Süleyman Demirel Üniversitesi
Fen Bilimleri Enstitüsü Dergisi
Cilt 23, Sayı 2, 590-596, 2019

Süleyman Demirel University
Journal of Natural and Applied Sciences
Volume 23, Issue 2, 590-596, 2019

DOI: 10.19113/sdufenbed.514167

An Efficient Electronic Checkbook Scheme with Mutual Authentication

İsa SERTKAYA [*1], **Öznur KALKAR**[2]

[1,2]TÜBİTAK BİLGEM, UEKAE, MCS Labs & BC Labs, 41470, Kocaeli, Türkiye
[1](ORCID:https://orcid.org/0000-0002-4739-0515)
[2](ORCID:https://orcid.org/0000-0002-7875-3892)

(Alınış / Received: 17.01.2019, Kabul / Accepted: 13.07.2019, Online Yayınlanma / Published Online: 30.08.2019)

Keywords
Cryptography,
Security,
E-commerce,
Electronic checkbook,
Electronic check,
Mutual authentication

Abstract: In 1988, Chaum *et al.* introduced the idea of electronic check. Then, Pasupathinathan *et al.* tried to come up with an electronic checkbook scheme. However, their scheme requires signature for each e-check and is not considered as an e-checkbook. Later, three e-checkbook propositions are made by T.H Chen *et al.*, Chang *et al.*, and C.L. Chen *et al.* based on the scheme of W.K. Chen *et al.*. Recently, Sertkaya and Kalkar showed that these three e-checkbook schemes are susceptible to e-check forgery and/or e-check manipulation attacks. They also proposed fixes for these schemes. Nonetheless, fixed versions also carry out drawbacks of the original schemes, like heavy hashing computations, time-synchronization issues, and multiple communication rounds. This study offers an efficient and secure e-checkbook scheme with mutual authentication.

Karşılıklı Kimlik Doğrulaması Sağlayan Etkin Elektronik Çek Defteri Şeması

Anahtar Kelimeler
Kriptografi,
Güvenlik,
E-ticaret,
Elektronik çek defteri,
Elektronik çek,
Karşılıklı kimlik doğrulama

Özet: 1988'de Chaum vd. elektronik çek fikrini ortaya attılar. Ardından, Pasupathinathan vd. elektronik çek defteri çözümü üretmeye çalıştılar. Fakat sistemlerinde her bir çek için bir imza bulunduğu için, tam bir elektronik çek defteri çözümü sayılmaz. Daha sonra T.H. Chen vd., Chang vd. ve C.L. Chen vd., W.K Chen vd.'nin çözümünü geliştiren çek defteri sistemleri önerdiler. Yakın zamanda Sertkaya ve Kalkar önerilen bu sistemlerin, çek sahteciliğine ve manipulasyonlarına karşı dayanıklı olmadıklarını gösterdiler. Ayrıca, bu sistemlerin nasıl düzeltileceğine dair çözümler önerdiler. Bu sistemlerin güvenli versiyonları hala daha eski hallerinin temel problemlerini taşımaktadırlar; örneğin, fazla sayıda özet hesaplama, zaman senkronizasyon problemleri ve çok sayıda iletişim turu. Bu çalışma, karşılıklı kimlik doğrulaması sağlayan verimli ve güvenli bir elektronik çek defteri şeması önermektedir.

1. Introduction

Paper check, or shortly check, is a payment instrument that transfers money from payer's checking account to the receiver when deposited. Basically, there are four actors in a paper check system. Two of them are the payer and the payee, and the other two are their banks which are called issuer and acquirer, respectively. A payer gets a checkbook from an issuer bank, writes a check to a payee. Then the payee deposits this check to her own bank which has the role of acquirer. Finally, the acquirer bank initiates inter-bank transactions with the issuer bank and this results in deduction of desired amount from payer's account into payee's account. This process is described in Figure 1.
Electronic check, shortly e-check, is electronic form of paper check. Same as paper check system, there are four entities: payer, payee, issuer bank, and acquirer bank. Payment process is also similar to paper check system and can be seen in Figure 1.

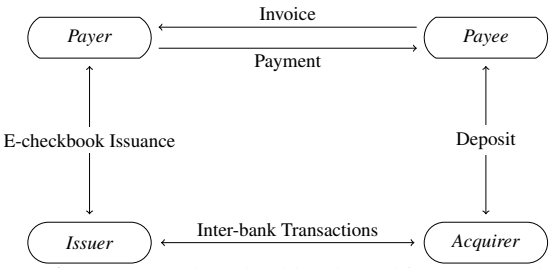

Figure 1. Conceptual E-checkbook Architecture Model

In 2016, 27 trillion USD transferred by checks in USA, according to [3]. In Turkey, 681 billion USD transferred by checks in the same year [3]. Paper check system causes large processing costs and forgery problems. In order to prevent these problems and keep up with digitized world's needs, e-check systems need to be studied. In this study, we propose an efficient and secure e-checkbook mechanism

Corresponding author: isa.sertkaya@tubitak.gov.tr

with mutual authentication of the payer and the payee.

1.1. Related Work

The idea of electronic check is introduced by Chaum, Fiat, and Naor in in [6] in which they also proposed an offline e-check system. However, the proposed scheme has very high computational complexity. An improved version is proposed [7] by Chaum *et al.*, but the amount needs to be determined before e-check issuance. Later, another offline e-check mechanism is given by Brands [4] based on the representation problem and claimed to be more efficient than [6, 7].

An analysis and comparison of initial propositions on electronic payment systems is given by Yu, Hsi, and Kuo [24] and FSTC E-check [2] and SET [22] are among the studied systems. A scheme based on partially blind RSA-based signatures and one-way accumulators is proposed by Kim and Oh [14], but again the amount needs to be determined before e-check issuance. In 2005, W. K. Chen suggested a solution [10] where the amount no longer needs to be determined before e-check issuance, instead it is embedded within e-check.

In traditional checkbook system, the bank issues a checkbook to the customer with multiple leafs. Satisfiability of the same property in the electronic version is an interesting topic. However, aforementioned mechanisms require the payer to interact with the issuer for each e-check issuance, and hence are not e-checkbook mechanisms. To the best of authors' knowledge, the first attempt on e-checkbook scheme is presented by Pasupathinathan, Pieprzyk, and Wang [17]. At the end of issuance phase, the payer gets an e-checkbook with different Schnorr signatures [20] for each e-check. Even though the payer is no longer required to interact with the issuer bank for each e-check payment, computation and storage complexity is linear with respect to the number of e-checks. Achieving constant computational complexity and constant storage is then studied.

Three e-checkbook schemes are proposed based on W.K. Chen's scheme [10] and all three [5, 8, 9] satisfy constant computational complexity and storage property. However, all of them are shown to be not secure [21]. Indeed, T.H. Chen *et al.* [9] is not secure against e-check manipulation and e-check forgery attacks, and Chang *et al.* [5] and C.L. Chen *et al.* [8] are susceptible to e-check manipulation.

Up to our knowledge, there are no new e-checkbook propositions except [21], but research about e-check is ongoing. An e-check scheme that satisfies anonymity and transferability is proposed by Hinarejos *et al.* [12]. Later security enhanced version of 3D-Secure protocol is given by Plateaux *et al.* [18]. Again, these schemes require e-check issuance for each e-check payment.

Finally, Sertkaya and Kalkar proposed secure versions of [5, 8, 9]. However, even though they become secure, they still suffer from the complexity of underlying computations.

1.2. Our Contribution

In this study, we focus on designing an efficient and secure electronic checkbook scheme with mutual authentication. There are four e-checkbook proposals [5, 8, 9, 21] and three of them [5, 8, 9] are already broken [21]. The last proposal fixes these three schemes. However, each of them have drawbacks even though they become secure. [9] contains heavy hashing computations, [5] suffers from time synchronization problems and requires heavy computation for large amounts and [8] involves multiple rounds and requires predefined maximum total face value for e-checkbook. In this study, we propose a secure and efficient e-checkbook scheme with mutual authentication.

1.3. Organization

The rest of this paper is organized as follows. Cryptographic primitives and security notions are defined in Section 2. Proposed scheme is given in Section 3. Security and performance of the proposed mechanism is analyzed in Section 4 and Section 5, respectively. Finally, Section 6 concludes the article.

2. Preliminaries and Security Model

2.1. Cryptographic Primitives

Here, we summarize the cryptographic primitives that are going to be used in the proposed e-checkbook scheme, the definitions are given with respect to the book of [13] as much as possible, for further details please refer to and the references therein.

We use "$a||b$" to denote the ordered concatenation of two strings a then b. $\{0,1\}^*$ and $\{0,1\}^\kappa$ denotes an arbitrary length bit-string and a bit string of length κ, respectively.

Cryptographically Secure Hash: $\mathscr{H} : \{0,1\}^* \to \{0,1\}^\kappa$ is a *pre-image resistant hash function*, and $\mathscr{H}^x(\cdot)$ is used for denoting iteratively computing x-th hash of the given input. For formal definition of pre-image resistance of hash function please refer to [19].

Public-key Encryption: A CCA-secure public-key encryption system consists of four polynomial time algorithms $\mathsf{Pub} = (\mathscr{G}_{\mathsf{pub}}, \mathscr{K}_{\mathsf{pub}}, \mathscr{E}_{\mathsf{pub}}, \mathscr{D}_{\mathsf{pub}})$ described as follows.

- $\mathscr{G}_{\mathsf{pub}}(1^\kappa) \to \mathsf{pp}_{\mathsf{pub}}$. Setup phase takes a security parameter κ as input and outputs public parameters $\mathsf{pp}_{\mathsf{pub}}$ for the encryption scheme.
- $\mathscr{K}_{\mathsf{pub}}(\mathsf{pp}_{\mathsf{pub}}) \to (\mathsf{sk}_{\mathsf{pub}}, \mathsf{pk}_{\mathsf{pub}})$. Given public parameters $\mathsf{pp}_{\mathsf{pub}}$, key generation phase creates a private and public encryption key pair for a user.
- $\mathscr{E}_{\mathsf{pub}}(\mathsf{pk}_{\mathsf{pub}}, m) \to c$. Given a public key $\mathsf{pk}_{\mathsf{pub}}$ and a message m, $\mathscr{E}_{\mathsf{pub}}$ encrypts the message m and outputs ciphertext c.
- $\mathscr{D}_{\mathsf{pub}}(\mathsf{sk}_{\mathsf{pub}}, c) \to m$. Given a private key $\mathsf{sk}_{\mathsf{pub}}$ and a ciphertext c, $\mathscr{D}_{\mathsf{pub}}$ decrypts c to m or outputs $\perp$ if decryption fails.

For indistinguishable under a chosen-ciphertext attack (or is CCA-secure) property of Pub, please refer to [13, Definition 11.8 at page 389].

Digital Signature: $\mathsf{Sig} = (\mathscr{G}_{\mathsf{sig}}, \mathscr{K}_{\mathsf{sig}}, \mathscr{S}_{\mathsf{sig}}, \mathscr{V}_{\mathsf{sig}})$ is a strongly unforgeable digital signature scheme, where each algorithm is given as follows.

- $\mathscr{G}_{\mathsf{sig}}(1^\kappa) \rightarrow \mathsf{pp}_{\mathsf{sig}}$. Setup phase takes a security parameter κ as input and outputs public parameters $\mathsf{pp}_{\mathsf{sig}}$ for the signature scheme.

- $\mathscr{K}_{\mathsf{sig}}(\mathsf{pp}_{\mathsf{sig}}) \rightarrow (\mathsf{sk}_{\mathsf{sig}}, \mathsf{pk}_{\mathsf{sig}})$. Given the public parameters $\mathsf{pp}_{\mathsf{sig}}$, $\mathscr{K}_{\mathsf{sig}}$ algorithm generates a private and public signing key pair for a user.

- $\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}_{\mathsf{sig}}, m, \sigma) \rightarrow b$. Given a public signing key $\mathsf{pk}_{\mathsf{sig}}$, a message m and signature σ, outputs $b = 1$ if the signature σ is valid, otherwise outputs $b = 0$.

For formal definition of strongly existential forgery under chosen message attack, please refer to [1, 11].
In the next section, our scheme will be constructed regarding the definitions given above. In order to realize the scheme concretely, one can refer to [16] for standard $\mathscr{H}$, Pub and Sig choices. Naturally, the secret keys of the involved entities play key role in source authentication, non-repudiation and integrity assurance. Therefore, we assume that protection of these keys in a tamper-proof secure storage is assured, indeed one can refer to [15] for NIST FIBS 140-2 standards.

2.2. Security Notions

In this section, we follow the same notation and definitions given in [21]. As given in Figure 1, an e-checkbook scheme involves four entities:

- **Issuer**: The bank that issues e-checkbook for its registered customers/users, deploys the actual e-check settlement, and makes the money transfers.

- **Payer**: A customer of the Issuer bank, who wants to get an e-checkbook and make payments with e-checks.

- **Payee**: An entity who receives an e-check from a payer. Upon receiving an e-check, payee makes necessary verification and requests the check settlement via her own bank.

- **Acquirer**: The bank who keeps the payee's bank account.

Since the inter-bank transmissions can be handled by the banks by utilizing the existent mechanisms, for simplicity, we are going to assume the Issuer and the Acquirer banks are the same and denote by B. As in the paper-check schemes, we assume that the involved banks are honest and only follow the protocol. On the other hand, the payer and the payee, who are respectively denoted by U and M, will be assumed to be malicious, and thus they would try to circumvent the protocol whenever possible.
Following the above assumptions, any adversary A (a malicious payer, payee or an eavesdropper) may perform the following attacks.

- **E-checkbook forgery**: unauthorized creation of a verifiable e-checkbook, as if it is issued by B,

- **E-check forgery**: unauthorized creation of a verifiable new e-check, as if it is spent by U,

- **E-check manipulation**: manipulation (changing the payee, the amount or the date) of a transmitted e-check,

- **Double spending**: paying with the same indexed e-check more than once, probably with different payee, amount or date,

- **Replay attack**: depositing same e-check more than once.

In order to resist these attacks, any e-checkbook scheme should satisfy the following requirements.

- **E-checkbook validation**: E-checkbook is issued by B for U,

- **E-check validation**: E-check belongs to an e-checkbook issued for U by B,

- **E-check integrity**: E-check has not been manipulated since it was created.

- **Source authentication**: Deposited e-check is created by U.

3. Proposed Scheme

Our proposition consists of four phases; namely Initializing, Issuing, Paying, and Depositing phases. Initializing phase describes the process of public parameter generation, creation of public-private keys of the involved entities. A registered user acquires an e-checkbook by following Issuing phase together with the bank. How a user creates an e-check belonging to her e-checkbook is given in Paying Phase. Finally, the Depositing phase outlines how a payee gets her payment interacting with the bank.
Any e-check possess payee's public signing key, the amount value a, and date value d along with e-checkbook page index i and source authenticator hash chain value $\mathscr{H}^{r-i}(\alpha)$, in encrypted and signed forms. Here, date is not restricted, it can be used as the e-check creation or payment date.

Algorithm 1 Initializing Phase:

Input: security parameter κ

1: B runs $\mathscr{G}_{\mathsf{pub}}(1^\kappa)$ and $\mathscr{K}_{\mathsf{pub}}(\mathsf{pp}_{\mathsf{pub}})$, creates $(\mathsf{sk}^{\mathsf{B}}_{\mathsf{pub}}, \mathsf{pk}^{\mathsf{B}}_{\mathsf{pub}})$,
2: B runs $\mathscr{G}_{\mathsf{sig}}(1^\kappa)$ and $\mathscr{K}_{\mathsf{sig}}(\mathsf{pp}_{\mathsf{sig}})$, creates $(\mathsf{sk}^{\mathsf{B}}_{\mathsf{sig}}, \mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}})$,
3: B publishes $\{\mathscr{H}, \mathsf{pp}_{\mathsf{pub}}, \mathsf{pp}_{\mathsf{sig}}, \mathsf{pk}^{\mathsf{B}}_{\mathsf{pub}}, \mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}\}$ while keeping $\{\mathsf{sk}^{\mathsf{B}}_{\mathsf{pub}}, \mathsf{sk}^{\mathsf{B}}_{\mathsf{sig}}\}$ as secret.
4: Any user U, payer or payee, creates $(\mathsf{sk}^{\mathsf{U}}_{\mathsf{pub}}, \mathsf{pk}^{\mathsf{U}}_{\mathsf{pub}})$ and $(\mathsf{sk}^{\mathsf{U}}_{\mathsf{sig}}, \mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}})$ pairs and publishes $\{\mathsf{pk}^{\mathsf{U}}_{\mathsf{pub}}, \mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}\}$ while keeping $\{\mathsf{sk}^{\mathsf{U}}_{\mathsf{pub}}, \mathsf{sk}^{\mathsf{U}}_{\mathsf{sig}}\}$ as secret.

After the Initializing Phase, a payer U registers with the bank B and acquires an e-checkbook with r checks as follows. Here, public keys of the involved entities are published in an authentic channel.

Algorithm 2 Issuing Phase:

Inputs: B's public keys $\{\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \mathsf{pk}^{\mathsf{B}}_{\mathsf{pub}}\}$ and secret keys $\{\mathsf{sk}^{\mathsf{B}}_{\mathsf{sig}}, \mathsf{sk}^{\mathsf{B}}_{\mathsf{pub}}\}$

1. U first selects $\alpha \leftarrow \{0,1\}^{\kappa}$ uniformly at random,
2. U computes $\beta = \mathscr{E}_{\mathsf{pub}}(\mathsf{pk}^{\mathsf{B}}_{\mathsf{pub}}, \mathscr{H}^{r}(\alpha)||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}})$.
3. U sends $\{\beta, r\}$ to B for signing.
4. Upon receiving $\{\beta, r\}$, B decrypts $\mathscr{H}^{r}(\alpha)||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}} = \mathscr{D}_{\mathsf{pub}}(\mathsf{sk}^{\mathsf{B}}_{\mathsf{pub}}, \beta)$, and signs $\sigma_{\mathsf{U}} = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{r}(\alpha)||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}})$. Then stores $\{\sigma_{\mathsf{U}}, r\}$ and sends σ_{U} back to U,
5. U checks $\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{r}(\alpha)||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}, \sigma_{\mathsf{U}}) \stackrel{?}{=} 1$ holds, if so $(\alpha, \mathscr{H}^{r}(\alpha), \sigma_{\mathsf{U}}, r)$-tuple is kept secret.

After successfully generating an e-checkbook $(\alpha, \mathscr{H}^{r}(\alpha), \sigma_{\mathsf{U}}, r)$,assume that U has used $i-1$ $(i < r)$, checks and wants to attach a face value a, and date d, for the payee M to the i-th e-check. Then U follows the Paying Phase.

Algorithm 3 Paying Phase:

Inputs: $(\alpha, \mathscr{H}^{r}(\alpha), \sigma_{\mathsf{U}}, r)$, i, a, d

1. U computes

$$\beta^{i}_{\mathsf{UM}} = \mathscr{E}_{\mathsf{pub}}(\mathsf{pk}^{\mathsf{M}}_{\mathsf{pub}}, i||\mathscr{H}^{r-i}(\alpha)||a||d),$$
$$\sigma^{i}_{\mathsf{UM}} = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}^{\mathsf{U}}_{\mathsf{sig}}, i||\mathscr{H}^{r-i}(\alpha)||a||d||\mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}}).$$

2. U sends the tuple $(\beta^{i}_{\mathsf{UM}}, \sigma^{i}_{\mathsf{UM}}, \sigma_{\mathsf{U}})$ to M as the face-value attached e-check.

Algorithm 4 Depositing Phase:

Inputs: B's public keys $\{\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \mathsf{pk}^{\mathsf{B}}_{\mathsf{pub}}\}$ and secret keys $\{\mathsf{sk}^{\mathsf{B}}_{\mathsf{sig}}, \mathsf{sk}^{\mathsf{B}}_{\mathsf{pub}}\}$, received e-check $(\beta^{i}_{\mathsf{UM}}, \sigma^{i}_{\mathsf{UM}}, \sigma_{\mathsf{U}})$.

1. M decrypts $i||\mathscr{H}^{r-i}(\alpha)||a||d = \mathscr{D}_{\mathsf{pub}}(\mathsf{sk}^{\mathsf{M}}_{\mathsf{pub}}, \beta^{i}_{\mathsf{UM}})$ with her secret key $\mathsf{sk}^{\mathsf{M}}_{\mathsf{pub}}$,
2. M verifies the signatures as follows.

$$\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}, i||\mathscr{H}^{r-i}(\alpha)||a||d||\mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}}, \sigma^{i}_{\mathsf{UM}}) \stackrel{?}{=} 1\,,$$
$$\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{i}(\mathscr{H}^{r-i}(\alpha))||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}, \sigma_{\mathsf{U}}) \stackrel{?}{=} 1\,.$$

3. M computes

$$\beta^{i}_{\mathsf{MB}} = \mathscr{E}_{\mathsf{pub}}(\mathsf{pk}^{\mathsf{B}}_{\mathsf{pub}}, i||\mathscr{H}^{r-i}(\alpha)||a||d),$$
$$\sigma^{i}_{\mathsf{MB}} = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}^{\mathsf{M}}_{\mathsf{sig}}, i||\mathscr{H}^{r-i}(\alpha)||a||d||\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}).$$

4. M sends the tuple $(\beta^{i}_{\mathsf{MB}}, \sigma^{i}_{\mathsf{MB}}, \sigma^{i}_{\mathsf{UM}}, \sigma_{\mathsf{U}})$ to B.
5. B decrypts $i||\mathscr{H}^{r-i}(\alpha)||a||d = \mathscr{D}_{\mathsf{pub}}(\mathsf{sk}^{\mathsf{B}}_{\mathsf{pub}}, \beta^{i}_{\mathsf{MB}})$ with her secret key $\mathsf{sk}^{\mathsf{B}}_{\mathsf{pub}}$,
6. B first assure that $(i, \mathscr{H}^{r-i}(\alpha))$ was not already spent, then verifies

$$\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}}, i||\mathscr{H}^{r-i}(\alpha)||a||d||\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \sigma^{i}_{\mathsf{MB}}) \stackrel{?}{=} 1\,,$$
$$\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}, i||\mathscr{H}^{r-i}(\alpha)||a||d||\mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}}, \sigma^{i}_{\mathsf{UM}}) \stackrel{?}{=} 1\,,$$
$$\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{i}(\mathscr{H}^{r-i}(\alpha))||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}, \sigma_{\mathsf{U}}) \stackrel{?}{=} 1\,.$$

7. After verifying all the signatures, B records $(i, \mathscr{H}^{r-i}(\alpha), a, d, \mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}})$, deducts the amount a from U's account, adds it into the M's account b, and informs M.

Whenever a payee M receives an e-check, she first verify that the e-check belongs to an e-checkbook issued by B for U and is cretead and signed with the desired amount, date, and then send it to the bank B for depositing and double-spending control.

4. Security Analysis

This section discusses security properties of the proposed scheme.

4.1. Correctness

At the Issuing Phase (Algorithm 2), the checkbook owner U checks if $\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{r}(\alpha)||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}, \sigma_{\mathsf{U}}) = 1$. Due to well-definedness of the digital signature scheme and B honestly following the protocol with $\sigma_{\mathsf{U}} = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{r}(\alpha)||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}})$, we always have

$$\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{r}(\alpha)||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}, \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{r}(\alpha)||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}})) = 1.$$

During the Depositing Phase (Algorithm 4), the same signature is also verified by M and B. Additionally, σ_{UM} is checked by both M and B while σ_{MB} is checked by only B. Whenever M and B verify these signatures, the followings hold accordingly.

$$\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}}, m||\mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}}, \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}^{\mathsf{U}}_{\mathsf{sig}}, m||\mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}})) = 1,$$
$$\mathscr{V}_{\mathsf{sig}}(\mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}}, m||\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}^{\mathsf{M}}_{\mathsf{sig}}, m||\mathsf{pk}^{\mathsf{B}}_{\mathsf{sig}})) = 1,$$

where $m = i||\mathscr{H}^{r-i}(\alpha)||a||d$.

4.2. Mutual authentication

At the Issuing phase, U verifies the signature σ_{U} of B which authenticates B. Note that U keeps α secret. This together with pre-image resistance property of $\mathscr{H}$ and U's signature authenticates the source of the i-th e-check. At the Paying Phase, U encrypts "$i||\mathscr{H}^{r-i}(\alpha)||a||d$" for the payee M and signs "$i||\mathscr{H}^{r-i}(\alpha)||a||d||\mathsf{pk}^{\mathsf{M}}_{\mathsf{sig}}$" with her secret signing key. Only M, with her secret key $\mathsf{sk}^{\mathsf{M}}_{\mathsf{pub}}$, can decrypt β^{i}_{UM}. Additionally, by verifying the signatures σ^{i}_{UM} and σ_{U}, M authenticates U and B, respectively. At the Depositing Phase, only B can decrypt β^{i}_{MB}, and furthermore B can authenticate both U and M by verifying the signatures σ^{i}_{UM} and σ^{i}_{MB}.

4.3. E-checkbook unforgeability

Suppose that an adversary A is able to forge an e-checkbook as $(\alpha', \mathscr{H}^{r}(\alpha'), \sigma'_{\mathsf{U}}, r')$-tuple for a victim U. Since U's $\{\mathsf{sk}^{\mathsf{U}}_{\mathsf{pub}}, \mathsf{sk}^{\mathsf{U}}_{\mathsf{sig}}\}$ keys are kept secret, A can manage to forge the e-checkbook only if she can forge B's signature as $\sigma'_{\mathsf{U}} = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}^{\mathsf{B}}_{\mathsf{sig}}, \mathscr{H}^{r}(\alpha')||\mathsf{pk}^{\mathsf{U}}_{\mathsf{sig}})$, which contradicts with the strongly existential unforgeability under chosen message attack property of the digital signature scheme.

4.4. E-check unforgeability

Suppose that an adversary A has gathered all the e-checks $(\beta^{j}_{\mathsf{UMj}}, \sigma^{j}_{\mathsf{UMj}}, \sigma_{\mathsf{U}})$ of U up to i-th e-check. Since no information is sent unencrypted within the e-check tuple,

only a malicious payee can get further insight about the intercepted e-check. In order to cover all adversaries, we assume that A has the knowledge of $(j, \mathscr{H}^{r-j}(\alpha), a^j, d^j)$ for some $1 \leq j \leq i < r$. In order to successfully forge the $i+1$-th e-check, A needs to compute

$$\beta_{\mathsf{UM}}^{i+1} = \mathscr{E}_{\mathsf{pub}}(\mathsf{pk}_{\mathsf{pub}}^{\mathsf{M}}, i+1 || \mathscr{H}^{r-(i+1)}(\alpha) || a || d),$$
$$\sigma_{\mathsf{UM}}^{i+1} = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}_{\mathsf{sig}}^{\mathsf{U}}, i+1 || \mathscr{H}^{r-(i+1)}(\alpha) || a || d || \mathsf{pk}_{\mathsf{sig}}^{\mathsf{M}}),$$

for some payee M, amount a and date d. Here A faces two challenges. First, she should be able compute $\mathscr{H}^{r-(i+1)}(\alpha)$ from $\mathscr{H}^{r-j}(\alpha)$ for some $1 \leq j \leq i < r$. Second she should forge signature for $(i+1, \mathscr{H}^{r-(i+1)}(\alpha), a, d)$ on behalf of U. Former contradicts with the pre-image resistance of the cryptographically secure hash function $\mathscr{H}$, latter contradicts with the strongly existential unforgeability under chosen message attack property of the digital signature scheme Sig. Hence, we can conclude the scheme is secure against e-check forgeability attacks.

4.5. E-check manipulation resistance

E-check manipulation can be modeled in two different cases. In the first case, say **E-check manipulation Type-I**, an adversary A gathers the information sent between U, M and B and blocks if necessary. Then A tries to change the payee or the amount attached to the e-check. In the second case (**E-check manipulation Type-II**), the payee M behaves maliciously and tries to change the amount a and/or date d. We are going to analyze these two cases independently.

E-check manipulation Type-I. Assume that an adversary A, other than the payee M, intercepts a transmission of the i-th e-check $(\beta_{\mathsf{UM}}^i, \sigma_{\mathsf{UM}}^i, \sigma_{\mathsf{U}})$ for some $1 \leq i \leq r$ that is sent from the payer U to the payee M and blocks transmission of $(\beta_{\mathsf{MB}}^i, \sigma_{\mathsf{MB}}^i, \sigma_{\mathsf{UM}}^i, \sigma_{\mathsf{U}})$ from M to B. A can manage to manipulate the i-th e-check only if she can construct

$$\beta_{\mathsf{UM}'}^i = \mathscr{E}_{\mathsf{pub}}(\mathsf{pk}_{\mathsf{pub}}^{\mathsf{M}'}, i || \mathscr{H}^{r-i}(\alpha) || a' || d),$$
$$\sigma_{\mathsf{UM}'}^i = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}_{\mathsf{sig}}^{\mathsf{U}}, i || \mathscr{H}^{r-i}(\alpha) || a' || d || \mathsf{pk}_{\mathsf{sig}}^{\mathsf{M}'}),$$
$$\beta_{\mathsf{M}'\mathsf{B}}^i = \mathscr{E}_{\mathsf{pub}}(\mathsf{pk}_{\mathsf{pub}}^{\mathsf{B}}, i || \mathscr{H}^{r-i}(\alpha) || a' || d),$$
$$\sigma_{\mathsf{M}'\mathsf{B}}^i = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}_{\mathsf{sig}}^{\mathsf{M}'}, i || \mathscr{H}^{r-i}(\alpha) || a' || d || \mathsf{pk}_{\mathsf{sig}}^{\mathsf{B}}).$$

In order to learn $\mathscr{H}^{r-i}(\alpha)$ value, A needs to either break public-key encryption scheme Pub or compute a pre-image from some formerly known $\mathscr{H}^{r-j}(\alpha)$ with $1 \leq j < i \leq r$. However, these two cases contradict with the security assumptions of Pub and $\mathscr{H}$, respectively. Even if A has the knowledge of $(i, \mathscr{H}^{r-i}(\alpha), a, d)$, she needs to forge $\sigma_{\mathsf{UM}'}^i$ and $\sigma_{\mathsf{M}'\mathsf{B}}^i$, which also contradicts with the security assumption of the digital signature scheme Sig.

E-check manipulation Type-II. Suppose a malicious payee M aims to perform e-check manipulation by trying to change the amount a to a', d to d', or both. She knows i-th e-check $(\beta_{\mathsf{UM}}^i, \sigma_{\mathsf{UM}}^i, \sigma_{\mathsf{U}})$ for some $1 \leq i \leq r$ that is sent from the payer U to the payee M. She can compute

$$\beta'_{\mathsf{UM}} = \mathscr{E}_{\mathsf{pub}}(\mathsf{pk}_{\mathsf{pub}}^{\mathsf{M}}, i || \mathscr{H}^{r-i}(\alpha) || a' || d'),$$
$$\beta'_{\mathsf{MB}} = \mathscr{E}_{\mathsf{pub}}(\mathsf{pk}_{\mathsf{pub}}^{\mathsf{B}}, i || \mathscr{H}^{r-i}(\alpha) || a' || d'),$$
$$\sigma'_{\mathsf{MB}} = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}_{\mathsf{sig}}^{\mathsf{M}}, i || \mathscr{H}^{r-i}(\alpha) || a' || d' || \mathsf{pk}_{\mathsf{sig}}^{\mathsf{B}}),$$

and lets $\beta_{\mathsf{MB}}^i := \beta'_{\mathsf{MB}}$ and $\sigma_{\mathsf{MB}}^i := \sigma'_{\mathsf{MB}}$. However, in the Depositing Phase (at Step 4 of Algorithm 4), B expects $(\beta_{\mathsf{MB}}^i, \sigma_{\mathsf{MB}}^i, \sigma_{\mathsf{UM}}^i, \sigma_{\mathsf{U}})$ from M. Based on the strongly existential unforgeability under chosen message attack property of digital signature Sig, M cannot forge

$$\sigma'_{\mathsf{UM}} = \mathscr{S}_{\mathsf{sig}}(\mathsf{sk}_{\mathsf{sig}}^{\mathsf{U}}, i || \mathscr{H}^{r-i}(\alpha) || a' || d' || \mathsf{pk}_{\mathsf{sig}}^{\mathsf{M}}).$$

On the other hand, if M tries to send modified tuple $(\beta'_{\mathsf{MB}}, \sigma'_{\mathsf{MB}}, \sigma_{\mathsf{UM}}^i, \sigma_{\mathsf{U}})$ to B for e-check settlement; naturally, B finds out that the verification of the σ_{UM}^i fails and therefore rejects the depositing request.

Hence in both cases, provided that the security assumptions for underlying cryptographic primitives hold, the proposed scheme is resistant against e-check manipulation attacks.

4.6. Double-spending resistance

At Depositing Phase (Algorithm 4), B first checks if the e-check is already recorded and records $(i, \mathscr{H}^{r-i}(\alpha), a, d, \mathsf{pk}_{\mathsf{sig}}^{\mathsf{M}})$ as spent i-th e-check before transferring the amount from payer's account to the payee's account. B acts by first come first served rule. Hence, if a malicious payer U creates two different e-check as the i-th e-check, only the first received one will be settled and the second will be rejected. Thus, the proposed scheme is secure against double-spending problem.

4.7. Replay attack resistance

Similar arguments apply here. B assures the e-check is not already recorded and records $(i, \mathscr{H}^{r-i}(\alpha), a, d, \mathsf{pk}_{\mathsf{sig}}^{\mathsf{M}})$ as spent i-th e-check before transferring the amount from payer's account to the payee's account. If any adversary tries to re-send an e-check tuple to B, B accepts the first one and rejects the other. So, same as double-spending attack, replay attack is prevented.

5. Performance

W.K. Chen proposed an e-check scheme which requires bank's signature prior to each e-check payment, [10]. This scheme ensures integrity of face value a and payee's bank account number b of the e-check by computing

$$\beta_1 = H^a(x_1), \quad \beta_2 = H^{w-a}(x_2),$$
$$\beta_3 = H^b(x_3), \quad \beta_4 = H^{k-b}(x_4),$$

where x_i's are uniformly random secret values of the payer, H is a hash function, w is maximum possible face value and k is maximum possible account number.

Well, this may seem as an elegant way of e-check integrity assurance against e-check manipulation, however its practicality and efficiency is questionable. Based on the w and k values, the protocol at least requires $2(w+k)$ hash computations.

There is no world-wide accepted standard for bank account numbers, but we can consider International Bank Account Number (IBAN) as an instance. IBAN is adopted by the European Committee for Banking Standards (ECBS) and international standard under ISO 13616:2007. An IBAN consists of up to 34 alphanumeric characters including

a two letter country code, two check digits, and up to 30 alphanumeric characters that include domestic bank account number, branch identifier, and potential routing information, [23].

If we assume that W.K. Chen's protocol will be used with IBAN (maximum possible account number k would be over 2^{70} if IBAN set to be 24 characters long) and set an upper limit on maximum possible face value, $w \leq 2^{48}$, with a rough calculation, for each e-check payment one needs $2(2^{48}+2^{70})$ hash computations. Obviously, this computation cost does not lie within efficiency boundaries.

T.H. Chen *et al.*'s scheme also requires the same hash computations, [9]. Later Chang *et al.* modified W.K. Chen's scheme, but their scheme also requires similar hash computations based on w value, [5]. Recently, Sertkaya and Kalkar has given the secure versions of T.H. Chen *et al.* and Chang *et al.* schemes, [21]. These secure versions still suffer from the same hash computation burden.

C.L. Chen *et al.* proposed an improvement on Chang *et al.*'s scheme, [8], which was recently broken in [21]. Secure version of their scheme requires pre-defined maximum total face value for an e-checkbook, that is updated whenever an e-check payment is finalized. This scheme involves multiple communication rounds over both unsecured and secured channel. Furthermore, the payee learns remaining balance w_{new}, which should not be the case.

Apart from the above proposals, Pasupathinathan *et al.* proposed a scheme that supports multiple e-check issuance at once, [17]. However, at the Issuance phase, the issuing bank creates signature for each e-check individually.

Here, our proposed scheme requires only one signature of the issuing bank for an e-checkbook with r e-checks. Up to our knowledge, there is no fixed r value for paper-based checkbooks; there are examples with $10, 15, \ldots$ up to 200 checks. Based on this assumption, computation of the hash chain $\mathscr{H}^r(\alpha)$ value would be a reasonable cost. In fact, compared to the above hash computation costs, this cost is negligible. Our new scheme only assumes sharing of public keys in an authentically secure channel, all remaining communications can be carried under unsecured communication channel. Furthermore, this scheme mutually authenticates involved entities by utilizing their cryptographic keys.

6. Conclusion

Recently, Sertkaya and Kalkar showed that there exist security vulnerabilities in the previously proposed e-checkbook schemes and illustrated how to fix these schemes. In this work, we showed these scheme are not efficient as intended due to heavy hash computation, time-synchronization, or multiple-round communication requirements. Thus, we propose a new scheme that is more efficient and supports source authentication of the e-check and mutual authentication of the payer and the payee. We give details of the new e-checkbook scheme, discuss its security, show that it is resistant against known attacks, and finally compare its performance with the previously proposed secure schemes.

Acknowledgement

We would like to thank to the anonymous reviewers for their careful reading, insightful comments and suggestions for our manuscript.

References

[1] Jee Hea An, Yevgeniy Dodis, and Tal Rabin. On the security of joint signature and encryption. In *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques: Advances in Cryptology*, EUROCRYPT '02, pages 83–107, London, UK, UK, 2002. Springer-Verlag.

[2] Milton M. Anderson. The Electronic Check Architecture. Technical report, Financial Services Technology Consortium, 1998.

[3] Bank for International Settlements. Statistics on payment, clearing and settlement systems in the CPMI countries, 2017.

[4] Stefan Brands. An Efficient Off-line Electronic Cash System Based On The Representation Problem. Technical report, Centrum Wiskunde & Informatica (CWI), 1993.

[5] Chin-Chen Chang, Shih-Chang Chang, and Jung-San Lee. An on-line electronic check system with mutual authentication. *Computers & Electrical Engineering*, 35(5):757 – 763, 2009.

[6] David Chaum, Bert den Boer, Eugène van Heyst, Stig Mjølsnes, and Adri Steenbeek. Efficient offline electronic checks. In Jean-Jacques Quisquater and Joos Vandewalle, editors, *Advances in Cryptology — EUROCRYPT '89*, pages 294–301, Berlin, Heidelberg, 1990. Springer Berlin Heidelberg.

[7] David Chaum, Amos Fiat, and Moni Naor. Untraceable electronic cash. In Shafi Goldwasser, editor, *Advances in Cryptology — CRYPTO' 88*, pages 319–327, New York, NY, 1990. Springer New York.

[8] Chin-Ling Chen, Cheng-Hsiung Wu, and Wei-Chech Lin. Improving an on-line electronic check system with mutual authentication. In *Proceedings of International Conference on Advanced Information Technologies (AIT 2010)*, 2010.

[9] Tzung-Her Chen, Shu-Chen Yeh, Kuan-Chieh Liao, and Wei-Bin Lee. A practical and efficient electronic checkbook. *Journal of Organizational Computing and Electronic Commerce*, 19(4):285–293, 2009.

[10] Wei-Kuei Chen. Efficient on-line electronic checks. *Applied Mathematics and Computation*, 162(3):1259 – 1263, 2005.

[11] Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Comput.*, 17(2):281–308, April 1988.

[12] M. F. Hinarejos, J. Ferrer-Gomila, G. Draper-Gil, and L. Huguet-Rotger. Anonymity and transferability for an electronic bank check scheme. In *2012 IEEE*

11th International Conference on Trust, Security and Privacy in Computing and Communications, pages 427–435, June 2012.

[13] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography*. Chapman & Hall/CRC, 2nd edition, 2014.

[14] S. Kim and H. Oh. A new electronic check system with reusable refunds. *International Journal of Information Security*, 1(3):175–188, Nov 2002.

[15] Security Requirements For Cryptographic Modules. Federal Information Processing Standards Publication (FIPS) 140-2, National Institute for Standards and Technology, Gaithersburg, MD 20899-8900, USA, 2001.

[16] Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms. NIST Special Publication (SP) 800-175B, National Institute for Standards and Technology, Gaithersburg, MD 20899-8900, USA, 2016.

[17] Vijayakrishnan Pasupathinathan, Josef Pieprzyk, and Huaxiong Wang. Privacy enhanced electronic cheque system. In *Seventh IEEE International Conference on E-Commerce Technology (CEC'05)*, pages 431–434, July 2005.

[18] Aude Plateaux, Patrick Lacharme, Vincent Coquet, Sylvain Vernois, Kumar Murty, and Christophe Rosenberger. An e-payment architecture ensuring a high level of privacy protection. In Tanveer Zia, Albert Zomaya, Vijay Varadharajan, and Morley Mao, editors, *Security and Privacy in Communication Networks*, pages 305–322, Cham, 2013. Springer International Publishing.

[19] Phillip Rogaway and Thomas Shrimpton. Cryptographic hash-function basics: Definitions, implications and separations for preimage resistance, second-preimage resistance, and collision resistance. Cryptology ePrint Archive, Report 2004/035, 2004.

[20] C. P. Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology*, 4(3):161–174, Jan 1991.

[21] Isa Sertkaya and Oznur Kalkar. Forgery Attacks on Some Electronic Checkbook Schemes. submitted 2018.

[22] Visa and MasterCard. SET Secure Electronic Transaction Specification Book 1, 1997.

[23] Wikipedia. International Bank Account Number — Wikipedia, the free encyclopedia, 2019. [Online; accessed 09-January-2019].

[24] Hsiao-Cheng Yu, Kuo-Hua Hsi, and Pei-Jen Kuo. Electronic payment systems: an analysis and comparison of types. *Technology in Society*, 24(3):331 – 347, 2002.