A MULTI-FACTOR CERTIFICATELESS AUTHENTICATED KEY AGREEMENT PROTOCOL ON ECC

Jin Tang[1] and Xiaofeng Wang[2]

[1]Cyberspace Security Department, National University of Defense Technology,
Changsha, China

[2]Cyberspace Security Department, National University of Defense Technology,
Changsha, China

ABSTRACT

Key negotiation can establish a shared key between two or even multiple parties in a public network environment, ensuring communication confidentiality and integrity. Certificateless public key cryptography (CL-PKC) aims to achieve succinct public key management without using certificates, while avoiding the key escrow property in identity-based cryptography. As an important part of CL-PKC, certificateless authentication key agreement (CLAKA) has also received widespread attention. Most CLAKA protocols are constructed from bilinear mappings on elliptic curves which need costly operations. To improve the performance, some pairing-free CLAKA protocols have been proposed. In this paper, we propose a multi-factor authentication CLAKA protocol that can achieve local authentication factors joint unlocking. The protocol does not require bilinear pairing computation and has been proven to be secure under the mBR model.

KEYWORDS

Certificateless Public Cryptography, Multi-factor, CLAKA, Provable Security, Non-bilinear, Elliptic Curve Cryptography(ECC)

1. INTRODUCTION

In traditional public key cryptography (PKC), certificates are needed to assure users of the relationship between a public key and its corresponding private key holder's identity. This gives rise to problems in certificate management, including revocation, storage, and distribution. To address these issues, Shamir introduced the concept of identity-based encryption (ID-PKC)[1]. In ID-PKC setup, a user's public key can be derived from their identity (e.g., their name or email address), while their key is generated by a key generation center (KGC). Then comes the issue of key escrow, where the PKG knows all users' secret keys. In 2003, Al-riyami[2]proposed certificateless public key cryptography (CLPKC) to solve the key escrow problem. Since then, CLPKC has received great attention. After the work of Al-riyami[2], many certificateless authentication key agreement (CLAKA) protocols using bilinear mapping on elliptic curves have been proposed, such as [3-6]. However, the relative computational cost of pairings is about 3 times higher than scalar multiplication on elliptic curve groups . Therefore, pairing-free CLAKA protocols are more attractive in terms of efficiency.

David C. Wyld et al. (Eds): NLPML, AIAP, SIGL, CRIS, COSIT, DMA -2023
pp. 189-197, 2023. CS & IT - CSCP 2023 DOI: 10.5121/csit.2023.130816

He[7] and Hu[8] respectively proposed non-pairing CLAKA protocols, but Sun[9] showed that ,both schemes [7,8] are insecure. In both protocols, an adversary who obtains the temporary key can calculate the session key. Kim[10] constructed a non-pairing CLAKA protocol and claimed that it is secure in the eCK model. However, Bala[11] pointed out that the protocol is vulnerable to key compromise impersonation attacks. In recent years, some CLAKA protocols[12] still have similar vulnerabilities.

Due to various problems with single and double-factor identity-based authenticated key exchange, multi-factor identity-based authenticated key exchange has gradually become a research hotspot. Compared with traditional authentication factors, biometrics have advantages such as uniqueness, difficulty in guessing or cloning, difficulty in loss or forgery,which can provide more reliable protection for identity authentication. In 2013, Yoon[13] proposed the first biometric-based multi-server environment user identity authentication scheme. He[14] pointed out that Yoon's scheme is weak in its ability to resist simulation attacks and internal privilege attacks. Chuang[15] proposed an identity authentication scheme based on smart cards and biometrics, which solved the problems of weak user anonymity and inability to resist instant message attacks in their scheme, but the overall scheme was relatively time-consuming.Chatterjee[16] used Chebyshev mixed mapping to calculate a new biometric-based identity authentication protocol, which has the advantages of small keys, fast computation speed, and high efficiency.

However, these existing multi-factor key exchange schemes are only used locally to unlock keys distributed by servers or trusted third parties. Essentially, whoever owns the distributed key can impersonate the key user for authentication and cannot achieve true multi-factor authentication, most of which cannot resist internal privilege attacks. There are currently only two certificateless key exchange schemes[17,18]using multi-factor identity authentication technology, and essentially both of them generate an authentication key by combining multiple authentication factors, and they cannot achieve multi-factor joint unlocking.

This article discusses how to ensure secure identity authentication and key negotiation between different communication entities, proposing a multi-factor certificateless key negotiation scheme. It avoids the complex certificate management mechanism in PKI and the limitations of key custodianship in IBC. The communication entity can directly authenticate the identity remotely, and choose different identity authentication factors for combination, which is more autonomous and flexible. The proposed scheme has low computational overhead without bilinear pairing operations and has been proven to be secure under the mBR model.

The remainder of our paper is organized as follows. Section 2 presents the basic concept of ECC and certificateless public key cryptography.Section 3 presents our multi-factor CLAKA protocol. Section 4 gives security proof under the mBR model. Finally, Section 5 concludes the paper.

2. PRELIMINARIES

2.1. Background of elliptic curve group

In this section, we introduce some basic knowledge. The list of the key notations used in this article is represented in Table 1.

Table 1. Description of notations.

Notation	Description
$\in$	the operation to randomly choose from a set
(xp,yP)	the x-coordinate and y-coordinate of a point P respectively
G	A cyclic group of order q
P	The generator of group G
ID	Unique identification
KGC	The key generation center of domain

Notation	Description
$\in$	the operation to randomly choose from a set
U	The entity of user
S	The entity of server

The symbol E/F_q denote an elliptic curve E over a prime finite field Fq, defined by an equation $Y^2=x^3+ax+b$,$a,b\in F_q$,along with an imaginary point representing the infinity. An additive group G_q of all points on elliptic curve E includes an addition operation.
Let P be a generator of G_q. Let the order of G_q be an integer n. Let $Z_n^*=[1,n-1]$.The following problems are commonly used in the security analysis of many cryptographic protocols.
Computational Diffie-Hellman (CDH) problem: Given a generator P of G and (aP,bP) for unknown $a,b\in_R Z_n^*$, the task of CDH problem is to compute abP .
The CDH assumption and states that the probability of any polynomial-time algorithm to solve the CDH problem is negligible.
Divisible computational Diffie-Hellman (DCDH) problem: Given a generator P of G,(aP, bP) for unknown $a,b\in_R Z_n^*$, the task of DCDH problem is to compute $a^{-1}bP$.
The DCDH assumption and states that the probability of any polynomial-time algorithm to solve the DCDH problem is negligible.

2.2. Security of CLAKA protocol

There are two types of adversaries against a CLAKA protocol.
AI:The type I adversary AI can query the user's secret value and even replace the user's public key, but cannot obtain the user's partial private key.
(1) AII:The Type II adversary AII can obtain the master secret key of the system and query the user's private key, but cannot obtain the user's secret value and cannot replace the user's public key[6]. The ability of adversary $A\in\{AI, AII\}$ is simulated by some queries with the challenger C.
Let $\prod_{i,j}^{S}$ represents the s^{th} session between the user ID_i and the user ID_j.A CLAKA protocol is said to be secure if:
(1) In the presence of a benign adversary on $\prod_{i,j}^{n}$ and $\prod_{j,i}^{t}$,both oracles always agree on the same session key, and this key is distributed uniformly at random.
(2) For any adversary, $Advantage^A(k)$ is negligible.

3. OUR PROTOCOL

3.1. Initializing Phase

This algorithm takes a security parameter l as an input, returns system parameters and a master key. Given k , KGC does the following steps.
(1) KGC selects an elliptic curve $E:Y^3=X^2+aX+b$ defined over a prime field F_p. The curve has a cyclic point group G of prime order q.Pick a generator $P\in G$.
(2) KGC chooses the master private key $s\in_R Z_q^*$ and computes the master public key $P_{PUB}=sP$.
(3) KGC chooses two cryptographic hash functions: $H_1:\{0,1\}^*\rightarrow\{0,1\}^n$; $H_2:\{0,1\}^*\rightarrow Z_q^*$ for some integer n > 0.
(4) KGC publishes params=$\{a,b,p,q,P,P_{KGC},H_1,H_2\}$ as system parameters and secretly keeps the master key s.

3.2. The User Registration Phase

3.2.1. Set-secret-value

U_i denotes the user with identity ID_i, U_i picks multiple identity authentication factors $c_1... c_n$ as his secret values.Our scheme picks PIN codes and biometric fingerprints. The scheme supports the use of multiple devices to jointly complete identity authentication. Here, we will use mobile devices 1,2,and 3 as examples for explanation. Enter the PIN code c_1 on mobile device 1, enter the fingerprint Bio_i, and generate $(c_2, \varepsilon_i)=Gen(Bio_i)$.
Calculate $C_1=[c_1]P$,$C_2=[c_2]P$, and submit the registration information$<C_1,C_2,ID_i>$to the key generation center KGC.

3.2.2. Partial-private-key-extract

On receiving $<C_1,C_2,ID_i>$, KGC checks the validity of ID_i ,chooses at random $r_i \in_R Z_q^*$, computes $R_i =r_iP$, $Z_i=H_1(a||b||x_P||y_P||x_{Ppub}||y_{Ppub}||ID_i)$,$W_i=C_1+C_2+R_i=(x_{Wi},y_{Wi})$.
KGC computes $\lambda_i=H_1(x_{wi}||y_{wi}||Z_i)$,$c_0=(r_i+\lambda_i \cdot s) \bmod q$, sends to U_i the information containing (W_i,R_i,c_0).

3.2.3. Set-public-key

Let $PK_i=W_i$, W_i is kept in public and considered as the public key(PK_i) of the user with identity ID_i.

3.2.4. Verify-Key

The user with identity ID_i computes $Z_i=H_1(a||b|| x_P||y_P||x_{Ppub}||y_{Ppub}||ID_i)$, $\lambda_i=H_1(x_{wi}||y_{wi}||Z_i)$ and confirms the equation is true if $c_0P=R_i+[\lambda_i]P_{Pub}$ holds. The private key is valid if the equation holds and vice versa, U_i stores the value of $<ID_i,Z_i,\lambda_i,PK_i,c_0>$ in mobile device 1.

3.3. The Server Registration Phase

3.3.1. Set-secret-value

The server with identity ID_j, picks randomly $c_j \in_R Z_q^*$ sets c_j as his secret value,calculate $C_j=[c_j] P$, and submit the registration information$<C_j, ID_j>$to KGC.

3.3.2. Partial-private-key-extract

KGC checks the validity of ID_j and chooses at random $r_j \in_R Z_q^*$,computes $R_j =r_jP$,$Z_j=H_1(a||b||x_P||y_P||x_{Ppub}||y_{Ppub}||ID_j)$ and $W_j=C_j+R_j=(x_{Wj},y_{Wj})$
KGC computes $\lambda_j=H_1(x_{Wj}||y_{Wj}||Z_j)$,$t_j=(r_j+\lambda_j \cdot s) \bmod q$,then return $<W_j,R_j,t_j>$to the server S_j.

3.3.3. Set-public-key

Let $PK_j=W_j$, W_j is kept in public and considered as the public key(PK_j) of the server with identity ID_j.

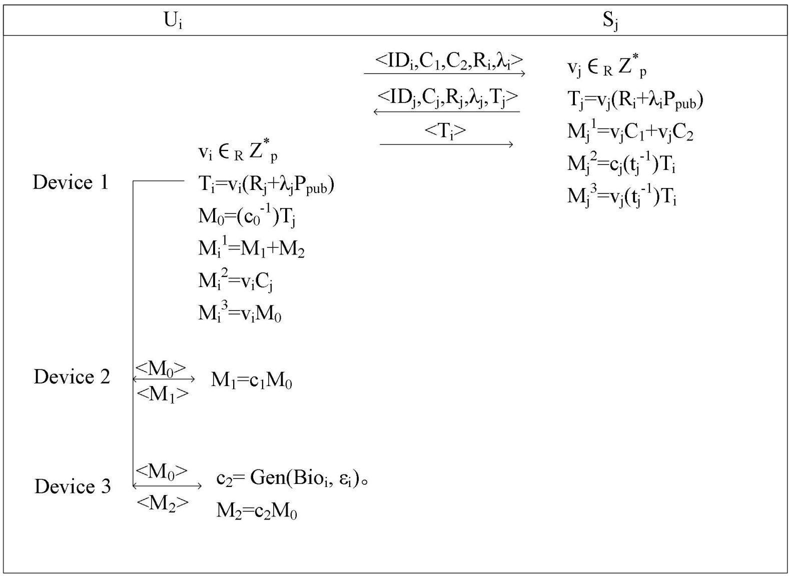
3.3.4. Verify-Key

The server computes Z_j=H_1($a||b||x_P||y_P||x_{Ppub}||y_{Ppub}||ID_j$),$\lambda_j$=$H_1$($x_{Wj}||y_{Wj}||Z_j$) and confirms the equation is true if c_jP= R_j+[λ_j]P_{Pub} holds. The private key is valid if the equation holds and vice versa,S_j stores the value of <ID_j, Z_j,λ_j,PK_j,r_j,t_j >.

3.4. Key Agreement Phase

Assume that the user U_i and the server S_j want to establish a session key, they can run the protocol as shown in Table 2.

Table 2. KEY AGREEMENT.


U_i
S_j
<ID_i,C_1,C_2,R_i,λ_i>
<ID_j,C_j,R_j,λ_j,T_j>
<T_i>
$v_j \in_R Z^*_p$
T_j=v_j(R_i+$\lambda_i P_{pub}$)
M_j^1=v_jC_1+v_jC_2
M_j^2=$c_j(t_j^{-1})T_i$
M_j^3=$v_j(t_j^{-1})T_i$
Device 1
$v_i \in_R Z^*_p$
T_i=v_i(R_j+$\lambda_j P_{pub}$)
M_0=(c_0^{-1})T_j
M_i^1=M_1+M_2
M_i^2=v_iC_j
M_i^3=v_iM_0
Device 2
<M_0>
<M_1>
M_1=c_1M_0
Device 3
<M_0>
<M_2>
c_2= Gen(Bio_i, ε_i)。
M_2=c_2M_0

U_i offers his ID_i in mobile device 1, sends <ID_i,C_1,C_2,R_i, λ_i > to the server S_j.

After receiving the message, the server S_j chooses at random the ephemeral key $v_j \in_R Z_q^*$, and computes T_j = v_j·(R_i+$\lambda_i P_{pub}$), then S_j sends < ID_j,C_j,R_j, λ_j, T_j > to U_i.

After receiving the message, U_i chooses at random the ephemeral key $v_i \in_R Z_q^*$ and computes T_i=v_i(R_j+$\lambda_j P_{pub}$), then A sends <T_i> to S_j. U_i computes M_0=(c_0^{-1})T_j , Send M_0 to mobile devices 2 and 3.

After receiving M_0, offers his ID_i and PIN code c_1 on mobile device 2. Calculate M_1=c_1M_0 and send M_1 to mobile device 1.

After receiving M_0, offers his identity ID_i and biometric information Bio_i^* on mobile device 3, using the helper string ε_i and the fuzzy extractor to obtain biometric feature extraction information c_2=Gen (Bio_i^*, ε_i)。 Calculate M_2=c_2M_0 and then send M_2 to mobile device 1.

After receiving M_1 and M_2, U_i calculate M_i^1=M_1+M_2, M_i^2=v_iC_j and M_i^3=v_iM_0 on mobile device 1.

S_j computes M_j^1=v_jC_1+v_jC_2,M_j^2=cj(t_j^{-1})T_i,M_j^3=v_j(t_j^{-1})T_i.

Thus, the agreed session keys for U_i and S_j can be computed as:

$$SK_{ij} = H_2(ID_i||ID_j||(C_1+C_2)||C_j||R_i||R_j||T_i||T_j||M_i^1||M_i^2||M_i^3)$$
$$= H_2(ID_i||ID_j||(C_1+C_2)||C_j||R_i||R_j||T_i||T_j||M_j^1||M_j^2||M_j^3)$$

4. SECURITY ANALYSIS

To prove the security of our protocol in the random oracle model, we treat H_1 and H_2 as two random oracles. For the security, the following lemmas and theorems are provided.

4.1. Lemma 1

If two oracles are matching, both of them will be accepted and will get the same session key which is distributed uniformly at random in the session key sample space.

Proof

Assuming I and J are session participants, A is a benign adversary,I and J send messages to each other in full accordance with the protocol rules. we know if two oracles are matching, then both of them are accepted and have the same session key.Based on the random oracle machine H_2, The session key are distributed uniformly since v_i and v_j in the protocol are randomly selected and M^1, M^2, M^3 are randomly generated.

4.2. Lemma 2

Assuming that the DCDH problem is intractable, the advantage of a type I adversary against our protocol is negligible.

Proof

Suppose that there is a type I adversary AI who can win the game defined in subsection 2.2 with a non-negligible advantage $Adv_{AI}^{DCDH}(l)$ in polynomial-time t. Then, we will show how to use the ability of AI to construct an algorithm C to solve the DCDH problem.

Suppose C is given an instance (a P,bP) of the DCDH problem, and is tasked to compute cP with $c=a^{-1}b \bmod n$. Supposed q_s be the maximal number of sessions each participant may be involved in,AI makes at most q_{H_i} times H_i queries and creates at most q_c participants. Before the game starts, C randomly selects two indexes I,J ∈ $\{1,\ldots q_c\}$, which represent the I th and the J th distinct honest party that the adversary initially chooses. Also, C chooses $T \in \{1,\ldots q_s\}$ and determines the Test session $\prod_{I,J}^{T}$, which is select with probability larger than $\frac{1}{q_s(q_c{}^2)}$. Let $\prod_{I,J}^{S}$ be the matching session of $\prod_{I,J}^{T}$. Then the challenger C can win the game with $\frac{1}{q_{H_2}q_s(q_c{}^2)}\mathrm{Adv}_{\mathrm{AI}}^{\mathrm{DCDH}}(l)$ in solving the DCDH problem given the security parameter l.

C Select an elliptic curve E: $y^3=x^2+ax+b$, which is defined in a prime field F_p and select (G_1,G_2,e),where G_1 is an additive group of order q, whose generator is P,G_2 is a multiplicative group of order q,bilinear pair $e:G_1\times G_1 \longrightarrow G_2$.C chooses the system parameter $\{a,b,p,q,P,P_{Pub},H_1,H_2,e\}$ and sends params to AI.Let t_JP= aP, chooses at random $r_J,\lambda_J \in_R Z_q^*$, , then C computes $R_J=r_JP$,$P_{Pub}=(t_JP -R_J)\lambda_J{}^{-1}$, C answers AI's queries as follows:

$H_1(ID_i,R_i)$: C maintains an initially empty list L_{H1} consisting of tuples of the form (ID_i,R_i,C_i,λ_i). If (ID_i,R_i,C_i) is on the list L_{H1},then it returns λ_i. Otherwise, C chooses at random $\lambda_i \in_R Z_q^*$ and stores in L_{H1}.

Create(ID_i): C maintains an initially empty list L_C consisting of tuples of the form (ID_i,c_i,t_i,C_i,R_i). On receiving this query, C answers as follows:

- If $ID_i \neq ID_J$, C chooses three random numbers $c_i,t_i,\lambda_i \in_R Z_q^*$, computes $C_i=c_iP$,$R_i= t_iP-\lambda_iPpub$ and stores $(ID_i,R_i,C_i, \lambda_i)$ and (ID_i,c_i,t_i,C_i,R_i) in L_{H1} and Lc separately.
- Otherwise, C chooses at random $c_J \in_R Z_q^*$, computes $C_i=c_iP$,$R_i=t_iP-\lambda_iPpub$ and stores $(ID_i,R_i,C_i, \lambda_i)$ and (ID_i,c_i,t_i,C_i,R_i) in L_{H1} and Lc separately.

Public-Key(ID_i): On receiving this query, C first searches for a tuple in L_C which is indexed by ID_i , then returns C_i as the answer.

Partial-Private-Key(ID_i): Whenever C receives this query, if $ID_i=ID_J$,C aborts. Otherwise,C searches for a tuple in L_C which is indexed by ID_i and returns t_i to AI as the answer.

Corrupt(ID_i): On receiving this query, if $ID_i=ID_J$,C aborts; else, C searches for a tuple in L_C which is indexed by ID_i and if $c_i=\perp$,returns null. Otherwise,C returns (c_i,t_i) to AI.

Public-Key-Replacement(ID_i,C_i'): if ID_i=ID_J,C aborts.Otherwise C searches for a tuple in L_C which is indexed by ID_i then updates C_i to C_i'.

Send($\prod_{I,j}^{n}$, M) : C maintains an initially empty list L_S consisting of tuples of the form($\prod_{i,j}^{n}$, $v_{i,j}^{n}$,$T_{i,j}^{n}$,$T_{j,i}^{n}$,C_i^n,R_i^n,C_j^n, R_j^n,$SK_{i,j}^{n}$),C answers the query as follows:

- If M=ω,C returns (IDi,C_i^n,R_i^n).
- If M=(IDi,C_i^n,R_i^n), C returns ($T_{i,j}^n$=$v_{i,j}^n(R_j^n+\lambda_j Ppub)$, C_i^n,R_i^n).
- If M=($T_{j,i}^n$,C_i^n,R_i^n), C returns ($T_{i,j}^n$=$v_{i,j}^n(R_j^n+\lambda_j Ppub)$).

If n=T,ID_i=ID_I, ID_j=ID_J,then let $v_{I,J}^T = \perp$, $T_{I,J}^T = b$ P,SK= $\perp$,and stores ($\prod_{I,J}^T$, $\perp$,bP,$T_{J,I}^T$,C_I^N,R_I^N,C_j^N, R_j^N, $\perp$) in L_s.

Otherwise, C chooses at random $v_{i,j}^n \in_R Z_q^*$, computes $T_{i,j}^n$=$v_{i,j}^n(R_j^n+\lambda j Ppub)$,let SK=$\perp$ and stores ($\prod_{i,j}^n$, $v_{i,j}^n$,$T_{i,j}^n$,$T_{j,i}^n$,C_i^n,R_i^n,C_j^n, R_j^n, $\perp$) in L_s.

Reveal($\prod_{i,j}^n$): On receiving this query, C searches for a tuple ($\prod_{i,j}^n$, $v_{i,j}^n$,$T_{i,j}^n$,$T_{j,i}^n$,C_i^n,R_i^n,C_j^n, R_j^n, $SK_{i,j}^n$) in L_s which is indexed by $\prod_{i,j}^n$,if $SK_{i,j}^n \neq \perp$,returns $SK_{i,j}^n$. Otherwise,C answers the query as follows:

- If $\prod_{i,j}^n = \prod_{I,J}^T$ or $\prod_{i,j}^n = \prod_{I,J}^S$, C aborts.
- Otherwise,C looks up the list L_{H2} for corresponding tuple (ID_i,ID_j,$T_{i,j}$,$T_{j,i}$,C_i,R_i,C_j,R_j,M^1,M^2,M^3,λ) if $\prod_{i,j}^n$ is the initiator oracle.
- C looks up the list L_{H2} for corresponding tuple (ID_j, ID_i,$T_{j,i}$,$T_{i,j}$,C_j,R_j,C_i,R_i,M^1,M^2,M^3,λ) if $\prod_{i,j}^n$ is the initiator oracle.
- If there are corresponding tuples in list L_{H2}, determine whether the following equation holds:
  $T_{i,j}^n$=$T_{i,j}$, T_{ij}^n=$T_{i,j}$,C_i^n=C_i, R_i^n=R_i, C_j^n=C_j, R_j^n=R_j, e(M^1,R_i+$\lambda_i P_{pub}$)=e($\sum$ Ci, $T_{j,i}$), M^2=$v_{i,j}^n C_j$, e(M^3,R_i+$\lambda_i P_{pub}$)=e($v_{i,j}^n$P, $T_{j,i}$),let $SK_{i,j}^n$=λ and return $SK_{i,j}^n$ if the equation holds,updates $SK_{i,j}^n$ in list Ls.
  Otherwise,C chooses at random $SK_{i,j}^n \in \{0,1\}^l$ and returns $SK_{i,j}^n$ as the reply. Then C updates the tuple indexed by $\prod_{i,j}^n$ in L_s.
- If the corresponding tuple does not exist in list L_{H2}, C chooses at random $SK_{i,j}^n \in \{0,1\}^l$ and returns $SK_{i,j}^n$ as the reply. Then C updates the tuple indexed by $\prod_{i,j}^n$ in L_s.

H_2 query: C maintains a list L_{H2} of the form (ID_i,ID_j,$T_{i,j}$,$T_{j,i}$,C_i,R_i,C_j,R_j,M^1,M^2,M^3,λ) . On receiving this query, C searches for a tuple in L_{H2} which is indexed by (ID_i,ID_j,$T_{i,j}$,$T_{j,i}$,C_i,R_i,C_j,R_j,M^1,M^2,M^3).If a tuple is already in L_{H2}, C replies with the corresponding λ.Else, C searches for a tuple ($\prod_{i,j}^n$, $v_{i,j}^n$,$T_{i,j}^n$,$T_{j,i}^n$,C_i^n,R_i^n,C_j^n, R_j^n,$K_{i,j}^n$)in L_s.

- If the corresponding tuple does exist in list L_s,and the following equation holds:
  $T_{i,j}^n$=$T_{i,j}$,$T_{j,i}^n$=$T_{j,i}$,C_i^n=Ci, R_i^n=R_i, C_j^n=Cj, R_j^n=R_j,e(M^1,R_i+$\lambda_i P_{pub}$)=e($\sum$ Ci, $T_{j,i}$), M^2=$v_{i,j}^n C_j$, e(M^3,R_i+$\lambda_i P_{pub}$)=e($v_{i,j}^n$P, $T_{j,i}$), $SK_{i,j}^n \neq \perp$,then let λ=$SK_{j,i}^n$,returns λ as the reply, updates the tuple in L_{H2}.
- Otherwise, searches for a tuple
  ($\prod_{j,i}^n$, $v_{j,i}^n$,$T_{j,i}^n$, $T_{i,j}^n$,C_j^n, R_j^n, C_i^n,R_i^n,$SK_{j,i}^n$) in L_s. If the corresponding tuple does exist,and the following equation holds:$T_{i,j}^n$=$T_{i,j}$,$T_{j,i}^n$=$T_{j,i}$,C_i^n=Ci, R_i^n=R_i, C_j^n=Cj, R_j^n=R_j,
  M^1=$\sum$ Ci $v_{i,j}^n$,e(M^2,R_j+$\lambda_j P_{pub}$)=
  e(C_j,$T_{j,i}$), e(M^3,R_j+$\lambda_j P_{pub}$)=e($v_{i,j}^n$P, $T_{j,i}$), $SK_{j,i}^n \neq \perp$,
  then let λ=$SK_{j,i}^n$,returns λ as the reply, updates the tuple in L_{H2}.If the corresponding tuple does not exist, C chooses at random $\lambda \in Z_q^*$,returns λ as the reply, updates the tuple in L_{H2}.

Test($\prod_{i,j}^{n}$): If $\prod_{i,j}^{n} \neq \prod_{i,j}^{T}$, C aborts. Otherwise, C chooses at random $\mu \in \{0,1\}$ as the output result.

In this case, C would not have made Corrupt($\prod_{I,J}^{T}$) or reveal($\prod_{I,J}^{T}$) queries, and so C would not have aborted. The probability that C select $\prod_{I,J}^{T}$ as the test oracle is$\frac{1}{q_s(q_c{}^2)}$. If C can win in such a game, then C must have made the corresponding H_2 query of the form (ID_i,ID_j,$T_{i,j}$,$T_{j,i}$,C_i,R_i,C_j,R_j,M^1,M^2,M^3). If $\prod_{I,J}^{T}$ is the initiator oracle, C can find the corresponding M^3 in the $L_{H2}$with the probability $\frac{1}{q_{H_2}}$,and output $M^3(v_{J,I}^{T}{}^{-1})=v_{J,I}^{T}(t_J{}^{-1})T_{I,J}^{T}(v_{J,I}^{T}{}^{-1})= (t_J{}^{-1})\ T_{I,J}^{T}=a^{-1}bP$ as a solution to the DCDH problem. The advantage of C solving DCDH problem with the advantage $\mathrm{Adv}_{\mathrm{C}}^{\mathrm{DCDH}}(l) \geq \frac{1}{q_{H_2} q_C^2 q_s} \mathrm{Adv}_{\mathrm{AI}}^{\mathrm{DCDH}}(l)$. Then $\mathrm{Adv}_{\mathrm{C}}^{\mathrm{DCDH}}(l)$ is non-negligible since we assume that $\mathrm{Adv}_{\mathrm{AI}}^{\mathrm{DCDH}}(l)$ is nonnegligible. This contradicts the DCDH assumption.

4.3. Lemma 3

Assuming that the CDH problem is intractable, the advantage of a type II adversary AII against our protocol is negligible.

Proof

The proof process is similar to Lemma 2 and will not be repeated.

5. CONCLUSION

In this paper, we have proposed a multi-factor certificateless authentication key agreement protocol on elliptic curve, which avoids the limitations of certificate management and key custody. Based on DCDH and CDH assumptions, the protocol's security is guaranteed under mBR model. The scheme implements multi-factor joint participation in key negotiation, where KGC does not need to participate.If a new user is added, there is no need to update the information of all registered entities.

There are, however, a number of potential directions that we can extend this work. For example, we intend to identify potential collaborators that can assist in the implementation of our proposed scheme in a real-world setting. In addition,we also intend to further reduce time and communication costs.

REFERENCES

[1] Shamir A."Identity Based Cryptosystems and Signature Scheme. In G. R. Blakley, and David Chaum (Eds.)",1984.

[2] Al-Riyami S S, Paterson K G. Certificateless Public Key Cryptography: 9th International Conference on the Theory and Application of Cryptology, 2003.

[3] Fiore D, Gennaro R."Making the Diffie-Hellman Protocol Identity-Based",lecture notes in computer science,2010.

[4] Geng M, Zhang F. Provably Secure Certificateless Two-Party Authenticated Key Agreement Protocol without Pairing: International Conference on Computational Intelligence & Security, 2010.

[5] Hou M, Xu Q. A Two-party Certificateless Authenticated Key Agreement Protocol without Pairing: IEEE, 2009.

[6] Yang G, Tan C H. Strongly Secure Certificateless Key Exchange without Pairing: Acm Symposium on Information, 2011.

[7] He D, Chen Y, Chen J, et al."A New Two-Round Certificateless Authenticated Key Agreement Protocol without Bilinear Pairings",Mathematical & Computer Modelling,2011,54(11-12),pp. 3143-3152.

[8] Hu X, Wu Q, Zhong C."Toward Pairing-Free Certificateless Authenticated Key Exchanges",Springer, Berlin, Heidelberg,2011.

[9] Sun H, Wen Q, Zhang H, et al."A Novel Pairing-Free Certificateless Authenticated Key Agreement Protocol with Provable Security",Frontiers of Computer Science,2013, pp. 544-557.

[10] Kim Y J, Kim Y M, Choe Y J, et al."An Efficient Bilinear Pairing-Free Certificateless Two-Party Authenticated Key Agreement Protocol in the eCK Model",Journal of Theoretical Physics & Cryptography,2013,3(1).

[11] Suman B, Gaurav S, Verma A K."Impersonation Attack On CertificateLess Key Agreement Protocol",International Journal of Ad Hoc and Ubiquitous Computing,2018,27(2),pp. 108.

[12] Xie Y, Wu L, Shen J, et al."Efficient Two-Party Certificateless Authenticated Key Agreement Protocol Under GDH Assumption",International Journal of Ad Hoc and Ubiquitous Computing,2019,30(1),pp. 11-25.

[13] Yoon E J, Yoo K Y."Robust Biometrics-Based Multi-Server Authentication with Key Agreement Scheme for Smart Cards On Elliptic Curve Cryptosystem",Journal of Supercomputing,2013,63(1),pp. 235-255.

[14] He D, Wang D."Robust Biometrics-Based Authentication Scheme for Multiserver Environment",IEEE Systems Journal,2015,9(3),pp. 1-8.

[15] Chuang M C, Chen M C."An Anonymous Multi-Server Authenticated Key Agreement Scheme Based On Trust Computing Using Smart Cards and Biometrics",Expert Systems with Applications,2014.

[16] Chatterjee S, Roy S, Das A K, et al."Secure Biometric-Based Authentication Scheme using Chebyshev Chaotic Map for Multi-Server Environment",IEEE Transactions on Dependable & Secure Computing,2016,PP(99),pp. 1.

[17] Cao L, Ge W."A Secure and Efficient Multi-Factor Mutual Certificateless Authentication with Key Agreement Protocol for Mobile Client-Server Environment on ECC without the third-party",International Journal of Security and its Applications,2016,10(10),pp. 215-226.

[18] Mandal S, Bera B, Sutrala A K, et al."Certificateless Signcryption-Based Three-Factor User Access Control Scheme for IoT Environment",IEEE Internet of Things Journal,2020,PP(99),pp. 1.