Electronic Commerce Research and Applications 10 (2011) 673–682

Contents lists available at SciVerse ScienceDirect

Electronic Commerce Research and Applications

journal homepage: www.elsevier.com/locate/ecra

A novel electronic cash system with trustee-based anonymity revocation from pairing

Yalin Chen [a], Jue-Sam Chou [b,*], Hung-Min Sun [a], Ming-Hsun Cho [b]

[a] National Tsing-hua University, 101, Section 2, Kuang-Fu Road, Hsinchu, Taiwan
[b] Nanhua University, No. 55, Sec. 1 Nanhua Rd., Zhongkeng, Dalin Township, 62248 Chiayi County, Taiwan

ARTICLE INFO

Article history:
Received 11 July 2009
Received in revised form 3 June 2011
Accepted 3 June 2011
Available online 21 June 2011

Keywords:
Anonymity revocation
Bilinear pairing
E-cash
Digital cryptography
Mutual authentication

ABSTRACT

Untraceable electronic cash is an attractive payment tool for electronic-commerce because its anonymity property can ensure the privacy of payers. However, this anonymity property is easily abused by criminals. In this paper, several recent untraceable e-cash systems are examined. Most of these provide identity revealing only when the e-cash is double spent. Only two of these systems can disclose the identity whenever there is a need, and only these two systems can prevent crime. We propose a novel e-cash system based on identity-based bilinear pairing to create an anonymity revocation function. We construct an identity-based blind signature scheme, in which a bank can blindly sign on a message containing a trustee-approved token that includes the user's identity. On demand, the trustee can disclose the identity for e-cash using only one symmetric operation. Our scheme is the first attempt to incorporate mutual authentication and key agreement into e-cash protocols. This allows the proposed system to attain improvement in communication efficiency when compared to previous works.

1. Introduction

A typical e-cash system consists of three roles – the customer, bank (issuer or acquirer), and merchant, and three protocols – a withdrawal protocol, a payment protocol, and a deposit protocol. As a requirement of an e-cash system, when a customer withdraws e-cash from an issuing bank and pays a merchant, and the merchant deposits it at an acquiring bank, no one can link the e-cash to the customer. Unlike other e-payment tools such as electronic account transfer, financial electronic data exchange (FEDI), or a credit card payment system, an e-cash system, which possesses an anonymity property like traditional cash, can ensure the privacy of payers and avoid the risk of identity theft and customer fraud (Ashrafi and Ng 2009). Therefore, e-cash payments are desirable for electronic commerce.

Chaum (1983) first proposed the concept of e-cash and its paper cash-like properties of anonymity, verifiability, and unforgeability. He implemented these required properties using a blind signature primitive. In this primitive, a customer chooses a random e-coin number, blinds it (makes it indistinguishable from the original one), and then asks his bank to sign on it. Upon receiving the request, the bank first confirms the customer's identity and then debits his account, signs on the blind e-coin number, and returns the signed result.[1] Then, the customer unblinds it and obtains the bank's signature on the e-coin number. Thus, the e-cash has the form {e-coin number, bank's signature on the e-coin number}. Here, the e-coin number is a random string that cannot be linked to any person, to ensure anonymity. Meanwhile, the bank's signature on the e-coin number can be verified publicly and is difficult to forge because of the intrinsic non-repudiation and unforgeability of a digital signature primitive.

E-cash is a series of digital bits. It can thus be easily duplicated and spent again. To prevent this, Chaum's e-cash system (1983) requires bank involvement in each customer-and-merchant transaction in order to check whether the e-coin is fresh. However, this approach will consume considerable bank resources and increase the communication overhead between merchants and banks. Hence, Chaum et al. (1990) later proposed a bank off-line (we use off-line instead in the remainder of this paper) e-cash system using a cut-and-choose technique to hide and reveal the user's identity in the e-cash. The e-cash remains anonymous when it is spent the first time, but the identity is revealed if it is spent again. Without using a cut-and-choose technique, Brands' e-cash system (1993, 1995) requires a payer to provide a zero-knowledge proof of the e-cash to a random challenge from a payee. If the e-cash is double spent, two different proofs can disclose the identity in the e-cash. However, these two kinds of approaches cannot prevent a perfect crime. They merely thwart double spending, but cannot deal with e-cash that was illegally spent only once; for example, Alice used e-cash to buy cocaine from Bob on the Internet. Bob

* Corresponding author. Tel.: +886 5 2721001; fax: +886 5 2427137.
E-mail address: jschou@mail.nhu.edu.tw (J.-S. Chou).

[1] A typical e-cash withdrawal protocol requires a pre-built authenticated channel, so it can ensure that the bank and customer have mutually authenticated before the e-cash withdrawal.

1567-4223/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.elerap.2011.06.002

then saved the e-cash in his bank account. One day the police arrested Bob and investigated the transactions for his bank account but the bank had no idea who Alice was.

Since Von Solms and Naccache (1992) first pointed out that blackmailers can commit a perfect crime by demanding a ransom in untraceable electronic e-cash, anonymity revocation has become a desirable property for e-cash. Researchers have therefore proposed trustee-based fair e-cash schemes (Stadler et al. 1995, Brickell et al. 1995, Fujisaki and Okamoto 1996, Camenish et al. 1996) in an attempt to achieve this property. In these schemes, a trustee is involved in the escrow of some critical information such as linking the owner's identity to the e-cash. Once a bank or a law enforcement agency asks for anonymity revocation, the trustee should execute an e-cash owner tracing protocol to recover the linkage to the e-cash owner.

Recently, Popescu and Oros (2007) and Wang et al. (2008) proposed two trustee-based anonymity-revocable e-cash systems using bilinear pairing. However, Popescu and Oros' scheme violates anonymity, and the scheme of Wang et al. has the deficiency that a malicious customer could use an unregistered certificate to withdraw e-cash from a bank. In addition, the withdrawal protocols in these two works, which both contain two sub-protocols, certificate proving and blind signature signing, require five and eight rounds, respectively. The number of rounds can be further improved to attain better efficiency. In view of this, this paper proposes a round-efficient scheme. This scheme has three features.

(1) It is very concise with respect to e-cash owner tracing. In this work, a novel identity (ID)-based blind signature scheme is constructed using pairing, in which the signed message contains a trustee-approved signer-unknown token and, if needed, the trustee can reveal the user's identity by using only one symmetric decryption and one exclusive-or (XOR) operation.
(2) It does not need to build an authenticated channel before each protocol run because the scheme embeds two functions, pairing-based mutual authentication and key agreement, into each protocol.
(3) It has round efficiency. The scheme requires only two rounds when issuing a license, two during a withdrawal, one during a payment, one during a deposit, and one during e-cash owner tracing. This makes the scheme more efficient in terms of communication than previous works.

The remainder of this paper is organized as follows. Section 2 provides some background information regarding bilinear pairing and its applications. It also reviews several recent e-cash systems and their deficiencies. Section 3 presents the trustee-based anonymity-revocable e-cash system based on pairing, and Section 4 analyzes its security. Section 5 offers a comparison of the e-cash features and communication efficiencies of the proposed scheme and recent works. Finally, Section 6 provides some concluding remarks and a description of future work.

2. Background techniques and recent work reviews

This section describes bilinear-pairing-related concepts in Section 2.1, the applications of an ID-based cryptosystem based on pairing in Section 2.2, and reviews recent works in Section 2.3.

2.1. Bilinear pairing

Bilinear pairing (Menezes et al. 1993) has been the subject of many research articles in the last decade. The following section briefly introduces this subject.

Let P be a generator of group G_1 over an elliptic curve with order q, and G_2 be a multiplicative group of the same order. A bilinear pairing e: $G_1 \times G_1 \rightarrow G_2$ is a mapping with the following properties (Boneh and Franklin 2001):

(1) Identity: For all $P \in G_1$, $e(P, P) = 1$.
(2) Alternation: For all $P_1, P_2 \in G_1$, $e(P_1, P_2) = e(P_2, P_1)$.
(3) Bilinearity: For all $P_1, P_2, P_3 \in G_1, a \in Z_q^*$, $e(aP_1, P_2) = e(P_1, aP_2) = e(P_1, P_2)^a$ and $e(P_1 + P_2, P_3) = e(P_1, P_3)e(P_2, P_3)$.
(4) Non-degeneracy: For all $P_1, P_2 \in G_1$, $P_1 \neq P_2$, $e(P_1, P_2) \neq 1$.
(5) Computability: There exists an efficient algorithm to compute $e(P_1, P_2)$ for all $P_1, P_2 \in G_1$.

In addition, two known computationally infeasible problems used in this study are shown in the following:

(1) Elliptic curve discrete logarithm problem (ECDLP): When given aP, where $a \in Z_q^*$, the ECDLP is how to compute a.
(2) Bilinear computational Diffie–Hellman problem (BCDHP): When given P, aP, bP, and cP, where a, b, and $c \in Z_q^*$, the BCDHP is how to compute $e(P, P)^{abc}$.

2.2. Applications of ID-based cryptosystem using pairing

An ID-based cryptosystem using a user's identity as a public key has the advantage of simple key distribution and management (Shamir 1984). Since Boneh and Franklin first demonstrated a practical ID-based cryptosystem using bilinear pairing, many ID-based pairing applications have been proposed. Such a system typically has a Key Generation Center (KGC) responsible for setting the system parameters and key distribution. To set the system parameters, the KGC chooses $\{G_1, G_2, P, q, e\}$ as the parameters, which have the same definitions as specified in Section 2.1, and defines two hash functions, H:$\{0,1\}^* \rightarrow \{0,1\}^q$ and H_1:$\{0,1\}^* \rightarrow G_1$. The KGC also randomly chooses a secret key, $s \in Z_q^*$, and makes $P_{pub} = sP$ public. For key distribution, when a user having identity ID asks the KGC to establish a public/private key pair over a secure channel, the KGC returns the user's public key as $Q_{ID} = H_1(ID)$ and private key as $S_{ID} = sQ_{ID}$.

One of the interesting properties in such a system is that any two registered users can share a default symmetric key. For example, if user A has a public/private key pair, $\{Q_A = H_1(ID_A), S_A = sQ_A\}$, and user B has $\{Q_B = H_1(ID_B), S_B = sQ_B\}$, then A and B can compute the default shared key, $K_{AB} = e(S_A, Q_B) = e(sQ_A, Q_B) = e(Q_A, Q_B)^s$ and $K_{BA} = e(S_B, Q_A) = e(sQ_B, Q_A) = e(Q_B, Q_A)^s$, respectively. Obviously, $K_{AB} = K_{BA}$.

After describing the concept of an ID-based cryptosystem using pairing, we introduce two of its applications.

(1) *Mutual authentication and session-key agreement*. We next describe the procedure and define explicit and implicit mutual authentications. The procedure is as follows. (i) A chooses a random number, $a \in Z_q^*$, and uses the public key of his counterpart B, Q_B, to compute the session key, $K_{AB} = e(S_A, Q_B)^a = e(sQ_A, Q_B)^a = e(Q_A, Q_B)^{sa}$. A then sends $\{ID_A, aQ_A, E_{KAB}(a)\}$ to B, where $E_{KAB}(a)$ is a symmetric encryption on a using K_{AB}. (ii) Upon receiving A's request message, B uses his private key, S_B, along with the received aQ_A, to compute the session key, $K_{BA} = e(aQ_A, S_B) = e(aQ_A, sQ_B) = e(Q_A, Q_B)^{sa}$. If B can decrypt $E_{KAB}(a)$, and successfully authenticates A's identity by confirming that $aQ_A = a \cdot H_1(ID_A)$, it is then obvious that $K_{BA} = K_{AB}$. B can therefore explicitly authenticate A. Conversely, A has implicitly authenticated B because only the intended B having private key S_B can compute the right K_{BA} to decrypt the request message. This type of mutual authentication is implicit. However, if B returns his identity in an