h(R_s)$. To avoid replay attacks and MITM attacks, S will save ID_A and M_5 in its database. When S receives the next message from U_A, it will compute $M_5^{'} = M_2^{'} \oplus M_4^{'}$ and determine whether M_5 and $M_5^{'}$ is equal. If it is, the received message is defined as replay message and it will be dropped by sever simply. Otherwise it is defined as fresh message and M_5 in database will be replaced by $M_5^{'}$.

2) S sends message $< M_7, M_6, M_8 >$ to U_A.

3) When receiving above messages, U_A checks whether M_7 is equal to $h(M_2 \| R_c)$. If it is not, U_A terminates the session. Otherwise U_A computes $M_9 = M_6 \oplus M_1$ and checks whether $h(M_9)$ is equal to M_8. If it is not, U_A terminates the session; otherwise it computes $M_{10} = h(M_6 \| M_9)$ and sends message $< M_{10} >$ to S.

4) After S receives the response message, it checks whether M_{10} is equal to $h(M_6 \| R_s)$. If it is not, S will refuse the login request of U_A; otherwise, S accepts the request.

2.1.4 Password modification protocol

1) U_A Inserts smart card and inputs BIO_A

2) The smart card checks whether $h(BIO_A)$ is equal to f_A. If U_A passes the biometric check, it continues to input its old password PWD_{Aold} and new one PWD_{Anew}.

3) The smart card computes $r'_A = h(PWD_{Aold}) \oplus f_A$. If $r'_A \neq r_A$ it believes the old password is wrong and the smart card will terminate the modification protocol. Otherwise, the smart card computes the following values:

$r_A" = h(PWD_{Anew}) \oplus f_A$,

$e'_A = e_A \oplus r'_A$,

$e"_A = e'_A \oplus r"_A$.

4) The smart card replaces e_A and r_A with $e"_A$ and $r"_A$.

2.2 EXISTING ATTACKS IN LI-HWANG'S PROTOCOL

2.2.1 Privileged insiders' attack

In Li-Hwang's scheme, directly sends identity and password to registration centre through security channel, so internal managers at the registry centre can monitor this message and acquire the identity and password. If has ever registered at another server with the same identity and password, this internal manager can impersonate to login on these servers and use the resources. Therefore, there exists privilege attack of the insiders.

2.2.2 Database attack and server impersonation attack

The follows will prove that there are database attack and server impersonation attack in Li-Hwang's protocol. The attacking process is:

1) When U_A sends login request message $< ID_A, M_2, M_3 >$ to S, the attacker intercepts this message and forwards the message to S.

2) When receiving the message from the attacker, S computes challenge information $< M_7, M_6, M_8 >$ according to protocol and forwards it to the attacker. Then the attacker computes $M_7 = h((h(ID_A \| x) \oplus R_C \| R_C)$ and terminates the connection with S. S Believes U_A already cancels the login request.

3) Since there is no time limit, the attacker will impersonate S to send response message $< M_7, M_2, M_3 >$ to U_A.

4) U_A checks the validity of the message fed back by impersonated S. It first checks whether $M_7 = h(M_2 \| R_C)$. From above steps we know the equation is tenable. So the message can pass user's validity check. U_A computes $M_9 = M_2 \oplus M_1$ and compares $h(M_9)$ to M_3. They are obviously equal so the message passes the check of U_A, meaning U_A authenticates the identity of impersonated server. Then U_A computes $M_{10} = h(M_2 \| M_9)$ and sends message $< M_{10} >$ to S. The attacker will intercept this message and simply drop it.

After this kind of authentication, U_A believes it is communicating with the right server. But it is the attacker that cheats the user to communicate with U_A actually. Therefore, it proves that there exists Database attack and server impersonation in Li-Hwang's protocol.

2.2.3 Parallel attack

We assume the attacker has recorded one round of the communication message of user and server. The attack process is shown in detail as below:

1) The attacker initializes a session and sends old message $< ID_A, M_2, M_3 >$ to server for login.

2) When S receives the login request, it computes each parameter according to the protocol and sends challenge message $< M_7, M_6, M_8 >$ to U_A. This message is intercepted by the attacker.

3) Though the attacker cannot create response message $M_{10} = h((h(ID_A \| x) \oplus R_s \| R_s)$ at this moment, he may make S create effective message for him. First, he initializes another session and sends login request $< ID_A, M_6, M_8 >$ to server. S Has no mechanism to refuse this message and the message will pass the validity verification successfully. S continues to compute $M_4^{'} = h(ID_A \| x)$, $M_5^{'} = M_6 \oplus M_4^{'}$, $M_7^{'} = h(M_6 \| M_5^{'})$, $M_6^{'} = M_4^{'} \oplus R_s^{'}$, $M_s^{'} = h(R^{'})$ and sends message $< M_7^{'}, M_6^{'}, M_s^{'} >$ to the attacker.

4) The attacker turns back to the previous session and sends response message $< M_7^{'} >$ to S. Because $M_7^{'} = h((h(ID_A \| x) \oplus R_s \| R_s)$, the attacker responses the server correctly. Then the identity of the attacker will pass the authentication of server.

3 Improved user authentication scheme

We propose an ECC-based three-factor authentication protocol to solve the attack problem existing in Li-Hwang's protocol. Our scheme is composed of six sub-protocols: system setting protocol, registration protocol, protocol phase, login protocol, authentication protocol and password modification protocol. The processes of protocol are described as follows:

3.1 SYSTEM SETTING

All the users and servers negotiate with the parameters of ECC system. The server chooses its key x and corresponding public key $P_s = x \times P$. S saves x and announces its system parameters p , a , b , P , n , h , P_s .

3.2 REGISTRATION PROTOCOL

Figure 1 depicts the registration protocol in our scheme. When the user want to login on the server and applies the sources on server, he must register on S . In this protocol the user communicates with server through a safe channel, such as VPN or SSL.

The details are described as follows:

1) U_A freely chooses its identity ID_A and password PWD_A . Then it inputs it's biometric BIO_A into the device. U_A chooses a random number and computes $HPWD_A = h(PWD_A \| N \| BIO_A)$. Finally, it sends registry message $< ID_A, HPWD_A, BIO_A >$ to S .

2) After S receives the registry message, it first checks private information of U_A . Then it computes $V_A = h(ID_A \| x) \oplus HPWD_A$ and saves public parameters V_A , BIO_A , $f(.,.)$, τ , $h(x)$ in one smart card. At last it sends the smart card to U_A through security channel. The server will maintain a table whose content is a record set with the format as $(ID_A, status - bit)$.

3) U_A receives its smart card and inputs N .

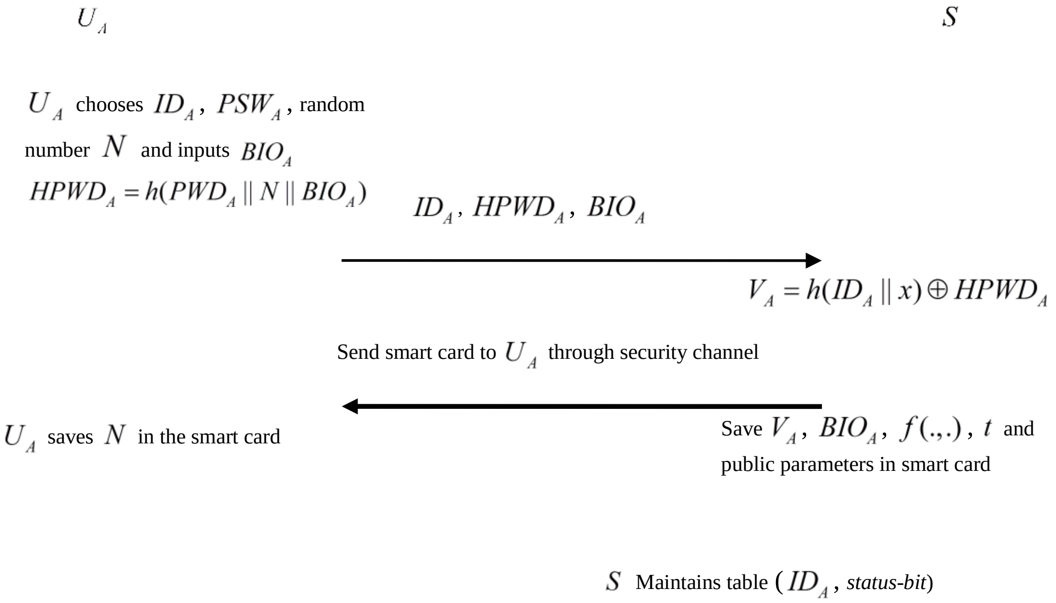
U_A
S
U_A chooses ID_A, PSW_A, random number N and inputs BIO_A
$HPWD_A = h(PWD_A \| N \| BIO_A)$
ID_A, $HPWD_A$, BIO_A
$V_A = h(ID_A \| x) \oplus HPWD_A$
Send smart card to U_A through security channel
U_A saves N in the smart card
Save V_A, BIO_A, $f(.,.)$, t and public parameters in smart card
S Maintains table (ID_A, *status-bit*)

FIGURE 1 Registry protocol

3.3 PRETREATMENT PROTOCOL

When U_A inserts the smart card, the card will choose a random number $r_1 \in Z_n^*$ and compute $R_1 = r_1 \times P$, $R_2 = r_1 \times P_S$. Then it saves R_1 and R_2 in subsequent protocols. When the protocol is executed the smart card will delete R_1 and R_2 .

3.4 LOGIN PROTOCOL

The login protocol is described in Figure 2. In this protocol, the user communicates with server through a common channel. When U_A wants to login S , it will first input it's biometric BIO_A^* .

Then the smart card begins the following process:

1) Smart card judges whether $f(BIO_A, BIO_A^*) < \tau$ holds. If it is not, the smart card terminates the session; otherwise it goes to the next step.

2) The smart card reminds user of inputting his ID_A and PWD_A . Then it computes $s = V_A \oplus h(PWD_A \| N \| BIO_A)$ and $V_1 = h(ID_A \| S \| R_1 \| R_2 \| s)$. At last, the smart card sends message $M_1 =< ID_A, R_1, V_1 >$ to S .

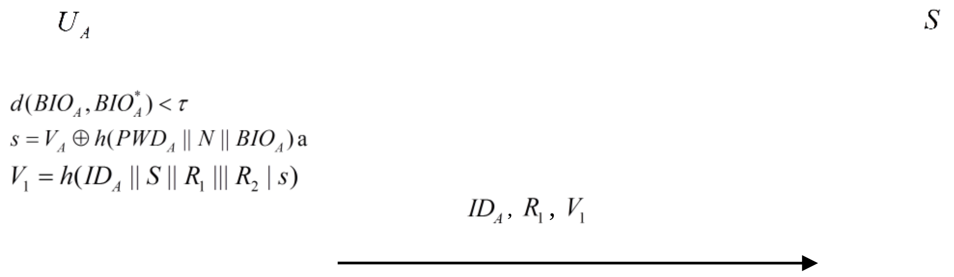
U_A
S
$d(BIO_A, BIO_A^*) < \tau$
$s = V_A \oplus h(PWD_A \| N \| BIO_A)$ a
$V_1 = h(ID_A \| S \| R_1 \||| R_2 | s)$
ID_A, R_1, V_1

FIGURE 2 Login protocol

3.5 AUTHENTICATION PROTOCOL

Figure 3 describes the authentication process. When S receives login message from U_A, it executes the following protocols to check M_1

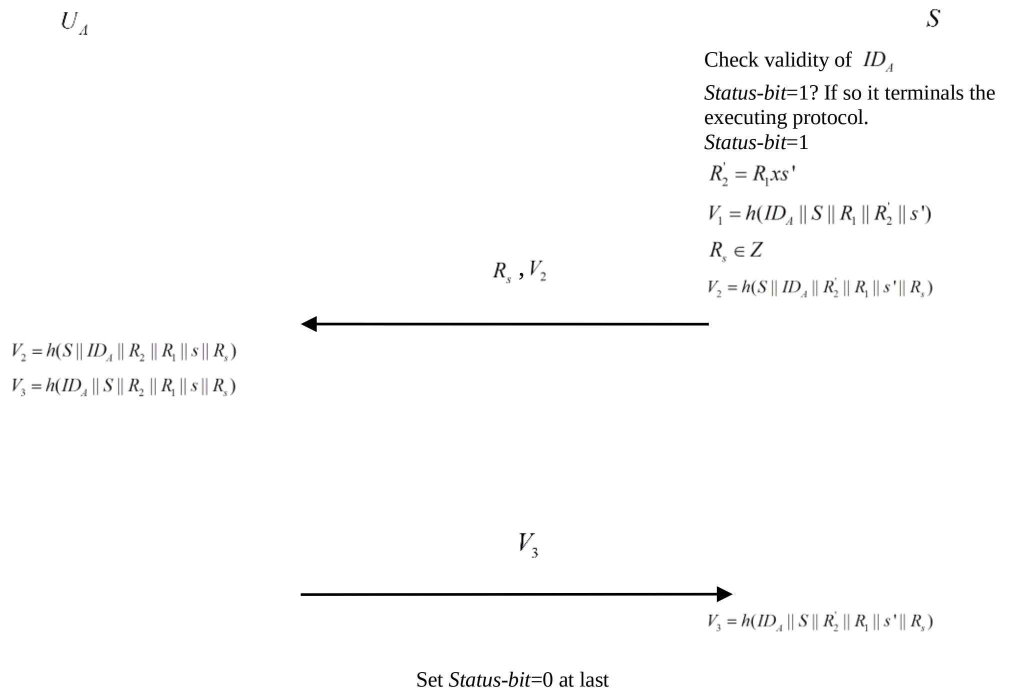
U_A
S
Check validity of ID_A
Status-bit=1? If so it terminals the executing protocol.
Status-bit=1
$R_2^{'} = R_1 x s'$
$V_1 = h(ID_A \| S \| R_1 \| R_2^{'} \| s')$
$R_s \in Z$
R_s, V_2
$V_2 = h(S \| ID_A \| R_2^{'} \| R_1 \| s' \| R_s)$
$V_2 = h(S \| ID_A \| R_2 \| R_1 \| s \| R_s)$
$V_3 = h(ID_A \| S \| R_2 \| R_1 \| s \| R_s)$
V_3
$V_3 = h(ID_A \| S \| R_2^{'} \| R_1 \| s' \| R_s)$
Set *Status-bit*=0 at last

FIGURE 3 Authentication protocol

1) First, S will check the validity of U_A. If it is not in the database, the server terminates the session and notices the user about related events; otherwise S continues to check whether *status-bit of* U_A is 1. If so, S believes there is a session example of U_A and denies the login message; otherwise, it let *status-bit=1* and goes to the next step.

2) S computes $R_2^{'} = x \times R_1 = r_1 \times P_s$ and $s' = h(ID_A \| x)$. Then it checks whether $V_1 = h(ID_A \| S \| R_1 \| R_2 \| s')$. If it is not, S will refuse the login request and notice the user about this event; otherwise, S chooses a random number R_s and computes $V_2 = h(S \| ID_A \| R_2^{'} \| R_1 \| R_s)$. Finally it will send message $M_2 =< R_s, V_2 >$ to U_A.

3) When receiving message M_2, U_A checks whether $V_2 = h(S \| ID_A \| R_2 \| R_1 \| s \| R_s)$. If it is not, it terminates the executing protocol; otherwise U_A authenticates the identity of S. It computes $V_3 = h(ID_A \| S \| R_1 \| R_2 \| s \| R_s)$ and sends message $M_3 =< V_3 >$ to S.

4) When M_3 is received, S will check whether $V_3 = h(ID_A \| S \| R_2^{'} \| R_1 \| s' \| R_s)$. If it is not, S terminates the protocol. Otherwise S authenticates the identity of the user and accepts his login request. After the session is finished, it sets *status-bit=0.*

3.6 PASSWORD MODIFICATION PROTOCOL

When U_A doubts its password is stolen, it can execute the password modification protocol depicted in Figure 4, to change its password.

The process in detail:

1) U_A inputs BIO_A^* into the device. The smart card checks whether $f(BIO_A, BIO_A^*) < \tau$ holds. If it is not, the smart card denies this modification and goes to the next step.

2) The smart card reminds user of inputting his identity ID_A, old password PWD_A and new password PWD_{Anew}. Then it executes the protocol of login and authentication. When success message of server authentication is acquired, the smart card goes to the next step; otherwise, it denies this modification and returns failure message to the user.

3) The smart card chooses a random number N_{new} and computes

$$V_A^{'} = V_A \oplus h(PWD_A \| N \| BIO_A) \oplus h(PWD_{Anew} \| N_{new} \| BIO_A).$$

Then V_A and N are replaced by $V_A^{'}$ and N_{new}. The success message of modification is also sent to the user.

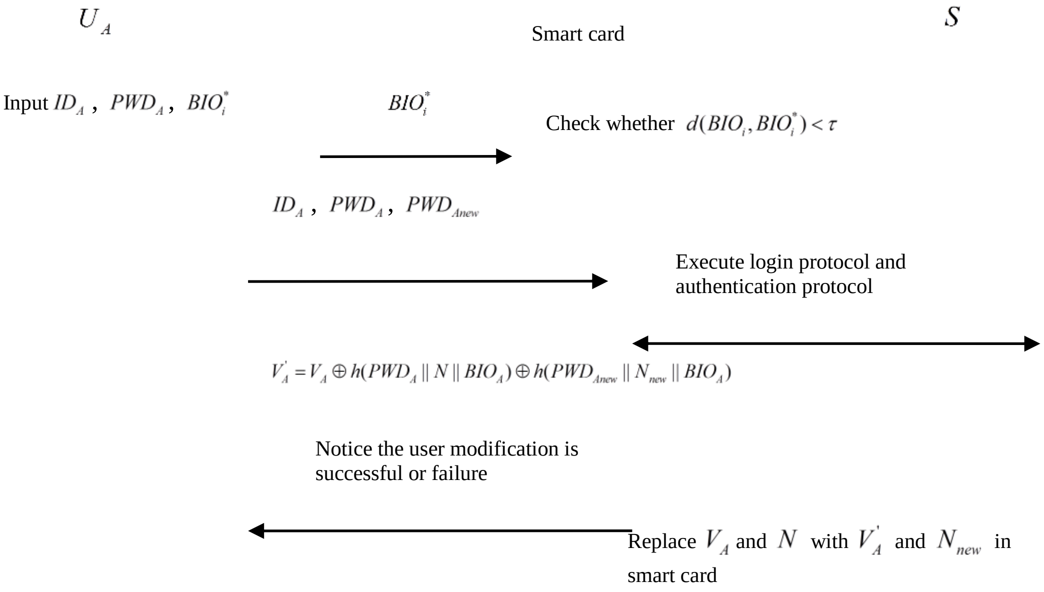
U_A
Smart card
S
Input ID_A, PWD_A, BIO_i^*
BIO_i^*
Check whether $d(BIO_i, BIO_i^*) < \tau$
ID_A, PWD_A, PWD_{Anew}
Execute login protocol and authentication protocol
$V_A^{'} = V_A \oplus h(PWD_A \| N \| BIO_A) \oplus h(PWD_{Anew} \| N_{new} \| BIO_A)$
Notice the user modification is successful or failure
Replace V_A and N with $V_A^{'}$ and N_{new} in smart card

FIGURE 4 Password modification protocol

4 Performance analysis and comparisons

4.1 SECURITY ANALYSIS

This sector analyzes the security performance of our multi-factor authentication protocol. The following propositions are adopted to prove the security performance.

Proposition 1: Improved protocol can resist privileged attack.

Proof: Privileged insiders at server cannot compute users' password PWD_A from message $h(PWD_A \| N \| BIO_A)$. Because $h(PWD \| N \| BIO_A)$ has one unknown high entropy N and the Hash function is irreversible. So this protocol can resist privileged insiders' attack.

Proposition 2: Improved protocol can resist smart card losing attack.

Proof: When the attacker gets the smart card of user, he may acquire BIO_A, N and V_A from it. Then he tries password guessing attack and he can perform the following attack:

1) The attacker selects a candidate password $PWD*$

from the dictionary set;

2) The attacker computes

$$s* = V_A \oplus h(PWD* \| N \| BIO_A)\ ;$$

3) The attacker must verify the correctness of $s*$ in recorded messages. In this protocol he has only one way, that is, checking V_1, V_2 and V_3. However, he ought to compute R_2 or R_2' correctly from R_1. Then he must get x of S or he can extract r_1 from R_1. Neither of these cases will occur, since the private key of server is confidential, and there is an ECDLP problem when extracting r_1 from R_1.

So it is believed that the new protocol can resist the smart card losing attack.

Proposition 3: Improved protocol can resist impersonation attack.

Proof: Attacker Eve cannot impersonate users and servers for communication.

First, it cannot be verified by biometrics. Secondly, it cannot correctly create message V_1 and V_3 without PWD_A or x. If he replays the old message of U_A, he cannot create the third response message V_3 correctly, so that the attacker cannot impersonate the user to obtain corresponding trust from servers.

Attacker cannot impersonate server to cheat the users. Without PWD_A or private key x, the attacker has no way to correctly compute R_2' from R_1 so he cannot impersonate the server. From above descriptions, it can be known that this protocol can resist impersonation attack.

Proposition 4: Improved protocol can resist database attack.

Proof: This protocol does not offer the attacker any chance to enter the database, so it can resist database attack successfully.

Proposition 5: Improved protocol can resist parallel attack.

Proof: This protocol ensures that one user can only start one session applies by *status-bit*. Then the attacker cannot launch parallel attack with multi-conversation.

Proposition 6: Improved protocol can resist DoS attack.

Proof: First, this protocol can modify local password instead of that of remote server, so password correcting will not cause DoS attack. Second, this protocol does not use Hash function on biometrics so there is not DoS problem in Li-Hwang's protocol. Next, this protocol adopts *status-bit* to resist multi-conversation and the attacker cannot use multi-session to cause DoS attack. Thus, this protocol can resist DoS attack.

Proposition 7: Improved protocol can provide bidirectional authentication.

Proof: In this protocol, the user will authenticate the server identity by challenging and verifying the accuracy of V_2, while the server challenges the user by M_2. If the user can correctly answer message M_3, the server will authenticate users' identity. Thus, this protocol can provide bidirectional authentication.

4.2 PERFORMANCE COMPARISON

The calculation cost in login protocol and authentication protocol are discussed. Since login protocol and authentication protocol are the key parts, they are the most frequently executed protocols during operations. XOR operation can be ignored in comparison with point multiplication, while the most time-consuming computation in protocol is point multiplication. Table 2 shows the needed computation cost and communication cost of login protocol and authentication protocol.

We respectively use p and h to denote elliptical curve point multiplication and Hash function calculation. The computation cost of this protocol is one time of point multiplication and 7 times of Hash calculations, while the needed communication cost includes transferring one identity message, 3 hash values, 1 fresh value and one elliptical curve point value. We suppose that the length of all identity strings, Hash value and fresh value is l_1, The elliptical curve point value is supposed to be l_2 so that the total communication cost is $5l_1 + l_2$. It is supposed that the length of identity ID, Hash value and *nonce* are the same as l_1, and l_2 denotes the point length of additive group G.

TABLE 2 Computation and communication cost of login protocol and authentication protocol

Cost	Login protocol and authentication protocol
Computation	$1p + 7h$
Communication	$5l_1 + l_2$

The security performance of this protocol and relative protocols, and the computation cost comparisons of login and authentication protocol are shown in Table 3. h denotes one-way Hash function computation, e denotes modular exponent computation and p denotes elliptical curve scalar computation, that is, point multiplication computation.

Attack$_1$ denotes privileged attack and Attack$_2$ denotes smart card loss attack; Attack$_3$ denotes password guessing attack and Attack$_4$ denotes DoS attack; Attack$_5$ denotes calibration stealing attack and Attack$_6$ denotes database attack; Attack$_7$ denotes impersonation server attack and Attack$_8$ denotes parallel attack. Attack$_9$ denotes impersonation attack and Attack$_{10}$ denotes replay attack. F$_1$ denotes whether bidirectional authentication is supported while F$_2$ denotes whether password correction function is supported. F$_3$ denotes whether clock synchronization is needed or not while F$_4$ denotes whether non-repudiation can be offered or not. C denotes the computation cost of login and authentication protocol.

For efficiency, the modular exponent computation is the most time-consuming computation of all. Since Yoo's and Cao's protocol use modular exponent computation, their computation costs are very expensive with low efficiency. Because Yoon, Chang, Khan, Li [15] and Li-Hwang’s protocols are only based on Hash function and XOR operation, their computation costs are low with high efficiency. While the protocol in this paper uses elliptical curve point multiplication computation, so the computation cost of our protocol is moderate and its efficiency is lower than Li-Hwang’s protocol. However, from the analysis on security vulnerability of Li-Hwang’s protocol, we know that they all have different defects with the same attack, compared to other relevant authentication protocols. In particular, relevant protocols are all suffering from smart card loss attack and off-line password guessing attack. Although this protocol has relatively large computation cost, it can resist various attacks mentioned above.

TABLE 3 Security attributes and comparisons of computation cost

	TAN	Cao	Tsai	Yoo	Khan	Li	Li-Hwang	Ours
Attack_1	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Safe
Attack_2	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Safe
Attack_3	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Unsafe	Safe
Attack_4	Safe	Safe	Unsafe	Safe	Unsafe	Unsafe	Unsafe	Safe
Attack_5	Safe	Safe	Safe	Safe	Safe	Safe	Unsafe	Safe
Attack_6	Safe	Safe	Safe	Safe	Safe	Unsafe	Unsafe	Safe
Attack_7	Unsafe	Unsafe	Safe	Safe	Safe	Unsafe	Unsafe	Safe
Attack_8	Safe	Safe	Safe	Safe	Safe	Unsafe	Unsafe	Safe
Attack_9	Unsafe	Unsafe	Safe	Safe	Unsafe	Unsafe	Unsafe	Safe
Attack_{10}	Unsafe	Unsafe	Unsafe	Safe	Unsafe	Unsafe	Unsafe	Safe
F_1	Not available	Not available	Available	Available	Available	Available	Not available	Available
F_2	Available	Available	Not available	Available	Available	Available	Available	Available
F_3	Not available	Not available	Not available	Available	Not available	Available	Available	Available
F_4	Available	Not available	Not available	Not available	Available	Available	Not available	Available
C	$3h+4e$	$4h+1e$	$5h$	$8h$	$7h$	$7h$	$10h$	$1p+7h$

5 Conclusion

This paper focuses on the key agreement protocol for multi-factor authentication. By the security study on Li-Hwang's three-factor authentication scheme, we point that there are privileged attack, calibration stealing attack, database attack, server impersonation attack, parallel attack, user impersonation attack and DoS attack in this protocol. Then we propose a new three-factor authentication protocol based on ECC. The new protocol only needs to compute point multiplication of elliptical curve once, keeping high efficiency during the process of login and authentication. It is also proved that our protocol can resist various attacks of password-based authentication protocol, including smart card loss attack. Therefore, it is suited for multi-factor authentication protocol combining password, mobile device with biometrics. Although a multi-factor authentication protocol is put forward in this paper, a satisfactory result is not achieved on biometric protection. So the next research object is protecting the biometric which is not abused by attackers.

References

[1] Zhang D 2000 Automated biometrics: technologies and systems *USA Kluwer Academic Publishers*

[2] Jain A K, Ross A, Prabhakar S 2004 An introduction to biometric recognition *IEEE Transaction on Circuits and Systems for Video Technology* **14**(1) 4-20

[3] Matyas V J, Riha Z 2003 Toward reliable user authentication through biometrics *IEEE Security ＆ Privacy* **1**(3) 45-9

[4] Kong S G, Heo J, Abidi B R, et al 2005 Recent advances in visual and infrared face recognition: A review *CVIU* **97** (1) 103-35

[5] Han X, Liu K, Zu X 2006 Design of Three-factor Authentication Project for ThinClient/Server System *Computer Engineering* **32**(24) 126-9

[6] Tan X, Bhanu B 2005 A robust two step approach for fingerprint identification *Pattern Recognition Letters* **24**(13) 2127-34

[7] Cao W, Zhao Y 2014 Research on the Technology of Mobile Payment Security Based on Two-factor Authentication *Information Security and Technology* **24**(2) 10-5

[8] Tsai J L 2009 Efficient nonce-based authentication scheme for session initiation protocol *International Journal of Network Security* **8**(3) 312-6

[9] Tang H, Liu X 2012 Cryptanalysis of Arshad et al.'s ECC-based mutual authentication scheme for session initiation protocol *Multimedia tools and applications* **51**(3) 1-13

[10] Yoo K Y, Yoon E J 2010 A three-factor authenticated key agreement scheme for SIP on elliptic curves *Proceedings of the 2010 Fourth International Conference on Network and System Security* 334-9

[11] Samaneh Sadat Mousavi-Nik, Samaneh Sadat Mousavi-Nik, Samaneh Sadat Mousavi-Nik 2012 Proposed SecureSIP Authentication Scheme based on Elliptic Curve Cryptography *Modeling Decisions for Artificial Intelligence* **58**(8), 25-30

[12] Liu X, Tang H 2013Cryptanalysis of Arshad et al.'s ECC-based mutual authentication scheme for session initiation protocol *Computer Application* **65**(3) 321-33

[13] Chun-Ta Li, Min-Shiang Hwang 2010 Corresponding author contact information, An efficient biometrics-based remote user authentication scheme using smart cards *Journal of Network and Computer Applications* **33**(1) 1-5

[14] Khan M K, Zhang J 2007 Improving the security of ‘a flexible biometrics remote user authentication scheme *Computer Standards & Interfaces* **29**(1) 82-5

[15] Li X, Xiong Y, Ma J 2012 An efficient and security dynamic identity based authentication protocol for multi-server architecture using smart cards *Journal of Network and Computer Applications* **35**(2) 763-9

Authors	
	Ying Wang, 10 May, 1981, Shangxi Province, P.R. China. **Current position, grades**: lecturer, department of computer science and technology, Taiyuan University of Technology, China. **University studies**: MSc degree in computer application technology from Taiyuan University of Technology in China. **Scientific interest**: computer network and security, trusted computing and cryptography.
	Xinguang Peng, 03 May, 1955, Shangxi Province, P.R. China. **Current position, grades**: professor, department of computer science and technology, Taiyuan University of Technology, China. **University studies**: PhD degree in computer application technology from the Beijing Institute of Technology in China. **Scientific interest**: computer network and security trusted computing.
	Jing Bian, 09 July, 1983, Shangxi Province, P.R. China. **Current position, grades**: PhD candidate, department of computer science and technology at Taiyuan University of Technology. **University studies**: MSc degree in computer application technology from Taiyuan University of Technology in China. **Scientific interest**: network security and data mining.