Abstract—E-payment scheme allows two users to securely exchange e-cash and digital product over an open network. A problem in the across administrative domains E-payment scenarios is how the participants can carry out the exchange between administrative domains. In other words, the participants are administrated by two trusted administrators respectively. How can they verify their identities each other? In this paper, a certificateless cross-domain authenticated key exchange (CL-CD-AKE) scheme was proposed to solve this problem, and the security and the effectiveness of the proposed CL-CD-AKE scheme were analyzed in the extended random oracle model. Following this work, an E-payment scheme, achieving unforgeability and unreusability of e-cash, customer anonymity and fairness, was then proposed, and the CL-CD-AKE scheme was adopted by the E-payment scheme to deal with the problem of cross-domain authentication and key agreement.

Index Terms—electronic commerce, electronic payment, ID-based cryptography, certificateless authenticated key exchange, across administrative domains

I. INTRODUCTION

E-payment scheme is an important payment means to realize E-commerce. With the staggering development of E-commerce, E-payment scheme has aroused the attention of many scholars all over the world.

The first E-payment scheme was proposed by Chaum [1]. Over the next 30 years, E-payment has been well studied and many E-payment schemes have been put forward. So far, however, most E-payment systems are of certificate-based public key cryptography (CA-PKC). With the gradual expansion of users, the certificate management in CA-PKC has been a heavy burden. This burden limited the development of E-payment. To eliminating such limitation, some scholars recently have adopted identity-based cryptography (ID-PKC) [2] and certificateless cryptography (CLC) [3] to E-Cash schemes [4-9]. In ID-PKC and CLC, the user’s public key is derived from his identifier, such as email address or IP address etc., and the corresponding private key (or named as partial key in CLC) is generated by a trusted third party called as private key generator (PKG) (or called as key generation center, KGC, in CLC). ID-PKC and CLC remove the burden of CA-PKC by replacing the certificate management with the management of users’ identities, which greatly reduces the PKG’s load. However, the E-Cash schemes do not pay attention to the exchange between customers and venders, but only focuses on the payment from customers to venders.

Recently, Lin and Liu [10] proposed a new electronic payment scheme for digital content transactions that fulfilled fair exchange between customers and venders. A main attention of Lin’s work is how to encourage authors’ motivation to create high-quality digital contents. Yang and Chang [11] proposed a non-signature authenticated encryption scheme on Elliptic curve and they constructed a fair electronic payment system based on the scheme. Other scholars focused on the mobile payment scenarios [12] and the payment scenarios that use the smart card [9].

As we can notice, most attentions have been paid to the payment systems within a single domain. In such systems, transactions are both easy and secure, because users have convenient systems for sharing and have a trusted administrator that can make them secure. This article might consider a practical situation- the across administrative domains E-commerce scenario- in which Alice is a user of a administrative domain supervised by a bank $B_i$ and Bob belongs to another domain supervised by a bank $B_j$ ($B_i \neq B_j$), then there is a problem how can Alice authenticate and fairly exchange with Bob?

Our contributions. We address this problem by presenting an efficient certificateless cross domain authenticated key exchange (CL-CD-AKE) scheme and an E-payment scheme based on the CL-CD-AKE. In contrast, all the existing electronic payment schemes available in the literature, to our knowledge, are limited within a single domain. In addition, formal security proofs are provided to support our CL-CD-AKE scheme,
in the random oracle model, with the hardness of the BDH problem and the CDH problem.

The rest of this paper is organized as follows: some preliminaries are presented in section 2; section 3 presents the system framework of the cross domain E-payment scheme; section 4 deals with the definition of CL-CD-AKE scheme and its security notions; section 5 presents our CL-CD-AKE scheme and E-Payment scheme, followed by security and efficiency analysis in section 6; finally, some concluding remarks are given in Section 7.

II. PRELIMINARIES

In their pioneer work of Boneh and Franklin [2], a bilinear map, called “pairing”, is typically used. The pairing is a modified Weil pairing or Tate pairing on a super singular elliptic curve or abelian variety. For the reason of brevity, we describe pairings and the related mathematics in a more general format. Let $k$ be a security parameter and $q$ be a $k$-bit prime number. Let $G_1$ and $G_2$ be two cyclic groups of the same large prime order $q$. We assume that $G_1$ is an additive group and $G_2$ is a multiplicative group, respectively. Let $e: G_1 \times G_1 \rightarrow G_2$ be an admissible pairing which satisfies the following properties:

- **Bilinearity**: For $(P, Q) \in G_1$, and $(a, b) \in \mathbb{Z}_q^*$, there are $e(aP, bQ) = e(P, Q)^{ab}$.

- **Non-degeneracy**: there are $(P, Q) \in G_1$ such that $e(P, Q) \neq 1$.

- **Computability**: For $(P, Q) \in G_1$, one can compute $e(P, Q) \in G_2$ in polynomial time.

Next, we introduce two assumptions, the Computation-Diffie-Hellman (CDH) and the Bilinear Diffie-Hellman (BDH), which form the basis of the security for our schemes.

**Definition 1 (CDH)**: For $a, b \in \mathbb{Z}_q^*$, given $aP, bP \in G_1$, computing $abP \in G_1$ is hard.

**Definition 2 (BDH)**: For $a, b, c \in \mathbb{Z}_q^*$, given $(aP, bP, cP) \in G_1$, computing $e(P, P)^{abc} \in G_2$ is hard.

For reducing the communication and computation costs, Weil pairing and Tate pairing were proposed. These two kinds of bilinear pairings based on elliptic curves cryptosystems can provide a high security level while using relatively short keys.

III. THE FRAMEWORK OF THE E-PAYMENT SCHEME

This section presents the system framework of the cross administrative domains E-payment scheme. We assume that a Certificate Authority (CA) issues the public key certificates for the banks and a list of system parameters. That is, all the banks form a group supervised by the CA, and the registration approach from CA to banks adopts the Public Key Infrastructure (PKI) [13] technology. Each bank acts as a KGC and manages a group composed by all of its depositors. So, each bank and its customers form an administrative domain based on the CLC. We focus on the transactions between two customers, Alice and Bob, who belong to two banks respectively. The basic model is illustrated as Figure 1.

![Figure 1. The framework of the E-payment scheme](image)

IV. MODELING CERTIFICATELESS CROSS DOMAIN AKE

A. Algorithms of certificateless cross domain AKE

A certificateless cross-domain AKE scheme consists of eight polynomial-time algorithms: Setup, Set-Key-Pair, Certificate-Generation, Partial-Key-Generation, Set-Secret-Value, Set-Private-Key, Set-Public-Key, and Key-Exchange. These algorithms are defined as follows.

- **Setup**: This algorithm takes as input a security parameter $\lambda$ and returns a master secret key $msk$ of CA and a list of system parameters $params$.

- **Set-Key-Pair**: This algorithm takes as input the parameter list $params$ and an identity $ID_i$ to produce the public-private key pair $(P_i, S_i)$ for the bank $Bi$.

- **Certificate-Generation**: This algorithm is run by CA. It takes as input $params$, $msk$ and a public key $P_i$ selected by bank $Bi$, to produce the public key certificate $Cert(B_i)$ for the $Bi$.

- **Partial-Key-Generation**: This algorithm is run by $KGC_i$ for each bank $Bi$ having been selected by the CA. It takes as input $params$, $SKC_i$, a private key $S_i$ and an identity $ID_i$, to produce the partial private key $D_{ij}$ for the party $j$ whose identity is $ID_j$.

- **Set-Secret-Value**: This algorithm takes as input $params$ and an identity $ID_i$ to produce the secret value $x_i$ for the party $i$.

- **Set-Private-Key**: This algorithm takes as input $params$, an identity $ID_i$, a partial private key $D_{ij}$ and a secret value $x_i$ to produce a private key $SK_i$ for the party $j$.

- **Set-Public-Key**: This algorithm takes as input $params$, an identity $ID_i$ and secret value $x_i$ to produce a public key $PK_i$ for the party $j$.

- **Key-Exchange**: This is a probabilistic polynomial-time interactive algorithm which involves two parties $C$ and $V$. The inputs are the system parameters $params$ for both $C$ and $V$, plus $(SKC_i; ID_i; PK_i; BC_i)$ for $C$, and $(SKV_j; ID_j; PKV_j; BV_j)$ for $V$. Here, $SKC$ and $SKV$ are the respective private keys of $C$ and $V$, $ID_i$ is the identity of $C$ and $ID_j$ is the identity of $V$, $PK_i$ and $PKV_j$ are the public keys of $C$ and $V$, respectively; $BC_i$ is the identifier of $C$’s bank and $BV_j$ is the identifier of $V$’s bank. Eventually, if the protocol...
does not fail, C and V agree on a secret session key $K_{CV} = K_{VC}$.

B. Security model of certificateless cross domain AKE

Before modeling the certificateless cross-domain AKE protocols, we first review the adversaries in CLC. Al-Riyami and Paterson [3] defined the formal security notions for CLC that resist the attacks from two types of adversaries denoted as “AdvI” and “AdvII”. A type I adversary AdvI models an “outsider” adversary who can replace the public key of an arbitrary entity. A type II adversary AdvII models an “insider” adversary who is assumed to possess the master key and can obtain the partial keys instead of replacing the target user’s public key. Similarly, in a certificateless cross-domain AKE scheme, a type II adversary is assumed to possess the private keys of both KGCs.

AdvI. This adversary does not know the master key of CA and the private key of any KGC, but has the ability to replace the public key of any entity with a value of her choice.

AdvII. This adversary knows the private keys of both KGCs, but she does not know the master key of CA and cannot replace the target user’s public key.

We will use the random oracle model extended by Chen, Cheng, and Smart [14] and Zhang et al. [15] to model our certificateless cross domain AKE protocols. The model is defined by two games with two phases between a challenger CH and an adversary $Adv \in \{AdvI, AdvII\}$. Adv is modeled by a probabilistic polynomial-time Turing machine, and its behavior is modeled by many oracles maintained by CH. All communications go through the Adv. Participants only respond to the queries by Adv and do not directly communicate among themselves. Adv can relay, modify, delay, interleave or delete all the message flowed in the system.

In the first phase of the games, an adversary (AdvI or AdvII) is allowed to issue some queries (defined in Game I) in any order. Once the first phase is over, the adversary starts the second phase by choosing a fresh oracle $\Pi^{I}_{ij}$ and issuing a Test $(\Pi^{I}_{ij})$ query. An oracle $\Pi^{I}_{ij}$ denotes the $u$-th instance of party $i$ involved with a partner party $j$ in a session, and the fresh oracle and test query are defined in the following. After the second phase, the adversary may continue querying the oracles, but the queries are restrained. Finally the adversary outputs a guess for the Test $(\Pi^{I}_{ij})$.

Definition 3 (fresh oracle): An oracle $\Pi^{I}_{ij}$ is fresh if (1) $\Pi^{I}_{ij}$ has accepted; (2) $\Pi^{I}_{ij}$ is unopened (not being issued the Reveal query); (3) party $j \neq i$ is not corrupted (not being issued the Corrupt query); (4) there is no opened oracle $\Pi^{I}_{ij}$, that has a matching conversation with $\Pi^{I}_{ij}$.

Game I. The first game is played between a challenger CH and an AdvI adversary $A_{I}$ as follows.

- **Initialization.** The challenger runs the Setup and Set-Key-Pair algorithms. It then gives the adversary the system parameters $params$ and the banks’ public keys, $P_{i}$ for $B_{i}$ and $P_{j}$ for $B_{j}$ respectively, and it keeps the master secret key $msk$ and the banks’ private keys, $S_{i}$ for $B_{i}$ and $S_{j}$ for $B_{j}$ respectively, to itself.
- **Phase I.** $A_{I}$ may adaptively issue a polynomially bounded number of queries to CH, i.e. each query may depend on the answers to the previous queries.
  - Request public key ($ID_{i}$). When $A_{I}$ supplies an identity $ID_{i}$ and requests the public key for $i$, $CH$ returns the corresponding public key $PK_{i}$.
  - Extract partial key ($ID_{i}, B_{i}$). When $A_{I}$ supplies an identity $ID_{i}$ and requests $i$’s partial key, $CH$ responds with the partial key $D_{i}$ for the identity.
  - Replace public key ($ID_{i}, PK^{*}_{i}$). When $A_{I}$ supplies an identity $ID_{i}$ and a new public key value $PK^{*}_{i}$, $CH$ replaces the current public key with the new $PK^{*}_{i}$.
  - Request secret value ($ID_{i}$). $A_{I}$ may request the secret value for an identity $ID_{i}$. To respond, the challenger returns the secret value $x_{i}$ of party $i$.
  - Send ($\Pi^{I}_{ij}, M$): $A_{I}$ can send a message $M$ of her choice to an oracle $\Pi^{I}_{ij}$. Upon receiving the message $M$, oracle $\Pi^{I}_{ij}$ executes the protocol and responds with an outgoing message $m$ or a decision to indicate accepting or rejecting the session. If the oracle $\Pi^{I}_{ij}$ does not exist, it will be created as initiator if $M=\lambda$, or as a responder otherwise. We require $i \neq j$, since a party will not run a session with itself.
  - Reveal ($\Pi^{I}_{ij}$): $A_{I}$ can ask a particular oracle to reveal the session key. If the oracle has not accepted, it returns $\perp$; otherwise, it reveals the session key.

- **Phase II.** $A_{I}$ may choose a fresh oracle $\Pi^{I}_{ij}$ to ask a Test ($\Pi^{I}_{ij}$) query. Note that, $A_{I}$ has never requested the partial key of party $j$ in Phase I.
  - Test ($\Pi^{I}_{ij}$). To answer the Test query, oracle $\Pi^{I}_{ij}$ randomly flips a coin $b \in \{0, 1\}$ and responds with the session key if $b=0$, or a random sample from the distribution of the session key otherwise. After the second phase, $A_{I}$ can continue querying the oracles except that it cannot reveal the test oracle $\Pi^{I}_{ij}$ or its partner $\Pi^{I}_{ij}$, and it cannot corrupt party $j$. Finally, $A_{I}$ outputs a guess $b$ for $b$. If $b = b$, we say that the adversary wins. The $A_{I}$’s advantage is defined as

$$Adv_{A_{I}}(k) = \max \{0, \Pr[A_{I}, \text{wins}] - \frac{1}{2}\}. \quad (1)$$

Game II. The second game is played between a challenger CH and an AdvII adversary $A_{II}$ as follows.

- **Initialization.** The challenger runs the Setup and Set-Key-Pair algorithms. It then gives the adversary the system parameters $params$ and the banks’ key pairs, $(P_{i}, S_{i})$ for $B_{i}$ and $(P_{j}, S_{j})$ for $B_{j}$ respectively, and it keeps the master secret key $msk$ to itself.
• Phase I. $A_H$ may adaptively make a polynomially bounded number of queries as in Game I.

• Phase II. $A_H$ may choose a fresh oracle $\Pi_{s_j}^*$ to ask a test ({$\Pi_{s_j}^*$}) query as in Game I. Note that, $A_H$ has never replaced the public key of the party $j$ in Phase I.

After this point the adversary can continue querying the oracles except that it cannot reveal the test oracle $\Pi_{s_j}^*$ or its partner $\Pi_{s_j}^*$, and it cannot corrupt the party $j$. Finally, $A_H$ outputs a guess $b$ for $b$. If $b = b$, we say that the adversary wins. The $A_H$’s advantage is defined as

$$Adv_{A_H}(k) = \max \{0, \Pr[A_H\text{ wins}] - \frac{1}{2}\}. \quad (2)$$

Definition 4: A CL-CD-AKE scheme is said to be secure if:

1. In the presence of a benign adversary on $\Pi_{s_j}^*$ and $\Pi_{s_j}^*$, both oracles always gain the same session key, and this key is distributed uniformly on $\{0, 1\}^l$. If for any adversary ($A_I$ or $A_B$), $Adv_{A_I}(k)$ is negligible.

V. E-PAYMENT SCHEME

In this section, we propose a certificateless cross domain AKE scheme and an E-payment protocol as follows.

A. Certificateless cross domain AKE

The certificateless cross domain AKE scheme is proposed on the basis of above-mentioned bilinear pairings and assumptions of CDH and BDH.

Setup: This algorithm takes as input $l \in Z_2^*$, and outputs $\textbf{params} = (G_1, G_2, e, P, r_0, H, H_1, H_2, l)$. Where, $(G_1, G_2, e)$ is as above, $P$ is a generator of $G_1$, $s \in Z_2^*$ is a master-key randomly chosen for CA and $P_0 = sP$ is the associated public key. $H_1 : \{0,1\}^l \rightarrow G_1$ and $H_2 : \{0,1\}^l \times G_1 \times G_2 \rightarrow \{0,1\}^l$ are secure one-way hash functions.

Set-Key-Pair: This algorithm takes as input $(\textbf{params}, B), \textbf{B}$, and outputs $(P_i, S_i)$ as the public-private key pair for the bank $B_i$. Here, $S_i = a_i \in R Z_2^*$ is a private key, and $P_i = a_i P$ is the associated public key.

Certificate-Generation: This algorithm takes as input $(\textbf{params}, \text{msk}, P_i, B_i)$, and outputs $\text{Cert}(B)$ as the public key certificate for the bank $B_i$. This algorithm may refer to X.509 [13].

Partial-Key-Generation: This algorithm takes as input $(\textbf{params}, S_j, ID_j), \textbf{B}$, and outputs $D_j = a_j Q_j$ as the partial key for $j$. Here, $Q_j = H_j(ID_j)$ and $ID_j \in \{0,1\}^l$ is an identifier of $j$.

Set-Secret-Value: This algorithm takes as input $(\textbf{params}, ID_j)$, $x_j \in R Z_2^*$ as the secret value of $j$.

Set-Private-Key: This algorithm takes as input $(\textbf{params}, ID_j, D_j, x_j)$, and outputs $SK_j = (x_j, D_j)$ as the private key of $j$.

Set-Public-Key: This algorithm takes as input $(\textbf{params}, ID_j, D_j, x_j)$, and outputs $PK_j = x_j P$ as the public key of $j$.

Key-Exchange: Assume that entities $C$ and $V$ have key pairs $(SK_c, PK_c)$ and $(SK_v, PK_v)$ respectively, and that $C$ pertains to the bank $B_C$ and $V$ pertains to the bank $B_V$. $C$ runs the key-exchange with $V$ as follows.

i. $C$ selects $r_c \in Z_2^*$ at random, computes $R_c = r_c P$, and sends $(ID_c, PK_c, R_c, B_c)$ to $V$;

ii. When $V$ receives $(ID_c, PK_c, R_c, B_c)$ from $C$, he randomly choose $r_i \in Z_2^*$, calculates $R_i = r_i P$, and sends $(ID_i, PK_i, R_i, B_i)$ to $C$. Then, he obtains the $B_C$’s public key $P_c = a_i P$ from CA, and computes $\delta_c = e(R_i, D_c) \cdot (r_i P_i, Q_c)$, and the session key $K_{cV} = H_2(ID_c, ID_i, R_c, R_i, \delta_c) \cdot (r_i P_i, Q_c)$ and the session key $K_{cv} = H_2(ID_c, ID_i, R_c, R_i, \delta_c)$.

In every run, the session ID is $(ID_c, ID_i, R_c, R_i)$. The process of the Key-Exchange is illustrated as Figure 2.

![Figure 2](https://example.com/figure2.png)

The Key-Exchange is a typical Diffie–Hellman key exchange [16]. Nevertheless, our protocol achieves implicit identity authentication for entity, and avoids the man-in-the-middle attack that exists in the Diffie–Hellman key exchange. The correctness of the key exchange is verified as follows.

$$r_c R_c = r_c r_i P = r_i R_c. \quad (3)$$

$$\delta_c = e(R_i, D_c) \cdot (r_i P_i, Q_c) = e(r_i P_i, a_i Q_c) \cdot (r_i a_i P, Q_c) = e(r_i P_i, P_c) = e(r_i P_i, P_i) \cdot (r_i a_i P, Q_c) \cdot (r_i, P_i) \cdot (Q_c, D_c) = \delta_i. \quad (4)$$

$$r_i PK_i = r_i x_i P = x_i R_i. \quad (5)$$

$$r_c PK_c = r_c x_c P = x_c R_c. \quad (6)$$
So, it can be derived from the equations 3-6 that the equation $K_{cv} = K_{vc}$ is holds, which proves the correctness of the Key-Exchange protocol. Meanwhile, in order to calculate a same session key as $V$, entity $C$ must embed his private key $SK_v = (x_c, D_c)$ and a randomly generated temporary secret value $r_c$, which achieves the implicit authentication from $V$ to $C$, vice versa.

B. E-payment scheme

The new E-payment protocol involves five entities, that is, consumer $C$, vendor $V$, $C$’s bank $BC$, $V$’s bank $BV$ and a trusted third party $T$ (CA). Their identifier and key pairs are $(ID_C, SK_C, PK_C), (ID_V, SK_V, PK_V), (ID_{BC}, SK_{BC}, PK_{BC}), (ID_{BV}, SK_{BV}, PK_{BV}).$ We will use a general-purpose CL-AKA algorithm, e.g., the scheme had been proposed by Zhang et al. [15], to deal the intra-domain authentication and key agreement, and a certificateless blind signature algorithm suggested by Zhang and Gao [17]. We assume that $C$ and $V$ have opened an account $i, (i \in \{C, V\})$ at their banks $B_i(i \in \{C, V\})$ respectively. The main work flow is as follows:

Cash Generation. The customer draws e-cash from his bank $BC$.

First, $C$ establishes secure communication links with $BC$ and has to prove that an account number belongs to him by the CL-AKA. That is to say, the customer should prove he knows the knowledge $SK_C$ from an identifier $ID_C$. Then, $C$ sends related information to $BC$ (we assume that $C$ and $V$ reach a consensus on money, the price of product, before this scenario). $BC$ checks $C$’s account balance. If the account balance is greater than or equal to money, e-cash is generated and sent back to $C$, and the corresponding moneys are frozen. If not, a false would be sent back. The process is as follows.

\[ C \rightarrow BC: E_{sk_c}(ID_C, ID_V, ID_{BC}, ID_{BV}, money, t_1) \]
\[ BC \rightarrow C: E_{sk_c}(e - cash) / false \]

Here, e-cash=$((ID_C, ID_V, ID_{BC}, ID_{BV}, money, t_2, t_3), \sigma_B),$ and $\sigma_B$ is a signature on messages $(ID_C, ID_V, ID_{BC}, ID_{BV}, money, t_2, t_3)$ by the bank $BC$ using certificateless blind signature algorithm [17]. $E_k(N)$ denotes encrypted message by a session key $K$ agreed by using CL-AKA, and the encrypt algorithm may refer to the AES [18]. $ID_{pro}$ is an index of product, and $I_c$ is an account number belonging to $C$. $t_1$ and $t_2$ are timestamps, and $t_3$ is a deadline of e-cash.

Payment. Firstly, $C$ and $V$ run the certificateless cross domain AKE scheme described in section 5.1, and then, they exchange e-cash and digital product as follows.

\[ C \rightarrow V: E_{sk_c}(e - cash) \]
\[ V \rightarrow C: E_{sk_c}(product) \]
\[ C \rightarrow V: E_{sk_c}(\sigma_c) \]

In the process, $V$ checks whether payment is valid or not. That is, $V$ verifies $\sigma_c$ and checks time of validity. If the e-cash is valid, product will be sent to $C$. If $C$ verifies that product is valid, $C$ sends a payment confirmation $\sigma_c$ that is a signature on messages $(e - cash, t_4)$ to $V$. Here, $t_4$ is a new timestamp.

Compensation. We introduce a compensation sub-protocol to compensate the possible misbehaviors of participant in the payment stage. That is, if a dishonest party $C$ does not confirm payment after receiving product, $V$ can apply for compensation after waiting some time (note that the time must come before the expiration date of payment). The process is as follows.

Firstly, $V$ and $T$ establish a secure communication link. Then, operations are as follows.

\[ V \rightarrow T: E_{sk_v}(e - cash, product) \]
\[ T \rightarrow V: E_{sk_v}(\sigma_v) / false \]

In the process, $T$ should check the validity of e-cash and product (here, we assume that $T$ can verify the validity of product. Relevant technology may refer to the work of Ray, Ray and Natarajan [19]). If e-cash and product are both valid, the messages, $(e - cash, \sigma_v)$, is signed and sent to $V$ by $T$. Then, $T$ sends product to $C$.

\[ T \rightarrow C: E_{sk_v}(product) \]

Deposit. Firstly, $V$ establishes secure link with his bank $B_V$, and sends the payment instruments to $B_V$.

\[ V \rightarrow B_V: E_{sk_c}(e - cash, \sigma_c, \cdot \cdot \cdot) \]

$B_V$ forwards the e-cash and $\sigma_c$ to $C$’s bank $BC$. $BC$ checks that the e-cash and payment confirmation are both valid; money will be deposited into $V$’s account. Note that the deposit operations should be finished before the expiring date of e-cash, and the payment confirmation signed by $C$ or $T$ is valid. In addition, $V$ can do one-time settlement for multiple e-cashes in order to reduce communication costs.

VI. ANALYSIS AND COMPARISON

The security and effectiveness of the proposed schemes are discussed in this section. As the limitation of article space, we give a heuristic proof but not details.

A. Analysis of the certificateless cross domain AKE

We now state the security results for the certificateless cross domain AKE scheme under the definition 4.

Theorem 1. The proposed certificateless cross domain AKE scheme is a secure two-party AKE protocol.

Proof: This theorem follows Lemmas 1 - 4.

Lemma 1. In the presence of a benign adversary [14], both participants always agree on the same session key as if there was no adversary, and this key is distributed uniformly at random.

Proof: Suppose that the two participants $i$ and $j$ follow the certificateless cross domain AKE and the adversary $(A_1$ or $A_2)$ is benign. In this case, both parties correctly receive formatted messages as originally sent by the other one; according to the equations 3-6 in section $V$, they will agree on the same session key. Since $r_1$ and $r_2$ are randomly selected by participants $i$ and $j$, respectively, the session key can be considered as the output of the hash function $H_2$ on a random input. Based on the properties of cryptographic hash functions, the session key is uniformly distributed over $\{0, 1\}^l$.

Lemma 2. If the BDH assumption holds and the hash functions are modeled as random oracles, the advantage
of an AdvI against our certificateless cross domain AKE scheme is negligible.

Proof: The Game I is started by CH running the Setup and Set-Key-Pair algorithms. CH gives the AdvI adversary A; the system parameters params = (G, G*, e, P, Pp, H, Hl, l) and the banks’ public keys, P, for B and Pj for Bj respectively. The functions H1 and H2 are instantiated as random oracles under the control of CH. Suppose that A; can win the Game I with a non-negligible advantage AdvIp(k) in polynomial-time t. Given a BDH problem instance (aP, bP, cP) (specifically, let P = aP, Q = bP and P = cP for A; who impersonates initiator i, or let P = aP, Q = bP and P = cP for A; who impersonates responder j), we can show how construct an algorithm CH using the adversary A; against the protocol to solve the BDH problem, i.e. to compute eP(P, P)c. Specifically, suppose that, in the attack, A; who makes at most qf times queries to Hi for i = 1, 2 and creates at most qo oracles, wins the game with advantage AdvIp(k). Then there exists an algorithm CH to solve the BDH problem with advantage τ(k) ≥ 1qf − qo · q2 advIp(k).

Lemma 3. If the CDH assumptions holds and the hash functions are modeled as random oracles, the advantage of an AdvII against our certificateless cross domain AKE scheme is negligible.

Proof: The Game II is started by CH running the Setup and Set-Key-Pair algorithms. CH gives the AdvII adversary AII the system parameters params = (G, G*, e, P, Pp, H, Hl, l) and the banks’ key pairs, (P, S) for Bi and (Pp, S) for Bj respectively, and keeps the master secret key s to itself. The hash functions H1 and H2 are instantiated as random oracles under the control of CH. We suppose that AII wins the game with a non-negligible advantage AdvIIP(k) in polynomial-time t. Then we can show that there is an algorithm CH that solves the CDH problem in G1 with non-negligible probability. That is, given an arbitrary input (P, aP, bP) (specifically, let xP = aP and rP = bP for AII who impersonates initiator i, or let xP = aP and rP = bP for AII who impersonates responder j), we can show how CH can solve the CDH problem in G1, i.e. to compute abP. Specifically, suppose that, in the attack, AII who makes at most qf times queries to Hi for i = 1, 2 and creates at most qo oracles, wins the game with advantage AdvIIP(k). Then there exists an algorithm CH to solve the CDH problem with advantage τ(k) ≥ 1qf − qo · q2 advIIP(k).

Lemma 4. Our certificateless cross domain AKE scheme has the perfect forward secrecy property if the CDH problem in G1 is hard and H2 is modeled as random oracle.

Proof: Suppose that two parties i and j established a session key SKij using our scheme, and later, their private keys were compromised. Let ri and rj be the ephemeral secret values used to establish SKij by i and j respectively, and they are not exposed. Clearly, to compute the values of riP and SKij, the adversary who owns R = riP and R = rjP for unknown ri and rj must face to solve the CDH problem in G1. By the CDH assumption, this is impossible.

B. Analysis of E-payment scheme

In this subsection, we will show that the proposed E-payment scheme is secure and effective.

Claim 5. The proposed E-payment protocol achieves the unforgeability and unreusability of e-cash.

Proof: Before generating the e-cash, the identity of consumer C must be verified by his bank through CL-AKE [15], and a session key was established. Hence, the e-cash signed by B using certificateless blind signature algorithm [17] is unforgeable since the EUF-CMA of signature algorithm and the securities of CL-AKE have been proved. In addition, each e-cash is bound up with a sale transaction by bank’s signature, and it cannot be used to pay for other products. So, the proposed E-payment scheme realizes the unforgeability and unreusability of e-cash.

Claim 6. The proposed E-payment scheme meets the anonymity of users.

Proof: It is easy to see that, both C and V do not know any true Identity information but ID number and public key of their counterparty from transaction process. ID number, defined as a bit string randomly picked, and public key of user are unrelated with user’s true Identity. Hence, the new scheme achieves the anonymity of users.

Claim 7. The proposed E-payment scheme is a fair exchange protocol.

Proof: Suppose that banks and T are honest participants that do not conspire with C or V who is possible participant engaged in misbehaviors. We identify three misbehaviors of destroying fairness according to the definition of fair exchange [20]. The explanations and detections of possible misbehaviors are as follows.

Case 1. C sends an incorrect or stale e-cash to V in payment sub-protocol (in section 5.2). V will detect an error when he verifies the σC or timestamp, and would not send product to C.

Case 2. V receives a valid e-cash but does not deliver correct product to C, he would not attain the payment confirmation, σC, from C. If he wants to obtain σT from T by using compensation sub-protocol, he must send valid product to T. If not, he cannot receive σT.

Case 3. C receives valid product, but he does not send σC to V. Then, V needs to send compensation request to T and receives valid σT from T.

Based on above analyses, the proposed E-payment scheme shows good quality of fairness.

C. Comparisons

Based on current available knowledge, the AKE schemes and the E-payment schemes have seldom been considered in the across administrative domains setting. Table I compares the computing cost among our CL-CD-AKE scheme and two AKE protocols, the one (denotes SCK-1 [14]) is based on the ID-PKC and the other (denotes ZZWD [15]) is a certificateless one. Table II
compares our E-Payment scheme with four protocols, three E-cash schemes (denotes WTL-cash [7], WCW-cash [8] and RO-cash [9] respectively) and an E-payment scheme (denotes LL-payment [10]).

<table>
<thead>
<tr>
<th>AKE schemes</th>
<th>Cross domain authentication</th>
<th>Parings (times)</th>
<th>Multiplication (times)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCK-1</td>
<td>No</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>ZZWD</td>
<td>No</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>CL-CD-AKE</td>
<td>Yes</td>
<td>2</td>
<td>5</td>
</tr>
</tbody>
</table>

TABLE I. COMPARISONS AMONG DIFFERENT E-PAYMENT SCHEMES

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>WTL-cash</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>traceability</td>
<td>ID-PKC</td>
</tr>
<tr>
<td>WCW-cash</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>traceability</td>
<td>CLC</td>
</tr>
<tr>
<td>RO-cash</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>had not been</td>
<td>ID-PKC</td>
</tr>
<tr>
<td>LL-payment</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>fairness</td>
<td>CA-PKC</td>
</tr>
<tr>
<td>Our E-Payment scheme</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>fairness</td>
<td>CA-PKC + CLC</td>
</tr>
</tbody>
</table>

As we can see from the Table I, even though the computing cost of our CL-CD-AKE is slightly higher than the others’, it is acceptable. However, the CL-CD-AKE scheme is the only one that achieves the cross domain authentication.

Different from our E-payment scheme, References [7-10] did not take the cross domain exchange situations into account. WTL-cash and WCW-cash schemes only realized the traceability of users’ behaviors in the transaction, and the fairness of RO-cash scheme had not been discussed. The traceability cannot fully ensure that the E-payment scheme is finally fair, for external arbitration is uneasy. Similar to [10], we contributed compensation protocol to achieving fairness at the period of exchange. In addition, the schemes in References [7-9] were proposed based on the ID-PKC (or CLC in [8]), and those schemes assumed that all the banks and customers form a group supervised by a central bank. Obviously, their schemes do not provide effective solutions in across administrative domains E-commerce scenarios, since it is difficult to make all banks and their depositors pertain to one group in the real world. The LL-payment scheme adopted the CA-PKC cryptography, but it is known to all that there is a certificate management burden in the CA-PKC. This burden has limited the development of E-payment. In our system, CA is only used to administer the banks’ public key certificates and to issue the public system parameters, and every customer is supervised by his own bank instead of CA. Because the quantity of banks is far less than that of the depositors, we successfully remove the certificate management burden by reducing the number of users in CA-PKC.

VII. CONCLUSION

Digital product transactions through e-commerce will grow tremendously in the coming years. Well-designed electronic payment protocol is a critical successful factor. This paper proposes a certificateless cross domain authenticated key exchange scheme and an E-payment scheme. The former scheme circumvents the key escrow issue inherited in the ID-based AKE schemes and the certificate management burden in CA-based cryptosystems, and achieves the cross domain authentication. The proposed E-payment scheme, based on the CL-CD-AKE, makes two participants, who pertain to two authentication domains respectively, securely carry out their transaction. The security of proposed CL-CD-AKE is proved in the extended random oracle model, and the E-payment scheme can ensure all important properties listed ahead.

The proposed E-Payment scheme at this time is in its early stage. In the future, we will focus on the following topics. The performance evaluating and formal analyzing of the E-payment scheme must be implemented to measure whether the protocol performance is acceptable or not, and to ensure that the proposed scheme is truly fair in real world.

ACKNOWLEDGMENT

We would like to thank anonymous referees for their useful comments. This work was supported in part by the Major Research plan of the National Natural Science Foundation of China under Grant No.90818028.

REFERENCES


© 2011 ACADEMY PUBLISHER


Ming Chen received the B.S. and M.S. degrees in Computer system and Architecture from Chongqing University, China, in 2003 and 2007, respectively. He is currently a candidate of Ph. D. in College of Computer Science of Chongqing University. His current research interests are in the areas of e-commerce, protocol design and formal analysis, and monitoring distributed software.

Kaigui Wu received the M.S. and Ph. D. degrees from Chongqing University, China, respectively. He is currently an Associate Professor of in College of Computer Science of Chongqing University. His current research interests are in the areas of distributed computing, information security, and protocol design.

Jianjun Du is currently a candidate of Ph. D. in College of Computer Science of Chongqing University.

Jie Xu is Chair of Computing, Lead of a Peak of Excellence at the University of Leeds, UK and Director of the EPSRC-funded WRG e-Science Centre involving the three White Rose Universities of Leeds, York and Sheffield. He has worked in the field of Distributed Computer Systems for over twenty-five years. He is the recipient of the BCS/IEEE Brendan Murphy Prize 2001 for the best work in the area of distributed systems and networks. He has led and co-led many key research projects to the value of over £20M. Professor Xu has published more than 250 edited books, book chapters and academic papers. He received a PhD from the University of Newcastle upon Tyne, and then moved to the University of Durham as the head founder of the Durham Distributed Systems Engineering group. He was Professor of Distributed Systems at Durham before he joined the University of Leeds. He is a member of IEEE Computer Society.