symmetry

MDPI

Article

Untraceable and Anonymous Mobile Payment Scheme Based on Near Field Communication

Raylin Tso

Department of Computer Science, National Chengchi University, No. 64, Sec. 2, Zhi-nan Rd., Taipei 11605, Taiwan; raylin@cs.nccu.edu.tw; Tel.: +886-981-435-201

Received: 9 November 2018; Accepted: 23 November 2018; Published: 1 December 2018

Abstract: With the developments of mobile communications, M-commerce has become increasingly popular in recent years. However, most M-commerce schemes ignore user anonymity during online transactions. As a result, user transactions may easily be traced by shops, banks or by Internet Service Providers (ISPs). To deal with this problem, we introduce a new anonymous mobile payment scheme in this paper. Our new scheme has the following features: (1) Password-based authentication: authentication of users is done by low-entropy password; (2) Convenience: the new scheme is designed based on near field communication (NFC)-enabled devices and is compatible with EuroPay, MasterCard and Visa (EMV-compatible); (3) Efficiency: users do not need to have their own public/private key pairs and confidentiality is achieved via symmetric-key cryptography; (4) Anonymity: users use virtual accounts in the online shopping processes, thereby preventing attackers from obtaining user information even if the transaction is eavesdropped; (5) Untraceablity: no one (even the bank, Trusted Service Manager (TSM), or the shop) can trace a transaction and link the real identity with the buyer of a transaction; (6) Confidentiality and authenticity: all the transaction is either encrypted or signed by the sender so our new scheme can provide confidentiality and authenticity. We also present the performance and the security comparison of our scheme with other schemes. The results show that our scheme is applicable and has the most remarkable features among the existing schemes.

Keywords: anonymity; EMV-compatible; mobile payment; NFC; security

1. Introduction

Mobile commerce (M-commerce) [1–4] has come into the limelight in recent years thanks to the universality of smartphones and the rapid developments of wireless and mobile communication technologies. By using M-commerce, a user can use an internet-enabled mobile device such as a smartphone or a tablet and then he/she can perform online activities. Possible online activities include online shopping, online auctions, online payments, etc.

M-commerce is convenient and attractive to both users and merchants. For users, M-commerce is convenient and simple compared to traditional transactions. Moreover, it brings possible customers from all over the world from a merchant's viewpoint. Furthermore, merchants can collect customer behaviors via transactions and can make statistical analysis to speculate the consumption preferences of users. The analysis helps a merchant to send subsequent information that may be attractive to users in the future to push up sales.

1.1. Related Works

Due to the importance of M-commerce, many online payment schemes have been proposed [1,5–10]. Some of them are designed to increase the performance and some are designed to enhance the security or privacy of transactions. For example, Toorani et al. [11] proposed a secure short message service

payment protocol. The scheme allows users to pay the bill by the short message service (SMS). However, some weaknesses have been found including replay-attack and SMS message forging attack. Molloy et al. [12] proposed a payment protocol using a virtual credit card instead of a real card. In addition, the virtual credit card can be generated as many times as a user wishes. On the other hands, to provide user anonymity and unlinkability, Martínez-Peláez et al. [13] proposed a micropayment protocol basing on the anonymous electronic cash. To increase the performance of online payment protocol, Kungpisdan et al. [9] proposed an account-based online payment protocol. The protocol adopted symmetric-key cryptography instead of public-key cryptography for achieving confidentiality. Compared to many schemes using public-key cryptographies, the scheme achieves low computation during tractions. Liao [14] proposed a cross-domain anonymous online payment scheme which allows users to consume in different merchants in mobile communication with user anonymity. On the other hand, near field communication (NFC) [15–17] has come into limelight in recent years and has become increasingly popular. For this reason, NFC-enabled mobile payment protocols are provided [18–21] in which credit card information is combined into NFC-enabled mobile devices. The NFC-chip embed into a smartphone will change itself to the card simulation mode to simulate a credit card when an online transaction is proceeded and the information inside the (simulated) card is requested. In this way the card information is transmitted securely via NFC standard protocol to the merchant or to the card issuer for authentication. In practice, Apple [22], Microsoft [23], and Google Inc. [24] have introduced their idea separately to replace the traditional credit card by a virtual credit card. In addition, the cards are stored in NFC-enabled smartphones. By using the smartphone, mobile payments, online transactions can be proceeded very efficiently and conveniently. For this reason, IT industries and many researches have continuously focusing on such a promising technology to continuously improve its security, performance and/or to add new features on it [25–29]. For example, Pasquet et al. [28] proposed an infrastructure to test the security of the simulated credit card in the NFC-enabled smartphone. Pailles et al. [27] focus on the protection of private data in user accounts. Mainetti et al. [26] proposed a protocol for message-exchange between the NFC-enabled smartphone and the Point of Sale (PoS) terminal using a peer to peer method . The advantage of the scheme in [26] is that the transaction confirmation message can be stored and customized by merchant. Finally, Urien and Piramuthu [29] assumes that a user's NFC-enabled smartphone may be untrustworthy and, instead of using the built-in security element in NFC-enabled smartphones, they proposed the cloud security element to achieve the goal . Moreover, their scheme follows the EMV standard and can execute the EMV credit card protocol. Their concept is similar to the Host Card Emulation [25] technique.

1.2. Motivation

Anonymity is an important issue for mobile payment and online transactions from customers' viewpoints. In general, speaking, the identity of a user is required to be presented to its counterparty who may be a card issuer, a merchant, or a Trusted Service Manager (TSM) during the process of an online payment. The identity presented here is used for authentication. However, this may leak the information on who the owner of the card holder is and/or by whom and where the goods or items have been bought, and from which merchant. Furthermore, with this information, an impersonation-attack may be launched to forge an invalid transaction. To deal with this problem, Luo et al. [30] in 2016 proposed an NFC-based mobile payment protocol with user anonymity. However, we found that their scheme has some security issues and may not be functional in practice. For example, they use the same private key for digital-signature signing and for ciphertext decryption. Unlinkability is also a problem. It is difficult to be achieved according to the definition of *unlinkability*. Furthermore, Lee et al. [31], also mentioned that Luo et al.'s idea suffers from the symmetric-key leakage problem. Although Lee et al. [31] introduced their remedy, the new scheme is designed for pre-paid system but not for credit card applications. In addition, the new scheme is not EMV-compatible.

1.3. Our Contributions

In this paper, we are going to introduce a new NFC-based mobile payment protocol. The new scheme has the following features:

- **Password-based authentication:** Password-based authentication does not require expensive infrastructure compared to digital-signature and biometrics-based authentications. In addition, it is convenient for users since users can use low-entropy and easy-to-remember password to establish a high-entropy session key. Consequently, in our protocol, we assume that a user of our scheme possesses no high-entropy secret key and no public/private key pair in advance. What the user has in advance is only a low-entropy password (pw) shared with a bank (card issuer). The pw will then be used for user-authentication and for securing communication.
- **Convenience:** The new NFC-based anonymous mobile payment scheme is compatible with EMV standard [32–34] which is set up by the largest three credit card magnate corporations named Europay, MasterCard and Visa in 1993, and it also can be operated on any NFC function-enabled smartphones.
- **Efficiency:** Users of our scheme do not need to have their own public/private key pairs and confidentiality is achieved via symmetric-key cryptography.
- **Anonymity:** A user's virtual account is all set up and registered via the bank. Except the bank, no one else will know the actual identity of the user even when eavesdropping is occurred during the transactions.
- **Untraceablity:** No one (even the bank, TSM or the shop) can trace a transaction and link the real identity with the buyer of a transaction.
- **Confidentiality and Authenticity:** Every communication for transactions is either encrypted by a session key from Diffie-Hellman key exchange [35] or by a pre-shared key between a bank and TSM.

1.4. Paper Organization

The paper is organized as follows: Section 2 presents some preliminaries required for our construction as well as Luo et al.'s scheme. The model and our new NFC-based anonymous mobile payment protocol is provided in Section 3. We provide the security of our protocol in Section 4 and the conclusion is given in Section 5.

2. Preliminaries and Luo et al.'s Scheme Revisited

This section reviews some cryptographic primitives and definitions required for our construction. We will also review Lou et al.'s work and discuss the security flaws of their scheme.

2.1. Security Assumptions

Definition 1. Discrete Logarithm (DL) Problem: *G is a cyclic group with prime order p. g is a primitive root of G. The DL problem to the base g means the following problem:*

$$\text{Given } g, h \in G, \text{ find an integer } x \text{ such that } h = g^x \bmod p.$$

The DL problem is believed to be difficult and to be the hard direction of a one-way function. Based on the DL problem, Computational Diffie-Hellman Assumption can be defined as follows:

Definition 2. Computational Diffie-Hellman (CDH) Problem: *G is a cyclic group of prime order p and g is a primitive root of G, the CDH assumption says that given (g, g^a, g^b) for $a, b \in Z_p^*$ picked randomly, there exists no polynomial-time algorithm to find an element $C \in G$ such that $C = g^{ab} \bmod p$ with non-negligible probability.*

CDH problem is also believed to be difficult and based on the problem, any two entities can generate $C = g^{ab} \bmod p$ as their session key which is called the Diffie-Hellman (key-exchange)-based key [35].

2.2. Luo et al.'s Scheme Revisited

We first review Luo et al.'s protocol (UAPS for short) in this subsection and discuss the security flaws of their scheme. There are four entities in UAPS: a secure element (SE) embedded in a smart phone, a user, a card issuer (i.e., the bank) and a virtual credit card issuer (i.e., the TSM). In addition, the protocol consists of four stages: registration stage, anonymous virtual bank account generation stage, anonymous transaction account generation and issuing of virtual credit card stages. Details are described as follows:

2.2.1. Registration Stage

In this phase, each entity must generate its own identity ID and an asymmetric key pair (PK_{ID}, SK_{ID}) with a certificate issued by a Certificate Authority (CA). At the beginning, a user with identity ID_U must open a bank account and to register his NFC-enabled smartphone to the bank. The bank then generates a shared key $K_{B,U}$ between the bank and the user and returns it to the user.

2.2.2. Anonymous Virtual Bank Account Generation Stage

At this stage, the user with identity ID_U requests the bank to establish a virtual account AID_i for him/her. The SE in the user's NFC-enabled smartphone will generate a public/private key pair (PK_{AID_i}, SK_{AID_i}), and then uses the private key SK_{AID_i} to sign the public key PK_{AID_i}. Then it delivers the signature to the bank. After authenticating the identity of the user, the bank will issue the corresponding certification of AID_i to user. Figure 1 shows the communication flaws and the detailed descriptions are listed as follows:

1. The user sends $ID_U \parallel E_{K_{B,U}}(ID_U \parallel N_1 \parallel Sign_{SK_U}(ID_U \parallel N_1))$ to the bank. Here N_1 is a nonce, $Sign_{SK_U}(M)$ denotes the signature on message M signed by the signing key SK_U, and $E_K(M)$ denotes the ciphertext of message M encrypted by the key K.
2. The bank decrypts the message with the share key and verifies the signature. If it passed, then the bank generates a virtual account AID_i and a nonce N_2. The bank then sends back $ID_B \parallel E_{K_{B,U}}(ID_U \parallel AID_i \parallel N_2)$ back to the user.
3. The user receives AID_i and stores $ID_U \parallel AID_i \parallel N_2$ into the SE.
4. The SE generates a key pair (PK_{AID_i}, SK_{AID_i}) corresponding to AID_i. SK_{AID_i} is stored in the SE and the SE returns $PK_{AID_i} \parallel Sig_{SK_{AID_i}}(ID_U \parallel AID_i \parallel PK_{AID_i} \parallel N_2)$ to the user.
5. The user sends $ID_U \parallel E_{K_{B,U}}(Sig_{SK_U}(ID_U \parallel AID_i \parallel N_2 \parallel Sig_{SK_{AID_i}}(ID_U \parallel AID_i \parallel PK_{AID_i} \parallel N_2)$ to the bank.
6. The bank decrypts the message and gets PK_{AID_i}. It will then create a certificate $CERT^B_{AID_i}$ corresponding to PK_{AID_i}. The bank then returns $ID_B \parallel E_{PK_{AID_i}}(AID_i \parallel AID_i\_ExpTime \parallel AID_i\_Limit \parallel CERT^B_{AID_i} \parallel K_{AID_i,B}) \parallel E_{K_{AID_i,B}}(AID_i \parallel CERT^B_{AID_i} \parallel K_{AID_i,B})$ to user. Here $AID_i\_ExpTime$ is the expiry time of AID_i and AID_i_Limit is the credit limit of AID_i.
7. The user sends the ciphertext to the SE. SE retrieves the shared key $K_{AID_i,B}$ and the certificate $CERT^B_{AID_i}$.

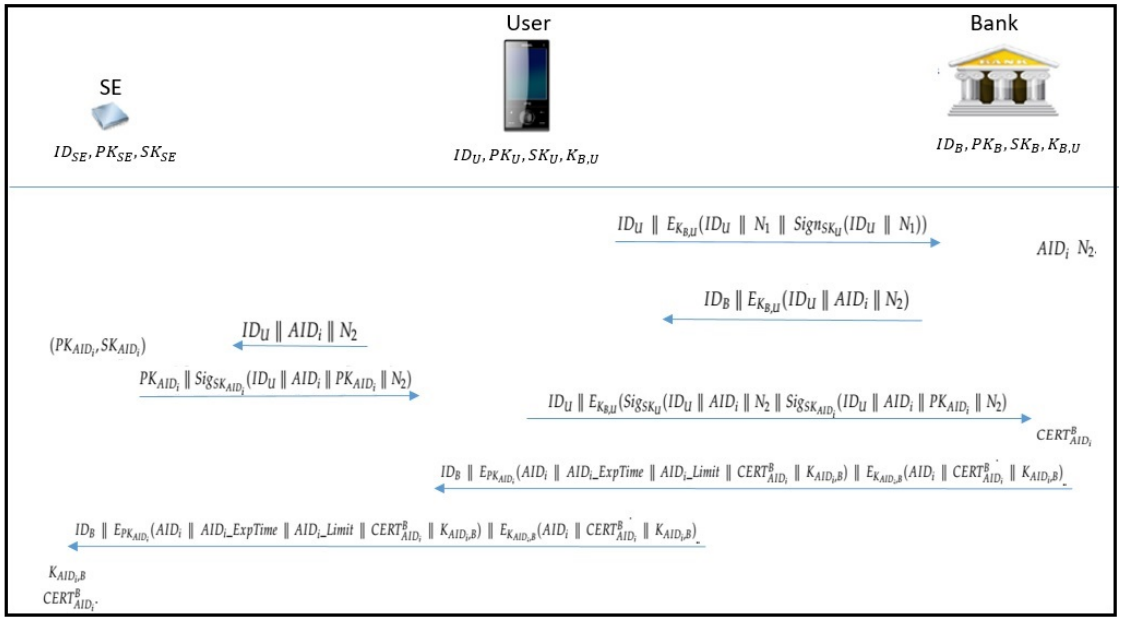

Figure 1. Anonymous Virtual Bank Account Generation Phase.

2.2.3. Anonymous Transaction Account Generation Stage

After making registration to the bank, the user then needs to register its (virtual) identity to TSM to get the virtual credit card. The virtual credit card will be used in the actual transactions. The user will establish a pre-stored credit account in TSM and the credit limit for this account can be decided by the user itself. This account will be linked to the virtual account AID_i in the bank for consuming via mobile payment. On the other hand, for security of the payment, user generates $BINFO$ (i.e., payment information) which is composed of virtual account AID_i, account expiry date $AID_i\_ExpTime$, account limit AID_i_Limit, and the session key $K_{AID_i,B}$. The message is signed by the signing key SK_{AID_i} , and then be sent to the bank via TSM. Besides, user will encrypt the payment message by session key $K_{TID_i,TSM}$, and then signed the cipher text by SK_{TID_i}. The signature as well as $TSMINFO$ are then sent to TSM. TSM decrypts it and then encrypts the content by using the bank's public key PK_B. TSM signs the cipher text to generate $TSMBINFO$. TSM transmits the $BINFO$ and $TSMBINFO$ to the bank. The bank can retrieve $BINFO$ and $TSMBINFO$. After comparing the information between the $BINFO$ and $TSMBINFO$, the bank authenticates the identities and returns the credit information of the virtual account to TSM. Figure 2 shows the communication flaws and the details are described as follows:

1. The user generates virtual transaction account TID_i and a key pair (PK_{TID_i}, SK_{TID_i}). He/she then signs PK_{TID_i} with SK_{TID_i} and encrypts it with PK_{TSM}. Then the user sends the ciphertext $E_{PK_{TSM}}(Sign_{SK_{TID_i}}(TID_i \| PK_{TID_i} \| Timestamp))$to TSM.
2. After decrypts the message, TSM establishes a session key $K_{TID_i,TSM}$ and returns $TID_i \| E_{PK_{TID_i}}(TID_i \| K_{TID_i,TSM})$ to the user.
3. The user requests identifiers SID, AID_i, ID_B and nonce N_1 to the SE.
4. The SE generates the payment message $BINFO$ and send it with N_1 to user. Here $BINFO = Sign_{SK_{AID_i}}(E_{K_{AID_i,B}}(SID \| AID_i \| ID_{TSM} \| ID_B \| AID_i\_ExpTime \| AID_i\_Limit \| N_2))$.
5. The user generates transaction message $TSMINFO = Sign_{SK_{TID_i}}(E_{K_{TID_i,TSM}}(SID \| AID_i \| ID_{TSM} \| ID_B \| N_2 \| AID_i\_ExpTime \| AID_i\_Limit))$ and encrypts $BINFO$, $TSMINFO$ and N_1 with PK_{TID_i} . That is, user sends $Sign_{SK_{TID_i}}(TID_i \| E_{PK_{TID_i}}(TID_i \| TSMINFO \| BINFO \| N_2)$ to TSM.
6. After decrypted $TSMINFO$, TSM will generate the authentication message $TSMBINFO = Sign_{SK_{TSM}}(E_{PK_B}(SID \| AID_i \| N_2 \| AID_i\_ExpTime \| AID_i\_Limit \| K_{TSM,B}))$ and sends $E_{PK_B}(ID_B \| AID_i \| BINFO \| SID \| ID_{TSM} \| TSMBINFO)$ to the bank for confirmation.

7. The bank uses its corresponding keys to decrypt the ciphertext. The bank then compares BINFO with TSMBINFO. The bank accepts the message if they are identical. In this case, the bank will send the credit information of AID_i to TSM.
8. After receiving the returned message, TSM verifies that TID_i is authorized to access the service and TSM will send $TID_i \parallel E_{K_{TID_i,TSM}}(Status \parallel TID_i\_ExpTime \parallel TID_i\_Limit)$ to the user.

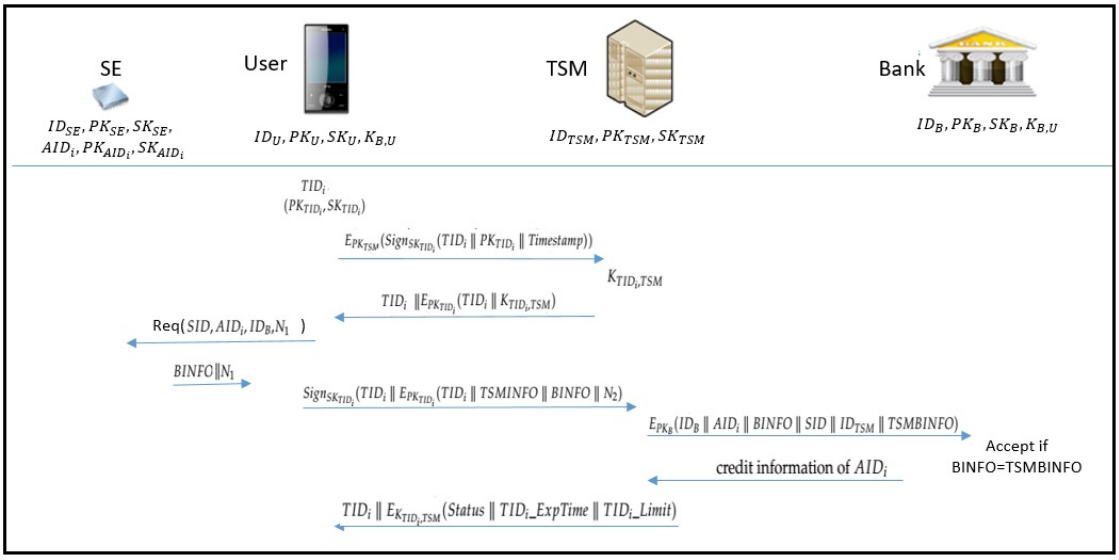

Figure 2. Anonymous Transaction Account Generation Phase.

2.2.4. Issuing of Virtual Credit Card Stage

During this phase, user can apply to TSM for a virtual credit card. TSM will issue a virtual credit card with shorter expiry date and lower credit limit. Besides, the credit card is complied with be EMV standard and is stored in the SE. A user can repeat this stage to get new virtual credit card when the expiry date is coming or remained limit is exhausted. Figure 3 shows the communication flaws and detail steps are listed as below:

1. The user sends a request to the SE with anonymous transaction identifier TID_i.
2. The SE generates a new public/private key pair (PK_{TID_i}, SK_{TID_i}) corresponding to TID_i and sends $Sign_{SK_{TDI_i}}(AID_i \parallel TID_i \parallel N_1 \parallel N_2) \parallel N_1$ to user.
3. The user sends the encrypted message $E_{K_{TID_i,TSM}}(Sign_{SK_{TDI_i}}(AID_i \parallel TID_i \parallel N_1 \parallel N_2) \parallel N_1))$ by key $K_{TID_i,TSM}$ to TSM.
4. After receiving the request, TSM will issue a new virtual credit card $TID_i - CreditINFO$ and generate a new certification $CERT^{TSM}_{TID_i}$, and sends the encrypted message $E_{K_{TID_i,TSM}}(TID_i - CreditINFO \parallel CERT^{TSM}_{TID_i})$ to the user.
5. After receiving the message, user decrypts ciphertext and stores the corresponding certification and the new credit card information into the SE.
6. The remaining process just follows the EMV standard.

2.3. Comments on Luo et al.'s Scheme

As mentioned at the beginning, this scheme focusses on the topic of user anonymity. However, it suffers from several problems. Lee et al. [31] pointed out that the scheme leaks the symmetric key shared between the SE and the bank. It means an adversary may attack the mobile device to find sensitive information in the SE. The adversary may also impersonate the mobile phone owner and do mobile payments on behalf of the real user.

Besides, in this protocol, it uses the same key pair for encryption/decryption and for (digital) signature signing/verification. This kind of mixing use of the same key is not recommended since

it may cause some unexpected security flaws. For example, if using the same key for both RSA encryption scheme and for RSA digital-signature scheme, then an attacker may eavesdrop some ciphertexts sending from others to the user, then the attacker may cheat the sender to encrypt the ciphertexts (for any reason the user may believe). As a result, , the attacker will get the plaintexts corresponding to the ciphertext. The reason this attack may happening is the same key pair used for signature and for encryption and this kind of mix-using should be avoided.

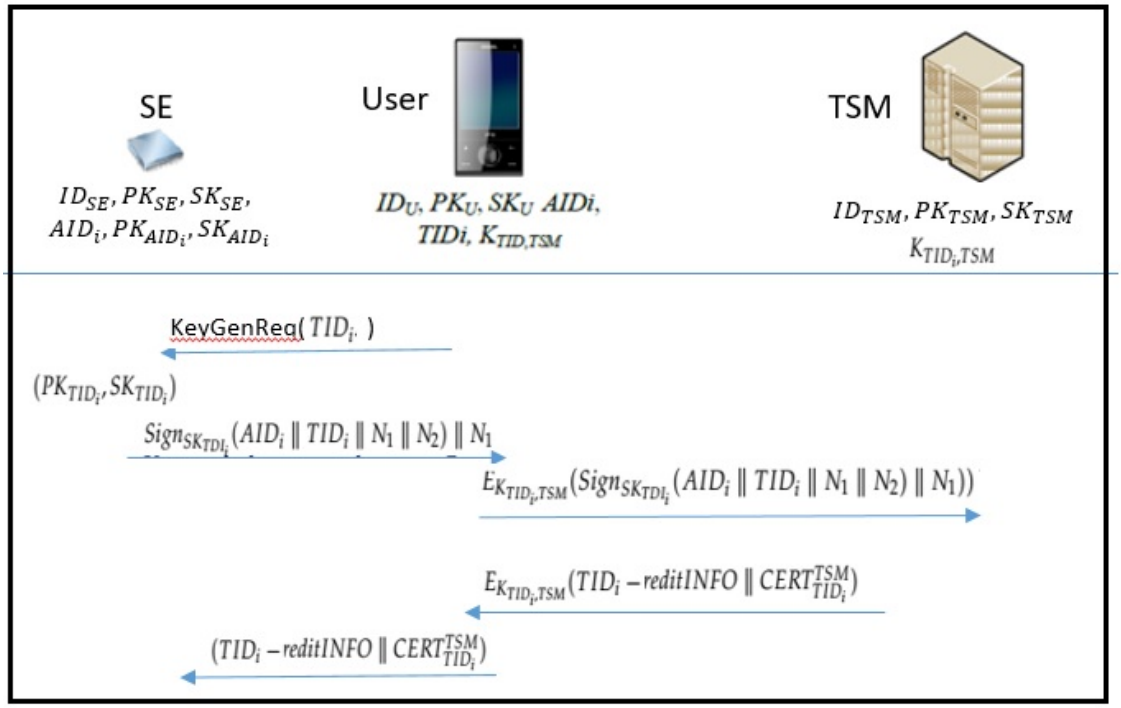

Figure 3. Issuing of Virtual Credit Card Phase.

3. Proposed Scheme

In this section, we introduce a new untraceable NFC-based anonymous mobile payment protocol to overcome the security weaknesses of Luo et al.'s scheme. There are three types of entities in our new scheme: a user, a bank and a TSM.

- A user is a customer who applies for a virtual account and a virtual transaction account for privacy protection reason. With the accounts he/she can pay using his NFC-enabled smartphone via our mobile payment protocol in an anonymous and untraceable manner.
- A bank is a card issuer who generates a virtual account and issues the corresponding virtual card for users.
- TSM is a very important entity in NFC payment ecosystem. TSM is assumed to be the trusted third party who sets up technical connections and business agreements with mobile network operators, or other entities controlling the SE on smartphones.

In addition, our scheme consists of four stages, (1) Initialization; (2) Virtual Account Application; (3) Virtual Transaction Account Application and Virtual Credit Card Issuance; (4) Virtual Credit Card and/or Virtual Transaction Account Updating. The details are described as follows and Table 1 lists the notations we will use in our protocol. Figure 4 shows the communication flaws of our new scheme.

Table 1. Notations.

Notations	Description
ID_a	The identifier of entity a
AID_i	Anonymous virtual account identifier for user i
TID_i	Anonymous virtual transaction account identifier of i
PW	The shared password between user and the bank
PK_a, SK_a	Public and private key pair of entity a
$PSK_{B,TSM}$	A secure and pre-shared key between Bank and TSM
$VA-request$	Virtual account registration request
$VTA-request$	Virtual transaction account registration request
$Update-request$	Virtual credit card update request
X_ExTime	Expiry time of x's certificate
X_Limit	Credit limit of account X
$Credit\_Info$	Corresponding information of a virtual credit card
TSM_Info	Payment information for TSM
N_j	j-th random number equals to $N_{j-1}+1$
$Ticket_{B,TSM}$	A ticket for accessing TSM generated by the bank
$H()$	A cryptographic one-way hash function
$E_k(m)$	Encryption of message m with key k
Sig_{SK_a}	Signature of entity a on the message m
$x \parallel y$	Concatenation of messages x and y
TS	Time stamp

3.1. Initialization

This is the stage for initial setting. At first, assume every single entity (i.e., user U, TSM, Bank) has their own identifiers (i.e., ID_U, ID_{TSM}, ID_B) at the beginning.

- The user U is assumed to have a physical bank account and a password pw shared with the bank (for authentication). The pw is assumed to be low-entropy (i.e., not as secure as a high-entropy secret key) so it can be kept secretly very easily (e.g., store in the SE of a smart phone or just keep it in mind without memorizing it anywhere).

On the other hand, both TSM and the bank are the organizations possessing with high levels power of computation, it is reasonable to assume that they can have the public-key cryptosystem's key pairs (PK, SK). Furthermore, we assume that their keys are Discrete-Logarithm-based (DL-based) keys (ref. Definition 1). The public-key cryptosystems are mainly used for authentication and for Diffie-Hellman-based key-exchange (ref. Definition 2). There are many candidates of such (DL-based) public-key cryptosystems such as Digital-Signature Standard (DSS), Schnorr Signature and/or ElGamail signature. On the other hand, confidentiality is achieved via symmetric-key cryptography such as AES or RC4 et al.

- TSM has its own public/private key pair (PK_{TSM}, SK_{TSM}). More precisely, $PK_{TSM} = (y_{TSM}, g_{TSM}, p_{TSM}, q_{TSM})$ where p_{TSM} is a large prime, g_{TSM} is a generator of a multiplicative group $G =< g_{TSM} >$ of order q_{TSM} and $y_{TSM} = g_{TSM}^{x_{TSM}} \bmod p_{TSM}$. $SK_{TSM} = x_{TSM} \in \mathbb{Z}_{q^*_{TSM}}$. In addition, TSM holds a high-entropy secret key $PSK_{B,TSM}$ shared with the bank.
- The same as TSM, the bank has its own public/private key pair (PK_B, SK_B). More precisely, $PK_B = (y_B, g_B, p_B, q_B)$ where p_B is a large prime, g_B is a generator of a multiplicative group $G' =< g_B >$ of order q_B and $y_B = g_B^{x_B} \bmod p_B$. $SK_B = x_B \in \mathbb{Z}_{q^*_B}$. In addition, the bank holds a high-entropy secret key $PSK_{B,TSM}$ shared with TSM.

In short, at the end of the stage, a user with identity ID_U has pw; TSM with identity ID_{TSM} has a symmetric key $PSK_{B,TSM}$, a DL-based public/private key pair (PK_{TSM}, SK_{TSM}) ; the bank with identity ID_B has a symmetric key $PSK_{B,TSM}$, a DL-based public/private key pair (PK_B, SK_B) and a pw as the one shared with ID_U. All those keys are generated in advance before the starting of our protocol.

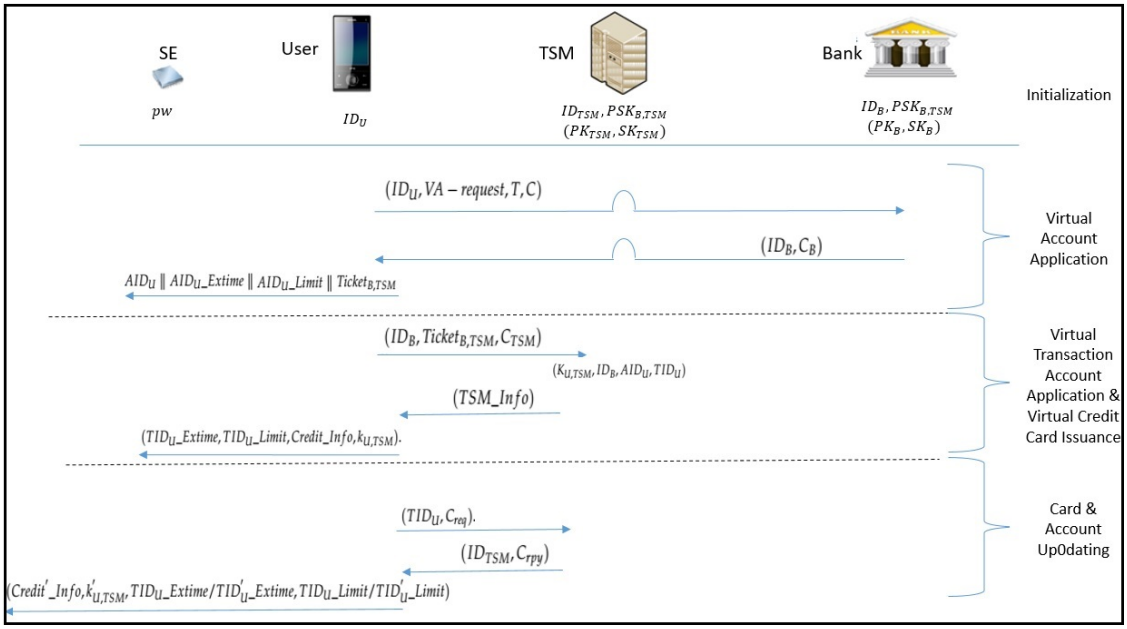

Figure 4. Communication Flaw of the Proposed Scheme.

3.2. Virtual Account Application

This stage is to authenticate the user via pw. It then aims to create a virtual account identifier AID_U and make it registered to the bank. The bank will record that together with user identity ID_U. Some information for later communication between U and TSM such as a ticket $Ticket_{B,TSM}$ will be sent back to the user side. Description in detail is listed as follows:

1. $U \rightarrow Bank: (ID_U, VA-request, T, C)$
   This step is to apply for registration and to inform the bank about which TSM the user will communicate with in the next stage. The user computes and does the following steps:
   - Pick $x \leftarrow Z^*_{q_B}$, use pw and bank's public key $PK_B = (y_B, g_B, p_B, q_B)$ to compute $T = g_B^x y_B^{pw} \bmod p_B$.
   - Pick $r \leftarrow Z^*_{q_{TSM}}$, use TSM's public key $PK_{TSM} = (y_{TSM}, g_{TSM}, p_{TSM}, q_{TSM})$ and compute $R = g_{TSM}^r \bmod p_{TSM}$.
   - Create a virtual account identifier AID_U, compute $k' = y_B^x \bmod p_B$, $k = H(ID_U \| ID_B \| k' \| T)$ and $h_{va} = H(VA-request \| ID_U \| ID_{TSM} \| AID_U \| T \| R \| N_1 \| TS_1)$ where N_1 is a nonce and TS_1 is a time stamp.
   - Compute $C = E_k(AID_U \| ID_{TSM} \| R \| N_1 \| TS_1 \| h_{va})$.
   - Send $(ID_U, VA-request, T, C)$ to the bank as request for virtual account registration.
2. $Bank \rightarrow U: (ID_B, C_B)$
   In this stage, the bank authenticates the user via pw, generates a virtual credit card and a ticket for AID_U. The ticket is generated for later communication between the user and the TSM. Detail steps of banks are described as follows:
   - Check the identity ID_U from its member-list and find the corresponding password pw. Reject and terminate if ID_U is not in the list.
   - Use pw to compute $k' = (T \times y_B^{-pw})^{sk_B} \bmod p_B$ and $k = H(ID_U \| ID_B \| k' \| T)$.
   - Decrypt C by key k and recover $(AID_U \| ID_{TSM} \| R \| N_1 \| TS_1 \| h_{va}) \leftarrow D_k(C)$.
   - Compute $h'_{va} = H(VA-request \| ID_U \| ID_{TSM} \| AID_U \| T \| R \| N_1 \| TS_1)$ and check the time stamp TS_1 . Accept the $VA-request$ if $h'_{va} = h_{va}$ and TS_1 is valid.
   - Records (ID_U, AID_U) in its database, determine the expiry time (i.e., AID_U_ExTime) and the credit limit (i.e., AID_U_Limit) of the credit card going to be issued to AID_U.
   - Use the symmetric key $K_{B,TSM}$ corresponding to ID_{TSM} and generate $Ticket_{B,TSM} = E_{K_{B,TSM}}(ID_B \| AID_U \| AID_U\_Extime \| AID_U\_Limit \| R \| TS_2 \| Lifetime \| Sig_{SK_B}(ID_B \|$

$AID_U \parallel AID_U\_Extime \parallel AID_U\_Limit \parallel R \parallel TS_2 \parallel Lifetime))$. Here $Sig_{SK_B}(M)$ is the signature of the bank on the message M.
- Generate a ciphertext by the session key k and get $C_B = E_K(ID_U \parallel AID_U \parallel AID_U\_Extime \parallel AID_U\_Limit \parallel N_1 + 1 \parallel Ticket_{B,TSM} \parallel TS_1^{'})$.
- Return ID_B and C_B to the user.

3. U: After receiving the returned information from the bank, the user U does the following computations:

- Decrypt C_B, check the time stamp $TS_1^{'}$ and the correctness of $N_1 + 1$.
- Store AID_U, AID_U_Extime, AID_U_Limit and $Ticket_{B,TSM}$ securely.

3.3. Virtual Transaction Account Application and Virtual Credit Card Issuance

This stage is to create a virtual transaction account TID_U of U and to make registration of TID_U to the TSM. Based on the account TID_U , TSM will issue a virtual credit card for U without knowing the real identity of U (i.e., TSM only knows AID_U alternatively). For the virtual credit card, TSM will generate the corresponding expiry time and limit of the card for TID_U, and then TSM will send the virtual credit card, the expiry time and account balance back to the user. Steps in detail are listed as follows:

1. $ID_U \rightarrow ID_{TSM} : (ID_B, Ticket_{B,TSM}, C_{TSM})$
The user does the following steps:

- Compute the session key $k_{U,TSM} = H(ID_U \parallel ID_{TSM} \parallel k_{TSM}^{'} \parallel R)$ where $k_{TSM}^{'} = y_{TSM}^{r} \bmod p_{TSM}$ and $r \in Z_{q_{TSM}}^{*}$ is the random number picked at step (1) of the previous stage.
- Generate a virtual transaction account TID_U and compute $h_{vta} = H(VTA - request \parallel AID_U \parallel TID_U \parallel ID_B \parallel Ticket_{B,TSM})$.
- Use $k_{U,TSM}$ and generate the ciphertext $C_{TSM} = E_{k_{U,TSM}}(VTA - request \parallel AID_U \parallel TID_U \parallel h_{vta})$.
- Sends $(ID_B, Ticket_{B,TSM}, C_{TSM})$ to TSM.
Note: The VTA-request is the request of registering U's virtual transaction account TID_U to TSM.

2. $ID_{TSM} : (K_{U,TSM}, ID_B, AID_U, TID_U)$
TSM does the following steps after receiving the information from U.

- Decrypt the $Ticket_{B,TSM}$ using the symmetric key $K_{B,TSM}$ shared with the bank in advance. Accept the ticket $Ticket_{B,TSM}$ if the signature from the bank is correct and the ticket is valid by checking the time stamp TS_2 and the lifetime. Do the following steps if $Ticket_{B,TSM}$ is accepted.
- Compute $k_{U,TSM} = H(ID_U \parallel ID_{TSM} \parallel k_{TSM}^{'} \parallel R)$ where $k_{U.TSM}^{'} = R^{SK_{TSM}} \bmod p_{TSM}$ and then decrypt C_{TSM} by the key $k_{U,TSM}$.
- Accept the $VTA - request$ if AID_U in C_{TSM} is the same as that in $Ticket_{B,TSM}$ and h_{vta} is correct.

3. $ID_{TSM} \rightarrow ID_U : (TSM\_Info)$
TSM does the following steps:

- Generate a virtual credit card and the corresponding information $Credit\_Info$ for TID_U.
- Determine the corresponding expiry time, TID_U_Extime, and credit balance, TID_U_Limit, where $TID_U\_Extime \leq AID_U\_Extime$ and $TID_U\_Limit \leq AID_U\_Limit$.
- Generate the ciphertext $TSM\_Info = E_{K_{U,TSM}}(ID_{TSM} \parallel AID_U \parallel TID_U \parallel TID_u\_Extime \parallel TID_U\_Limit \parallel Credit\_Info \parallel TS_3 \parallel Sig_{SK_{TSM}}(M))$ where M includes the whole message in the ciphertext TSM_Info excluding the signature.
- Return the ciphertext TSM_Info to the user U.

4. $ID_U \rightarrow SE : (TID_U\_Extime, TID_U\_Limit, Credit\_Info, k_{U,TSM})$.
The user does the following steps:

- Decrypt TSM_Info and verify the signature and the time stamp TS_3. Accept TSM_Info if it passed the verification.
- Store the necessary security information including the expiry date of TID_U, (i.e., TID_U_Extime), the credit balance of TID_U (i.e., TID_U_Limit), the information of the virtual credit card (i.e., $Credit\_Info$) and session key $k_{U,TSM}$ into the SE.

3.4. Virtual Credit Card and/or Virtual Transaction Account Updating

The goal of this stage is to allow a user to ask for a new virtual credit card from TSM. It will be launched when the date is closing to the expiry time or the remained limit is going to running out. Furthermore, user can change the virtual transaction account TID_U at this stage if necessary. Description in detail is listed as follows:

1. $ID_U \rightarrow ID_{TSM}: (TID_U, C_{req})$.
   The user ID_U does the following steps:

   - Pick a new random number $r' \in Z^*_{q_{TSM}}$ and compute $R' = g^{r'}_{TSM} \bmod p_{TSM}$.
   - (Optional) Generate a new virtual transaction account TID'_U in case of applying for update the virtual transaction account.
   - Compute $h_{req} = H(update-request \parallel TID_U \parallel (\text{optional})TID'_U \parallel R' \parallel TS_{req})$ and the ciphertext $C_{req} = E_{k_{U,TSM}}(update-request \parallel TID_U \parallel (\text{optional})TID'_U \parallel R' \parallel TS_{req} \parallel h_{req})$ where $k_{U,TSM}$ is the session key generated at the previous stage and stored in the SE.
   - Send (TID_U, C_{req}) to TSM

   Note: TID'_U is optional in this step.
2. $ID_{TSM} \rightarrow ID_U: (ID_{TSM}, C_{rpy})$
   TSM does the following steps after receiving the request.

   - Decrypt the ciphertext C_{req} by the session key shared with TID_U.
   - Accept the request if h_{req} is valid. Continue the following step if accepted.
   - Create a new virtual credit card and the corresponding information $(Credit\_Info', TID_U\_Extime, TID_U\_Limit)$. Alternatively, if TID'_U presented (i.e., user has also requested to change a new virtual transaction account TID'_U), change $(TID_U\_Extime, TID_U\_Limit)$ by $(TID'_U\_Extime, TID'_U\_Limit)$.
   - Compute the new session key $k'_{U,TSM} = R'^{SK_{TSM}} \bmod p_{TSM}$.
   - Sign and encrypt M_{rpy} where $M_{rpy} = ID_{TSM} \parallel TID_U \parallel TID'_U \parallel TID'_U\_Extime, TID'_U\_Limit \parallel Credit\_Info')$ if the virtual transaction account changed to TID'_U. Otherwise, $M_{rpy} = ID_{TSM} \parallel TID_U \parallel TID_U\_Extime \parallel TID_U\_Limit \parallel Credit\_Info'$. The resulted ciphertext is $C_{rpy} = E_{k'_{U,TSM}}(M_{rpy} \parallel Sig_{SK_{TSM}}(M_{rpy}))$.
   - Return (ID_{TSM}, C_{rpy}) to the user.
3. $ID_U \rightarrow SE: (Credit'\_Info, k'_{U,TSM}, TID_U\_Extime/TID'_U\_Extime, TID_U\_Limit/TID'_U\_Limit)$
   The user does the following steps:

   - Compute the new session key $k'_{U,TSM} = y^{r'}_{TSM} \bmod p_{TSM}$ and use it to decrypt C_{rpy}.
   - Accept C_{rpy} if the signature is valid.
   - Stores the necessary security parameters including new virtual credit card info, $Credit\_Info'$, new session key, $k'_{U,TSM}$, new expiry time, $TID_U\_Extime/TID'_U\_Extime$, and new credit balance, TID_U_Limit/TID'_U_Limit, to the SE.

Now, a new virtual credit card with lower credits, shorter expiry time and EMV-compatible has received by the user ID_U. It is stored in the SE and can be updated in accordance with the virtual credit card and/or virtual transaction account updating stage. The other rest transaction processes follow EMV standards, use an NFC smartphone with card emulation mode, and then transactions can be done in an efficient and secure way.

4. Security Analysis

We will analysis the confidentiality, anonymity and untraceability, integrity and unforgeability in this section. The security analysis is based on the assumption the bank and TSM are semi-trusted third parties (or called the honest-but-curious adversaries). A semi-trusted third party means a party that will act following the rules of the protocol but may also try to find sensitive data that should not be leaked to him/her if there is any security flaw in the protocol.

4.1. Confidentiality

Confidentiality is considered in three parts; the confidentiality between the user U and the bank during the virtual account application phase, the confidentiality between TSM and the user, and between TSM and the bank during the virtual transaction account application and virtual credit card issuance phase, and the confidentiality between the use and TSM during the account update phase.

Firstly, at the first stage, all the sensitive information sent between the user and the bank are encrypted by the Diffie-Hellman key $y_B^x \bmod p_B = k' = (T \times y_B^{-pw})^{sk_B} \bmod p_B$. Any passive attacker via eavesdropping is infeasible to compute the same key k' according to the CDH hardness problem (Definition 2). On the other hand, for any active attacker who attempts to impersonate the user with identifier ID_U or the bank, he/she must have the knowledge of the user's password pw and/or the bank's private key sk_B. The security of this part is protected by the technique of password-based authenticated key exchange protocol (PAKE). Our PAKE protocol is constructed following the concept of Abdalla and Pointcheval's simple password-based encrypted key exchange protocol [36] and can be proved secure following their security proofs. Consequently, no sensitive information is leaked during the virtual account application stage.

During the second phase (i.e., virtual transaction account application and virtual credit card issuance stage), TSM communicates with the user ID_U and the bank separately. All information sent between TSM and the bank are encrypted using the pre-shared key $K_{U,TSM}$. The information between user ID_U and TSM are encrypted using the Diffie-Hellman key $k'_{U.TSM} = R^{SK_{TSM}} \bmod p_{TSM} = y_{TSM}^r \bmod p_{TSM}$. No one can compute the same key without SK_{TSM} or r and r is chosen by the user ID_U. To avoid man-in-the-middle attack, $R = g_{TSM}^r \bmod p_{TSM}$ is firstly authenticated by the bank (i.e., the semi-honest third party) in the first stage. Then, R is encapsulated into the ticket $Ticket_{B,TSM}$ so that only TSM can decrypt it and recover R. Consequently, the session keys in this stage are all secure and authenticated. The same security analysis is also applied to the key in the virtual credit card and/or virtual transaction account updating stage. Finally, all the sensitive information of the user is stored in the SE of the user's smartphone. Therefore, we conclude that confidentiality is achieved in all stages of our protocol.

4.2. Anonymity and Untraceability

During the virtual account application stage, mutual authentications is achieved via password-based authentication. Only at this stage the user will reveal his real identity so only the bank knows a user's real identity. After the stage, the bank issues a virtual bank account AID_i to the user. Furthermore, at the virtual transaction account application stage, the user uses the virtual bank account AID_i to communicate with the TSM. Therefore, TSM does not know a user's real identity and TSM issues a temporary EMV-compatible virtual credit card to the virtual transaction account TID_i to the user. On the other hand, the communication between the user and TSM is encrypted using the Diffie-Hellman-based session key $k_{U,TSM}$ so even the bank has no ability to find the relationship between the virtual bank account AID_i and the virtual transaction account TID_i. Finally, a merchant will only know the TID_i and will not know the real identity of the user. Consequently, the anonymity of a user is achieved at all stages.

In short, the bank will only know the real identity ID_i and the virtual bank account AID_i. On the other hand, TSM will only know AID_i and virtual transaction account TID_i. Finally, a merchant only

knows TID_i of the corresponding credit card. Therefore, we conclude that no one can trace a transaction and link the real identity with the buyer of a transaction. The untraceability is consequently achieved.

4.3. Integrity and Unforgeability

Firstly, we assume a user of our scheme has no high-entropy secret key and no public/private key pair in advance. All the communication to or from the user is encrypted based on the Diffie-Hellman-based session key, so unforgeability is achieved by the secrecy of the generated session key. We also assumed TSM and the bank are semi-honest, so they will be considered as passive attackers who will not forge new messages. In addition, the hash values (i.e., h_{va}, h_{req}) provide information for the bank or TSM to verify the integrity of the received message. On the other hand, in our scheme, all transactions sent from the bank or from TSM are all signed by the sender and are recorded by the corresponding entities. Therefore, integrity and unforgeability can be achieved by the generated signatures from the viewpoints of the bank and TSM.

5. Performance and Comparison of Security Features

In this section, we show the performance of our proposed scheme from the viewpoint of communication costs. We consider only the cost that is required to be transmitted online during a transaction. That is, the transmission between user and bank, and the transmission between user and TSM. On the other hand, we will compare the features of our scheme with some other schemes.

To compute the online communication cost, we use the following assumptions.

- A request message (i.e., $\{VA, VTA, Update\} - request$) costs 20 bytes each.
- A DL-based signature is 40 bytes (using DSA [37], for example).
- A hash value is 32 bytes (using SHA-256, for example).
- A personal ID is about 20 bytes (using Unicode standard , a number or alphabet is 2 bytes. We assume an ID has 10 numbers or alphabets on average).
- A $Ticket_{B,TSM}$ has 112 bytes (i.e., $20 * 4 + 4 * 4 + 16 + 40 = 112$).
- A $Credit\_Info$ has 83 bytes according to [4].
- The size of a ciphertext is the same as its corresponding plaintext.
- All other information not defined here such as X_Limit, X_ExTime, TS cost 20 bytes each.

Table 2 shows the communication cost of our scheme.

Table 2. Communication Cost of the Proposed Scheme.

Stage	Length in Byte
Virtual Account Application	
$U \rightarrow Bank: (ID_U, VA - request, T, C)$	$20 + 20 + 128 + 256 = 424$
$Bank \rightarrow U: (ID_B, C_B)$	$20 + 132 = 152$
Virtual Transaction Account Application	
$ID_U \rightarrow ID_{TSM}: (ID_B, Ticket_{B,TSM}, C_{TSM})$	$20 + 112 + 316 = 448$
$ID_{TSM} \rightarrow ID_U: (TSM\_Info)$	$20 * 5 + 83 + 20 + 40 = 243$
Virtual Credit Card and/or Updating	
$ID_U \rightarrow ID_{TSM}: (TID_U, C_{req})$	$20 + 464 = 484$
$ID_{TSM} \rightarrow ID_U: (ID_{TSM}, C_{rpy})$	$20 + 223 = 243$

Next, we show the comparison of security features of our scheme with other schemes in Table 3.

Table 3. Comparison of Features.

	pw-Based	Anonymity	Untraceability	EMV-Compatible
[2]	X	X	X	X
[5]	X	X	X	X
[6]	X	X	X	X
[30]	X	O	X	O
[13]	X	O	O	X
[20]	X	X	X	X
[29]	X	X	X	O
Proposed Scheme	O	O	O	O

6. Conclusions

In this paper, we investigate the scheme introduced by Luo et al. at Computers and Electrical Engineering in 2016. We found some security flaws and weakness of the scheme. In addition, we introduce a new EMV-compatible NFC-based anonymous payment scheme. The important feature of the new scheme is the user needs only a low-entropy password shared with a bank in advance instead of a high-entropy secret key or a cumbersome public/private key pair. The new scheme provides many privacy preserving properties such as anonymity, untraceability and is suitable for mobile payments of users.

Funding: This research was supported by the Ministry of Science and Technology, Taiwan (ROC), under Project Numbers MOST 106-3114-E-004-002, MOST 105-2221-E-004-001-MY3 and MOST 106-3114-E-011-003.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Carr, M. Mobile Payment Systems and Services: An Introduction. 2007. Available online: http://www.mpf.org.in/docs/02/Mobile%20Payment%20Systems%20an%20Services.pdf (accessed on 8 August 2018).
2. Chen, Y.; Chou, J.; Sun, H.; Cho, M. A novel electronic cash system with trustee-based anonymity revocation from pairing. *Electron. Commer. Res. Appl.* **2011**, *10*, 673–682.
3. Fan, C.; Huang, V. Provably secure integrated on/off-line electronic cash for flexible and efficient payment. *IEEE Trans. Syst. Man. Cybern. Part C Appl. Rev.* **2010**, *40*, 567–579.
4. Ruiter, J.D.; Poll, E. Formal analysis of the EMV protocol suite. In Proceedings of the Theory of Security and Applications (TOSCA 2011), Saarbrücken, Germany, 31 March–1 April 2011; pp. 113–129.
5. Chen, W.; Hancke, G.; Mayes, K; Lien, Y.; Chiu, J. NFC mobile transactions and authentication based on GSM network. In Proceedings of the 2010 Second International Workshop on Near Field Communication (NFC), Monaco, Monaco, 20–20 April 2010; pp. 83–89.
6. Chen, W.; Hancke, G. Mayes, K; Lien, Y.; Chiu, J. Using 3G network components to enable NFC mobile transactions and authentication. In Proceedings of the IEEE International Conference on Progress in Informatics and Computing (PIC), Shanghai, China, 10–12 December 2010; pp. 441–448.
7. Hassinen, M.; Hypponen, K.; Trichina, F. Utilizing national public-key infrastructure in mobile payment systems. *Electron. Commer. Res. Appl.* **2008**, *7*, 214–231.
8. Kabir, Z. User Centric Design of an NFC Mobile Wallet Framework. Master's Thesis, The Royal Institute of Technology (KTH), Stockholm, Sweden, 2011.
9. Kungpisdan, S.; Srinivasan, B.; Le, P. A secure account-based mobile payment protocol. In Proceedings of the International Conference on Information Technology: Coding and Computing, Las Vegas, NV, USA, 5–7 April 2004; pp. 35–39.
10. Yang, J.H.; Lin, P.Y. A mobile payment mechanism with anonymity for cloud computing. *J. Syst. Softw.* **2016**, *116*, 69–74.
11. Toorani, M.; Beheshti, A. SSMS-a secure SMS messaging protocol for the m-payment systems. In Proceedings of the IEEE Symposium on Computers and Communications, ISCC, Marrakech, Morocco, 6–9 July 2008; pp. 700–705.

12. Molloy, I.; Li, J.; Li, N. Dynamic Virtual Credit Card Numbers. In Proceedings of the 11th International Conference on Financial Cryptography and 1st International Conference on Usable Security, Scarborough, Trinidad and Tobago, 12–16 February 2007; Springer: Berlin, Germany, 2007; pp. 208–223.
13. Martínez-Peláez, R.; Rico-Novella, F.; Satizábal, C. Mobile payment protocol for micropayments: Withdrawal and payment anonymous. In Proceedings of the New Technologies, Mobility and Security, NTMS'08, Tangier, Morocco, 5–7 November 2008; pp. 1–5.
14. Liao, H. Cross-domain anonymous online payment protocol. *J. Electron. Commer.* **2007**, *9*, 779–799. (In Chinese)
15. Haselsteiner, E.; Breitfuβ, K. Security in near field communication (NFC). In Proceedings of the RFIDSec'06 on RFID Security, Graz, Austria, 12–14 July 2006; pp. 12–14.
16. NFC. Available online: https://zh.wikipedia.org/wiki/%E8%BF%91%E5%A0%B4%E9%80%9A%E8%A8%8A (accessed on 1 May 2016).
17. NFC Comparison Table. Available online: http://blog.mtkfan.com/?p=86 (accessed on 1 August 2016).
18. Cheng, H.C.; Chen, J.W.; Chi, T.Y.; Chen, P.H. A generic model for NFC-based mobile commerce. In Proceedings of the 11th International Conference on Advanced Communication Technology, Gangwon-Do, Korea, 15–18 February 2009; pp. 2009–2014.
19. Noh, S.K.; Choi, D.Y.; Kim, H.G.; Seo, D.K.K.J.H.; Kim, J.W.; Cha, B.R. Proposed of micropayment and credit card model using NFC technology in mobile evironment. *Int. J. Multimed. Ubiquitous Eng.* **2013**, *8*, 295–305.
20. Noh, S.K.; Lee, S.K.; Choi, D. Proposed m-payment system using near-field communication and based on WSN-enabled location-based services for m-commerce. *Int. J. Distrib. Sens. Netw.* **2014**, *10*, 856172.
21. Steffens, E.-J.; Nennker, A.; Ren, Z.; Yin, M.; Schneider, L. The SIM-based mobile wallet. In Proceedings of the 13th International Conference on Intelligence in Next Generation Networks (ICIN), Bordeaux, France, 26–29 October 2009; pp. 1–6.
22. Apple Inc. Apple Pay. Available online: https://www.apple.com/apple-pav/ (accessed on 12 December 2017).
23. Microsoft Corp. Trusted Platform Module (TPM) Virtual Smart Card Management Protocol Specification. Available online: http://msdn.microsoft.com/en-us/library/hh880895(prot.20).aspx (accessed on 24 December 2017).
24. Google Corp. Google Wallet. Available online: http://www.google.com/wallet/ (accessed on 12 March 2017).
25. HCE. Available online: https://en.wikipedia.org/wiki/Host$_$card$_$emulation (accessed on 6 June 2017).
26. Mainetti, L.; Patrono, L.; Vergallo, R. IDA-Pay: An innovative micro-payment system based on NFC technology for android mobile devices. In Proceedings of the 20th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 11–13 September 2012; pp. 1–6.
27. Paillés, J.C.; Gaber, C.; Alimi, V.; Pasquet, M. Payment and privacy: a key for the development of NFC mobile. In Proceedings of the 2010 International Symposium on Collaborative Technologies and Systems (CTS), Chicago, IL, USA, 17–21 May 2010; pp. 378–385.
28. Pasquet, M.; Reynaud, J.; Rosenberger, C. Secure payment with NFC mobile phone in the smart touch project. In Proceedings of the International Symposium on Collaborative Technologies and Systems (CTS), Irvine, CA, USA, 19–23 May 2008; pp. 121–126.
29. Urien, P.; Piramuthu, S. Securing NFC mobile services with cloud of secure elements (CoSE). In Proceedings of the 5th International Conference on Mobile Computing, Applications and Services (MobiCASE), Paris, France, 7–8 November 2013; pp. 322–331.
30. Luo, J.N.; Yang, M.H.; Huang, S.Y. An unlinkable anonymous payment scheme based on near field communication. *Comput. Electr. Eng.* **2016**, *49*, 198–206.
31. Lee, H.; Kim, J.; Jung, J.; Lee, Y.; Won, D. An enhanced unlinkable anonymous payment scheme based on near field communication. In Proceedings of the 11th International Conference on Ubiquitous Information Management and Communication, Beppu, Japan, 5–7 January 2017; p. 38.
32. EMV. Available online: https://zh.wikipedia.org/wiki/EMV (accessed on 6 June 2017).
33. EMVCo. Available online: https://www.emvco.com/ (accessed on 6 June 2017).
34. EMVCo Tokenization. Available online: https://www.emvco.com/specifications.aspx?id=263 (accessed on 6 June 2017).

35. Diffie, W.; Hellman, M.E. New directions in cryptography. *IEEE Trans. Inf. Theory* **1976**, *22*, 644–654. Diffie-Hellman Key Exchange. Available online: https://en.wikipedia.org/wiki/Diffie-Hellman$_$key$_$exchange (accessed on 8 August 2018).
36. Abdalla, M.; Pointcheval, D. Simple password-based encrypted key exchange protocols. In Proceedings of the CT-RSA'05 2005 international conference on Topics in Cryptology, San Francisco, CA, USA, 14–18 February 2005; Volume 3376, pp. 191–208.
37. *Digital Signature Standard (DSS)*; National Institute of Standards and Technology: Gaithersburg, MA, USA, 2013.