Computer Standards & Interfaces 54 (2017) 55–60

Contents lists available at ScienceDirect

Computer Standards & Interfaces

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

A secure and privacy-preserving mobile wallet with outsourced verification in cloud computing

Zhen Qin[a], Jianfei Sun[a], Abubaker Wahaballa[a], Wentao Zheng[a], Hu Xiong[b,*], Zhiguang Qin[a]

[a] University of Electronic Science and Technology of China, Chengdu 610051, China
[b] State Key Laboratory of Cryptology, P.O. Box 5159, Beijing 100878, China

ARTICLE INFO

Keywords:
Mobile wallet
Digital signature
Secure computation outsourcing
Cloud computing

ABSTRACT

Mobile wallet, also known as mobile payment, is becoming one of the most frequently used approach to provide payment services under financial regulation via mobile device and may redefine our lifestyle with the rapid popularity of mobile Internet. In this paper, we address the security of the mobile wallet by providing a detailed threat analysis and identifying some unique design requirements in terms of security and privacy protection for mobile wallet. We then provide a novel approach to secure the mobile wallet and protect the privacy of the mobile user by incorporating the digital signature and pseudo-identity techniques. In view of several advantages of cloud computing, the computation task on the client side, which is usually featured with limited computation resources, is outsourced to the untrusted cloud server securely. The performance of our approach is evaluated via both theoretic analysis and experimental simulations. Also, the security analysis demonstrate that our approach can achieve desirable security properties of mobile wallet.

1. Introduction

The growth of financial-services apps and the availability of mobile device drives the growth of mobile payment services. As one of the modern components of mobile payment services, mobile wallet (m-Wallet) [1,2] provides a very convenient way to allow the clients to conduct the payment via his/her mobile device from anywhere and anytime. According to a recent report from Transparency Market Research [3], the global mobile wallet market is expected to reach USD 1,602.4 billion in 2018. The most famous mobile wallets including Google Wallet, MasterPass and Apple Pay.

In view of the tremendous benefits provided by the mobile wallet and the huge number of potential users (hundreds of millions world-wide), it is obvious that mobile payment is likely to become the most popular payment method in the near future. The wide adoption of mobile devices, such as smart phone, iPAD and PDA, does not only introduce huge business opportunities, but also raises daunting security challenges due to the open-medium nature of wireless communications and the limited resources of the mobile devices. So far, limited attention has been paid to the security of mobile wallet. Unfortunately, the mobile wallet would not be preferred by the public without the guarantee of message authentication and privacy preserving. First of all, it is essential to ensure that payment information exchanged between the client and the merchant cannot be impersonated or modified by any attacker. Otherwise, the forged payment information may be fatal to the reputation of the mobile wallet. On the other hand, the real identity of the malicious customer should be disclosed by the system manager; but meanwhile the privacy of the honest customer should be protected as far as possible. Despite these concerns seem similar to those identified in other wireless networks, the nature of mobile payment such as the size of the network and the limited resources of the mobile devices make the problem very novel and challenging. The purpose of this paper is to bring a first glance to this challenge. As there is quite limited resources (e.g., computing resource and battery power) for mobile devices, the traditional secret methods can not be directly applied by the mobile devices in the scenarios of mobile payment. There is unlimited resources for the cloud computation, it has the ability of taking over the heavy computation workload instead of resources-constraint mobile devices. However, the cloud server can not be trusted. If the computation workload is directly outsourced to the third party cloud services provider, it would cause the privacy leakage of mobile wallet users. Thus, it is necessary to solve this problem with the secure outsourcing computing.

Digital signature [4] is a promising approach to offer the authentication and non-repudiation of the payment information during the mobile wallet. However, the digital signature in the traditional public key cryptography [5] and identity-based cryptography [6] suffers from the heavy cost of the certificate management and key escrow problem

* Corresponding author.
E-mail address: xionghu.uestc@gmail.com (H. Xiong).

http://dx.doi.org/10.1016/j.csi.2016.11.012
Received 14 March 2016; Received in revised form 3 November 2016; Accepted 25 November 2016
Available online 26 November 2016
0920-5489/

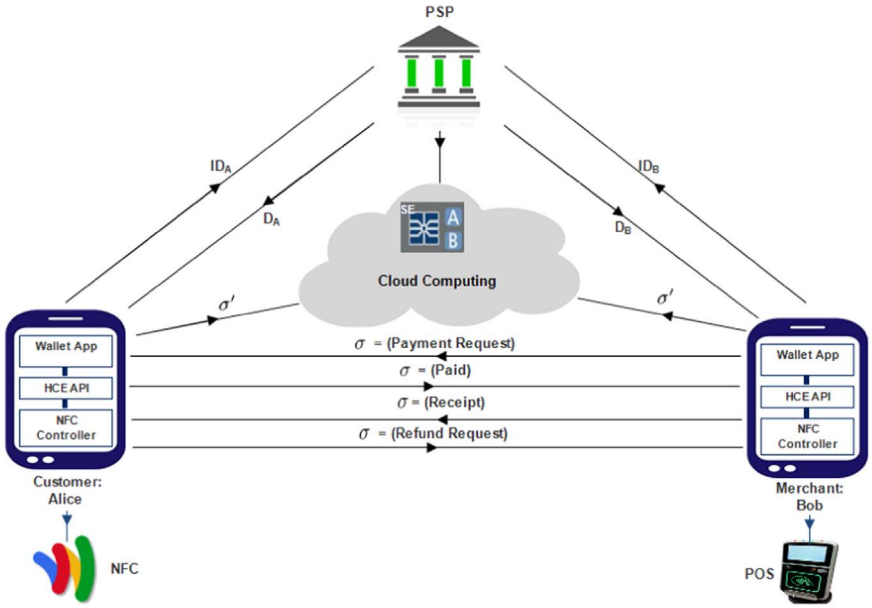

Fig. 1. System model.

respectively. Fortunately, certificateless signature [7–11] has been introduced to avoid both of the certificate management and key escrow problems simultaneously. Therefore, it is appropriate to design secure mobile wallet based on the idea of certificateless signature.

By carefully exploring the unique characteristics of mobile wallet and examining the existing cryptographic techniques, we present a lightweight privacy-preserving authentication protocol for securing mobile wallet based on certificateless signature scheme. Our main contributions can be summarized as follows:

- The security threats facing the mobile wallet have been identified according to its characteristics and the security requirements of mobile wallet are defined correspondingly to capture the potential attacks.
- Based on the certificateless signature and pseudo-identity technique, a novel secure mobile wallet protocol has been proposed to offer unforgeability, anonymity and traceability. Furthermore, the heavy verification overhead on the mobile customer side with limited resources has been outsourced to the untrusted cloud server.
- We have implemented our protocol to evaluate the performance. The experimental result shows that our protocol can be deployed in a resource-limited mobile device. Also, the security analysis is given to examine the correctness and soundness of the proposed protocol.

The remainder of this paper is organized as follows. In next section, the preliminaries and technical background that are needed in this paper are presented. A secure mobile wallet is introduced and discussed in Section 3. Section 4 deals with security analysis and efficiency comparison. Conclusion and future trends are pointed out in Section 5.

2. Preliminaries

2.1. System model

In this section, we give some intuition for the idea behind our protocol. A secure mobile wallet protocol follows the insights of prior works [1,2,12]. In order to offer unforgeability, non-repudiation and traceability of transmitted messages, we adopt certificateless signature scheme [13]. We incorporate the idea of outsourcing technique [14] with certificateless signature to reduce computation overhead in mobile devices. Our protocol consists of the following entities and components:

1. *Customer*: *Alice*. The consumer who wants to purchase goods or services provided by merchant.
2. *Merchant*: *Bob*. The merchant Bob who wants to sell goods or services to consumers.
3. *Payment Service Provider* ($\mathcal{PSP}$). A payment service provider PSP is responsible for the security and privacy of the payment information.
4. $\mathcal{CSVP}$. Untrusted Cloud Server Verification Provider. This entity reduces the computation overhead at user side by outsourcing the computation of verification to $\mathcal{CSVP}$.
5. *Wallet App*: An application that allows the user to perform payment transaction.
6. *Secure Element* (*SE*): A secure element is a platform used to store the credentials of users on secure cloud storage.
7. *Host card emulation API* (*HCE*): Host card emulation is an interface that allows the user (Alice) to perform card emulation on Near Field Communication Point-Of-Sale (NFC-POS) using her mobile phone.
8. *NFC technology*: Near-field communication (NFC) [15] refers to a set of communication protocols that enable two electronic devices, one of which is usually a portable/mobile device, to establish communication in case both of devices are within 4 cm of each other. The communication protocol between mobile device and merchant is assumed as NFC according to ISO 13157.

The interaction scenario between above entities and components is sketched in Fig. 1. This interaction is divided into three phases: *Setup and Key Generation Phase*, *Payment Transaction Phase* and *Outsourced Verification Phase*. In setup and key generation phase, $\mathcal{PSP}$ inputs a security parameter k and the description of a finite signature space and a description of a finite message space. Then, it generates the public parameters PP and a master key s. Further, it also takes the user's real identity $ID \in \{0, 1\}^*$ and its master key s as inputs, it outputs the user's secret key SK_{ID} and pseudo identity P_{ID}. In payment transaction phase, there are two types of payments: in-store payment and online payment. In-store payment, Alice tries to perform in-store payment transactions using her mobile wallet and Bob's NFC-POS. Upon receiving payment request on Bob's NFC-POS, she touches the NFC-POS by her mobile phone. Then, she uses her private key to sign a message with input (transaction id, her id and amount paid). In online payment, the payment transaction is performed remotely over

Table 1
Notations of our protocol.

Notation	Meaning	Notation	Meaning
$\mathcal{PSP}$	Payment Service Provider	$\mathcal{CSVP}$	Untrusted Cloud Server Verification Provider
POS	point of sale	NFC	Near Filed Communication
ID_A	Alice's real identity	ID_B	Bob's real identity
P_{ID_A}	Alice's pseudo identity	P_{ID_B}	Bob's pseudo identity
D_A	Alice's shot-time partial private key	D_B	Bob's shot-time partial private key
x_A	Alice's private key	x_B	Bob's private key
PP	Public parameters	P_{pub}	The $\mathcal{PSP}$'s public master key
H_1,H_2	Two hash functions	$\hat{e}$	A bilinear pairing
T_{ID}	Transaction identity	$Amount_{T\setminus R}$	Amounts transferred or received
σ	A signature on message m	$\oplus$	An Exclusive-OR (XOR)

the Internet, using 3G/4G or WiFi wireless connection. To achieve privacy-preserving, we employ the tamper-proof device [16,17] to drive enough pseudo identities at the user's request. At the end of payment transaction phase, Alice receives an acknowledgment of receipt, which is signed by Bob's private key. To reduce the computation overhead at user side, we utilize the server-aided verification protocol [18]. This allows the user to transfer the received signatures to untrusted cloud server verification provider $\mathcal{CSVP}$. Then, $\mathcal{CSVP}$ preforms the signatures verification, and sends the result back to the user.

For better readability, the notations of our proposed protocol are shown in Table 1.

2.2. Threat model

In recent years, a number of mobile payment protocols have been proposed. These protocols adopt various kinds of technologies such as NFC [19,20], SMS [21,22], RFID [23,24], bluetooth [25,26], WAP [27] and IrDA [28], to communicate with mobile phone users for payment transactions. However, the aforementioned schemes have some limitations particularly in terms of security, easy-of-use and efficiency Fig. 2.

The NFC's payment schemes use either SIM card or embedded hardware component to store the secure element (SE). However, the major drawback of SIM-based SE is that the user must use special-purpose SIM card, while the embedded SE component suffers from that the cell phone manufacturers control the access to the SE. In SMS-based payment schemes, the SMS messages are used to exchange the necessary credentials for users in the m-payment systems. However, these messages are stored in SMS Center (SMSC) in order to forward them to the target mobile device. Therefore, plaintext SMS messages become a target for attackers and dishonest SMSC staff [29]. RFID technology is similar to NFC, but with a longer transmission distance. However, RFID tags respond to the reader's request with their unique identifier, and without alerting their owners, even if these tags were protected with cryptographic algorithms [30,31]. Bluetooth Low Energy (BLE) has been supported on most mobile phone platforms, with its adoption in mobile payment system. However, BLE devices may be vulnerable to BlueSnarfing, Bluejacking, Bluebugging and DoS attacks [32]. WAP-based m-payment solutions are widely adopted to perform the payment transaction online by using the bank's WAP gateway. The disadvantages of these schemes are high cost of infrastructure and the extra charges of the third agency or party gateway [33].

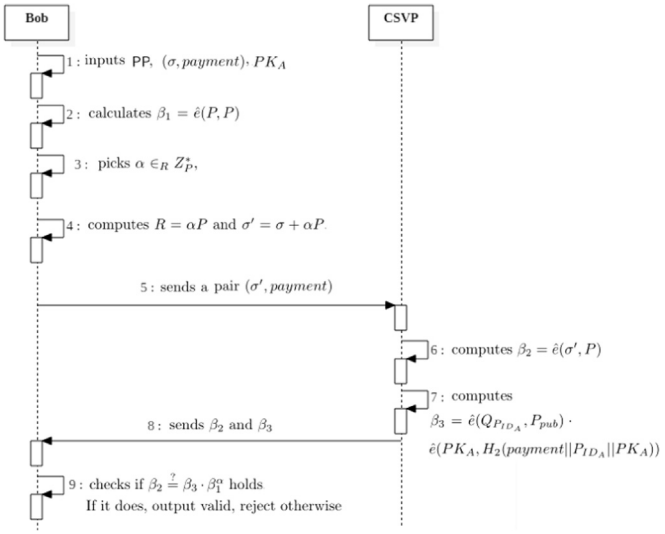

Fig. 2. Outsourced verification.

As in Google wallet [34], we adopt Near Field Communication (NFC) and Host Card Emulation (HCE) technologies in order to propose a secure and privacy-preserving mobile wallet.

2.3. Design goals

In order to maintain mobile wallet security, our protocol should be able to satisfy the following requirements:

- *Unforgeability*: Only legal users can make transactions. In other words, no one should be able to impersonate any user to submit a fake payment or a fake or illegal receipt.
- *Anonymity*: The identities of users must be kept confidential.
- *Traceability*: Bob cannot deny the received payment, while the Alice cannot deny her confirmed payment. Otherwise, the unique identifiers can be used to trace them.
- *Non-repudiation*: The merchant cannot repudiate the origin and the correctness of the receipt information. Also no customer can deny his/her confirmed payment.
- *Small Overhead*: Due to the limited resources of mobile devices, small overhead must be provided for both computational cost and communication overhead.

3. Our protocol

In this section, we concretely construct a secure mobile wallet protocol from certificateless signature with server-aided verification. For convenience, we let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of the same large prime q. Let $\hat{e}: \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be an admissible pairing which satisfies the following properties:

1. *Bilinear*: $\hat{e}(P+Q, R) = \hat{e}(P, R)e(Q, R)$ and $\hat{e}(P, Q+R) = \hat{e}(P, Q)e(P, R)$ for all $P, Q, R \in \mathbb{G}_1$;
2. *Non-degenerate*: There exists $P, Q \in \mathbb{G}_1$ such that $\hat{e}(P, Q) \neq 1$;
3. *Computability*: There is an efficient algorithm to compute $\hat{e}(P, Q)$ for all $P, Q \in \mathbb{G}_1$.

To prove the security of our scheme in random oracle model, we use of the following hardness assumption:

1. Discrete Logarithm (DL) Problem: Given (P,aP) then compute a, where $P \in \mathbb{G}_1$ is the generator and $a \in \mathbb{Z}_q^*$ is unknown.
2. Computational Diffie-Hellman (CDH) problem: Given (P, aP, bP) then compute abP, where $P \in \mathbb{G}_1$ is the generator and $a, b \in \mathbb{Z}_q^*$ is unknown.

As mentioned in Section 2.1, our protocol consists of the following phases:

3.1. Setup and key generation phase

- **Setup**: Initially, $\mathcal{PSP}$ inputs a security parameters. These include two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order p and a bilinear pairing $\hat{e}: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. The $\mathcal{PSP}$ randomly chooses its master-key $s \in Z_p^*$ and computes its public master-key $P_{pub} = sP$, where P is generator of $\mathbb{G}_1$. It also chooses two hash functions H_1: $\{0, 1\}^* \rightarrow \mathbb{G}_1$ and H_2: $\{0, 1\}^* \rightarrow Z_p^*$. Finally, the $\mathcal{PSP}$ publishes the system parameters: $\mathsf{PP} = (\mathbb{G}_1, \mathbb{G}_2, \hat{e}, p, P, H_1, H_2, P_{pub})$.
- **Key Generation**:
  - Step 1: Alice chooses her real $ID_A \in \{0, 1\}^*$. Afterward, Alice sends her real ID_A to the $\mathcal{PSP}$. The $\mathcal{PSP}$ stores Alice's ID_A and her credentials in secure cloud storage.
  - Step 2: Pseudo identity and short-time partial private key generation: We assume that Alice and other customers can obtain enough pseudo identities and partial private keys from $\mathcal{PSP}$ during payment transaction phase. The $\mathcal{PSP}$ uses the tamper-proof device [16,17] to generates Alice's pseudo identity P_{ID_A} and short-time partial private key D_A as follows.
    1. Pseudo identity generation: The $\mathcal{PSP}$ composes Alice's pseudo identity P_{ID_A} into P_{ID_a} and P'_{ID_a}. Afterward, $\mathcal{PSP}$ picks $x \in_R Z_p^*$ and computes $P_{ID_a} = xP$ and $P'_{ID_a} = ID_A \oplus H_1(xP_{pub})$. Finally, $\mathcal{PSP}$ sets Alice's pseudo identity as: $P_{ID_A} = (P_{ID_a}, P'_{ID_a})$.
    2. short-time partial private key generation: The $\mathcal{PSP}$ computes $Q_{P_{ID_A}} = H_1(P_{ID_A})$ and Alice's short-time private key $D_A = sQ_{P_{ID_A}}$. Then $\mathcal{PSP}$ sends the short-time partial private key D_A and pseudo identity P_{ID_A} to Alice secretly.
  - Step 3: Full private key and public key extraction: Alice picks as her private key $x_A \in_R Z_p^*$. Then, she computes her public key $PK_A = x_{AP}$.
  - Step 4: Bob with $ID_B \in \{0, 1\}^*$ repeats step 1, 2 and 3 to register at $\mathcal{PSP}$ and get his pseudo identity P_{ID_B}, short-time partial private D_B and full private key x_B.

3.2. Payment transaction phase

In this phase, there are two types of payments: in-store payment and online payment. In-store payment, Alice tries to perform in-store payment transactions using her mobile wallet and Bob's NFC-POS. In online payment, the payment transaction is performed remotely over the Internet, using 3G/4G or WiFi wireless connection. This phase looks into the following steps:

- Step 1: Bob initiates payment request on his NFC-POS or website. This includes transaction identity T_{ID}, amount to be paid and his pseudo identity P_{ID_B} as: $payment = (T_{ID} \parallel Amount_T \parallel P_{ID_B})$
- Step 2: Upon receiving payment request, Alice uses her private key to generate a signature as $\sigma = D_A + H_2(payment \parallel P_{ID_A} \parallel PK_A)x_A$.
- Step 3: Given a pair $(\sigma, payment)$ and Alice's public key PK_A, Bob can check the validity of a signature as follows:
  - Check if $\hat{e}(\sigma, payment) \stackrel{?}{=} \hat{e}(Q_{P_{ID_A}}, P_{pub})\hat{e}(PK_A, H_2(payment \parallel P_{ID_A} \parallel PK_A))$ holds. If it does, output valid, reject otherwise.
- Step 4: Once Bob gets the validity of Alice's wallet and the integrity of the payment information, he uses his private key to sign a receipt as:
  1. Set the receipt info as: $Receipt = (T_{ID} \parallel Amount_R)$
  2. Generate signature $\sigma = D_B + H_2(Receipt \parallel P_{ID_B} \parallel PK_A)x_B$.
- Step 5: Alice can check the validity of the signature pair $(\sigma, Receipt)$ as in Step 3 during this phase.

3.3. Outsourced verification phase

Due to the limited resources of mobile devices, server-aided verification technique is adopted in our protocol. This technique reduces the computation overhead at user side significantly, compared with original verification process in Step 3 during previous phase. In this phase, Bob and sever $\mathcal{CSVP}$ interact with each other in following steps. These steps are sketched in Fig. 3:

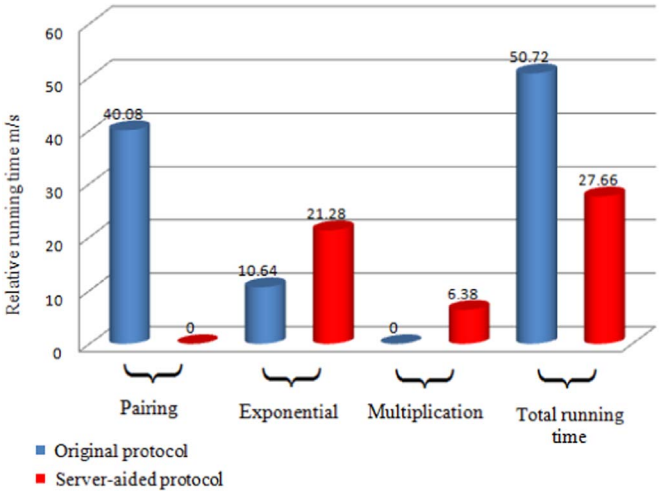

Fig. 3. Efficiency comparison.

1. Given the public parameters PP, Bob calculates $\beta_1 = \hat{e}(P, P)$. Then, he picks $\alpha \in_R Z_P^*$, computes $R = \alpha P$ and $\sigma' = \sigma + \alpha P$. Finally, he sends a pair $(\sigma', payment)$ to the $\mathcal{CSVP}$.
2. Upon $\mathcal{CSVP}$ receiving $(\sigma', payment)$ from Bob, It computes $\beta_2 = \hat{e}(\sigma', P)$ and $\beta_3 = \hat{e}(Q_{P_{ID_A}}, P_{pub})\cdot\hat{e}(PK_A, H_2(payment \parallel P_{ID_A} \parallel PK_A))$. Then, it sends β_2 and β_3 to Bob;
3. Bob checks if $\beta_2 \stackrel{?}{=} \beta_3\cdot\beta_1^{\alpha}$ holds. If it does, output valid, reject otherwise.

Correctness. If σ is a valid signature, it can be easily seen that $\beta_2 = \beta_3\cdot\beta_1^{\alpha}$ as:

$$\beta_2 = \hat{e}(\sigma', P) = \hat{e}(\sigma + \alpha P, P) = \hat{e}(Q_{P_{ID_A}}, P_{pub}) \cdot\hat{e}(PK_A, H_2(payment \parallel P_{ID_A} \parallel PK_A))\cdot\hat{e}(P, P)^{\alpha} = \beta_3\cdot\beta_1^{\alpha}$$

4. Security analysis and efficiency comparison

4.1. Efficiency evaluation

The scope of this subsection is to evaluate the performance of both original and server-aided proposed protocols. For simplicity, we define the following notations: T_{pa}, T_{ex} and T_{mp} for bilinear pairing, pairing exponentiation and scalar multiplication operations respectively. The performance evaluation is conducted based on the experimental results of [35]. The relative times of considered cryptographic operations are given in milliseconds as: $T_{pa} = 20.04$, $T_{ex} = 10.64$ and $T_{mp} = 6.38$, where the hardware platform is PIV 3 GHZ processor with memory 512 MB and the Windows XP operating system. As indicated in Table 2, the total running time of original proposed protocol is 50.72 m/s and 27.66 m/s for the Server-aided protocol. Fig. 3 shows the relative times of considered cryptographic operations for both original and server-aided protocols. It is worth remarking that, based on the aforementioned results. It is evident that the server-aided protocol is more practical and applicable as compared to the original protocol.

4.2. Requirements and security analysis

4.2.1. Requirements analysis

Considering our design goals that have been discussed in Section 2.3, we outline in the following how these goals can be achieved using our proposed protocol.

Table 2
Efficiency comparison.

Our protocols	Number of operations and running time m/s						Total time
	Exponential		Multiplication		Pairing		
	No.	Time	No.	Time	No.	Time	
Original	1	10.64	0	0	2	40.08	50.72
Server-aided	2	21.64	1	6.38	0	0	27.66

- *Anonymity*: In payment phase, the real identities of Alice ID_A has been kept confidential in all steps. Alice's real identity is converted into two random pseudo identities P_{ID_A} and P'_{ID_A} for unknown $x \in_R Z_p^*$. Furthermore, the pseudo identity pair (P_{ID_A}, P'_{ID_A}) is an ElGamal-type ciphertext, which is secure against chosen-plaintext attacks (CPA). Therefore, there is no way to extract or even guess the real identity from the pseudo identity pair without knowing the master-key s as illustrated in traceability process below.
- *Traceability*: Given the pseudo identity pair (P_{ID_i}, P'_{ID_i}) and master-key s, only $\mathcal{PSP}$ can trace the real identity of the user by computing $P'_{ID_i} \oplus H_1(sP_{ID_i}) = ID_i \oplus H_1(xP_{pub}) \oplus H_1(sxP) = ID_i$. Therefore, our protocol can trace the malicious users effectively.
- *Non-repudiation*: As we mentioned in the traceability process above, merchant cannot repudiate the origin and the correctness of the receipt information. Also no customer can deny his/her confirmed payment.
- *Unforgeability*: Suppose that the (t', ε') CDH assumption holds in a group $\mathbb{G}_1$. As is proved in [13], our proposed protocol is strongly unforgeable against chosen message attacks for any t and ε satisfying:

$$\varepsilon' \geq \frac{1}{\varrho^3 q_{H_1}(q_\sigma + 1)}\varepsilon,\ t' \leq t + c_{(\mathbb{G}_1,\mathbb{G}_2)}(q_{H_1}, q_{H_2}, q_\sigma + 1)$$

where q_{H_1} times partial private key H_1 queries, q_{H_2} times public key H_2 queries, q_σ times signature queries, and ϱ is the base of natural logarithm.

5. Conclusion

In this paper, we have explained the reason why mobile wallet needs to be secured, identified the security threats and formalized the security requirements for the mobile wallet. After that, we have proposed a secure mobile wallet by incorporating the digital signature and pseudo-identity techniques. To offload the heavy verification cost to the untrusted cloud server, the secure outsourcing verification has also been used. Finally, we have analyzed the security and evaluated the performance of our proposal. The analysis and results demonstrate that our protocol is practical and secure. The tracing and revocation of malicious user can be regarded as our future work.

Acknowledgments

This work was supported in part by the National Science Foundation of China (No. 61133016, No. 61300191, No. 61202445, No. 61272527, No. 61602096 and No. 61370026), the National High Technology Research and Development Program of China (No. 2015AA016007), the Sichuan Key Technology Support Program (No. 2014GZ0106), Science and Technology Project of Guangdong Province (No. 2016A010101002) and the National Science Foundation of China-Guangdong Joint Foundation (No. U1401257).

References

[1] D.L. Amoroso, R. Magnier-Watanabe, Building a research model for mobile wallet consumer adoption: the case of mobile suica in Japan, J. Theor. Appl. Electron. Commer. Res. 7 (1) (2012) 94–110.
[2] S. Smith, B. Rackley, B. Ackerman, N. Rainey, W. Porter, A. Osei, R. Dessert, K. Iyer, K. Cochran, J. Mason, et al., System and method of conducting transactions using a mobile wallet system, US Patent App. 12/562,593 (May 20 2010). ⟨https://www.google.com/patents/US20100125510⟩
[3] T.M. Research, Mobile wallet market - global industry analysis, size, share, growth and forecast 2012 - 2018, Online; (accessed 15.02.16), 2013. ⟨http://www.transparencymarketresearch.com/mobile-wallet.html/⟩.
[4] R.L. Rivest, A. Shamir, L.M. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Commun. ACM 21 (2) (1978) 120–126.
[5] S. Goldwasser, S. Micali, R.L. Rivest, A digital signature scheme secure against adaptive chosen-message attacks, SIAM J. Comput. 17 (2) (1988) 281–308.
[6] F. Hess, Efficient identity based signature schemes based on pairings, in: Selected Areas in Cryptography, Proceedings of the 9th Annual International Workshop, SAC 2002, 2002, pp. 310–324.
[7] S.S. Al-Riyami, K.G. Paterson, Certificateless public key cryptography, in: Advances in Cryptology - ASIACRYPT 2003, Proceedings of the 9th International Conference on the Theory and Application of Cryptology and Information Security, 2003, pp. 452–473.
[8] X. Huang, W. Susilo, Y. Mu, F. Zhang, On the security of certificateless signature schemes from asiacrypt 2003, in: Proceedings of the 4th International Conference Cryptology and Network Security, CANS, 2005, pp. 13–25.
[9] Y. Yu, Y. Mu, G. Wang, Q. Xia, B. Yang, Improved certificateless signature scheme provably secure in the standard model, IET Inf. Secur. 6 (2) (2012) 102–110.
[10] H. Xiong, Cost-effective scalable and anonymous certificateless remote authentication protocol, IEEE Trans. Inf. Forensics Secur. 9 (12) (2014) 2327–2339.
[11] H. Xiong, Z. Qin, F. Li, An improved certificateless signature scheme secure in the standard model, Fundam. Inform. 88 (1–2) (2008) 193–206.
[12] D.-H. Shin, Towards an understanding of the consumer acceptance of mobile wallet, Comput. Hum. Behav. 25 (6) (2009) 1343–1354.
[13] X. Huang, Y. Mu, W. Susilo, D.S. Wong, W. Wu, Certificateless signatures: new schemes and security models, Comput. J. 55 (4) (2012) 457–474.
[14] X. Chen, J. Li, X. Huang, J. Li, Y. Xiang, D. Wong, Secure outsourced attribute-based signatures, IEEE Trans. Parallel Distrib. Syst. 25 (12) (2014) 3285–3294.
[15] V. Coskun, B. Ozdenizci, K. Ok, A survey on near field communication (NFC) technology, Wirel. Pers. Commun. 71 (3) (2013) 2259–2294.
[16] C. Zhang, R. Lu, X. Lin, P.H. Ho, X. Shen, An efficient identity-based batch verification scheme for vehicular sensor networks, in: INFOCOM 2008. Proceedings of the 27th Conference on Computer Communications. IEEE, 2008, pp. 816–824.
[17] B.H. Kim, K.Y. Choi, J.H. Lee, D.H. Lee, Anonymous and traceable communication using tamper-proof device for vehicular ad hoc networks, in: 2007. International Conference on Convergence Information Technology, 2007, pp. 681–686.
[18] M. Girault, D. Lefranc, Advances in Cryptology - ASIACRYPT 2005 in: Proceedings of the 11th International Conference on the Theory and Application of Cryptology and Information Security, Chennai, India, December 4-8, 2005. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, Ch. Server-Aided Verification: Theory and Practice, pp. 605–623.
[19] W.D. Chen, K.E. Mayes, Y.H. Lien, J.H. Chiu, NFC mobile payment with citizen digital certificate, in: Proceedings of the 2nd International Conference on Next Generation Information Technology (ICNIT), 2011, pp. 120–126.
[20] E.J. Steffens, A. Nennker, Z. Ren, M. Yin, L. Schneider, The sim-based mobile wallet, in: ICIN 2009. Proceedings of the 13th International Conference on Intelligence in Next Generation Networks., 2009, pp. 1–6.
[21] H. Harb, H. Farahat, M. Ezz, Securesmspay: Secure SMS mobile payment model, in: ASID 2008. Proceedings of the 2nd International Conference on Anti-counterfeiting, Security and Identification, 2008, pp. 11–17.
[22] M. Toorani, A. Beheshti, SSMS - a secure SMS messaging protocol for the m-payment systems, in: ISCC 2008. IEEE Symposium on Computers and Communications., 2008, pp. 700–705.
[23] N. Park, J. Kwak, S. Kim, D. Won, H. Kim, WIPI mobile platform with secure service for mobile RFID network environment, Vol. 3842 LNCS, 2006, pp. 741–748.
[24] W. Liu, C. Zhao, W. Zhong, Z. Zhou, F. Zhao, X. Li, J. Fu, K. Kwak, The GPRS mobile payment system based on RFID, in: ICCT '06. International Conference on Communication Technology, 2006, pp. 1–4.
[25] M. Hassinen, K. Hyppönen, K. Haataja, Emerging Trends in Information and Communication Security: International Conference, ETRICS 2006, Freiburg, Germany, June 6-9, 2006. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, Ch. An Open, PKI-Based Mobile Payment System, pp. 86–100.
[26] T. Bamert, C. Decker, R. Wattenhofer, S. Welten, Security and Trust Management in: Proceedings of the 10th International Workshop, STM 2014, Wroclaw, Poland,

September 10-11, 2014. Proceedings, Springer International Publishing, Cham, 2014, Ch. BlueWallet: The Secure Bitcoin Wallet, pp. 65–80.
[27] J. Meng, L. Ye, Secure mobile payment model based on WAP, in: WiCOM '08. Proceedings of the 4th International Conference on Wireless Communications, Networking and Mobile Computing., 2008, pp. 1–4.
[28] P. Huang, A.C. Boucouvalas, Future personal e-payment: IrFM, IEEE Wirel. Commun. 13 (1) (2006) 60–66.
[29] M.W. Khan, SMS security in mobile devices: a survey, Int. J. Adv. Netw. Appl. 5 (2) (2013) 1873–1882.
[30] A. Juels, RFID security and privacy: a research survey, IEEE J. Sel. Areas Commun. 24 (2) (2006) 381–394.
[31] B. Alomair, R. Poovendran, Privacy versus scalability in radio frequency identification systems, Comput. Commun. 33 (18) (2010) 2155–2163.
[32] J. Padgette, K. Scarfone, L. Chen, Guide to Bluetooth Security: Recommendations of the National Institute of Standards and Technology (Special Publication 800-121 Revision 1), CreateSpace Independent Publishing Platform, USA, 2012.
[33] P. Soni, M-payment between banks using SMS [point of view], in: Proceedings of the IEEE, 98 (6), 2010, pp. 903–905.
[34] Google, Google wallet, ⟨https://www.google.com/wallet/⟩.
[35] D. He, J. Chen, R. Zhang, An efficient and provably-secure certificateless signature scheme without bilinear pairings, Int. J. Commun. Syst. 25 (11) (2012) 1432–1442.