Electron Commer Res
https://doi.org/10.1007/s10660-018-9302-2

Strong anonymous mobile payment against curious third-party provider

Chenglong Cao[1] · **Xiaoling Zhu**[2]

Abstract M-commerce provides convenient services and has developed rapidly in recent years. But security and privacy have always been major concerns for most users. Among existing payment systems, PayPal as well as Alipay has a third-party payment provider (TPP) but does not provide anonymity. Bitcoin provides anonymity but its decentralized framework without TPP causes high energy consumption and security attack issues. Further information can be deduced from the public decentralized ledger, Bitcoin cannot offer strong privacy guarantees. Therefore, unifying strong anonymity, security and efficiency is challenging in mobile payment. This paper proposes a strong anonymous mobile payment against a curious third-party provider (SATP). A ticket as a new means of payment is partially blindly signed by TPP using certificateless cryptographic primitives. SATP can ensure confidentiality of payment data, non-repudiation and revocation of payment operation, and anonymity of payer's identity. Especially, it can enable a user to pay anonymously even in face of a curious TPP. Performance analysis shows that SATP avoids high energy consumption like Bitcoin, and its communication cost is less than that of the existing anonymous research work.

Keywords Mobile payment · Ticket · Strong anonymous · Revocation · Partially blind signature

✉ Chenglong Cao
chenglongcao@sina.cn

Xiaoling Zhu
zhuxl@hfut.edu.cn

[1] Anhui Finance and Trade Vocational College, Hefei 230601, China

[2] School of Computer and Information, Hefei University of Technology, Hefei 230009, China

Published online: 31 March 2018

1 Introduction

As mobile terminals are widespread, m-commerce comes into the people's lives. It breaks the constraints of time and space and allows people to make transactions anywhere and anytime [1]. It brings great conveniences to consumers.

Payment plays an important role in m-commerce and current prevailing payment systems are based on a third-party payment provider (TPP). As an intermediary between merchants and customers, TPP integrates multiple payment ways with different bank cards into a unified interface and makes m-commerce more convenient and faster. PayPal has a TPP and it is used by many users around the world. Paypal does not provide anonymity and it can obtain more details other than the order and the receiving merchant. Preibusch et al. [2] analyzed the 881 most popular US web shops and found that more than half of the sites shared with PayPal customers' details including their names, emails, postal addresses and their bank accounts. Also, sites sell PayPal sensitive product details such as adult toys and medication. And even worse, PayPal forwards shopping details of customers to Omniture, a third party data aggregator with even larger tracking reach than PayPal. In China, Alipay is a prevailing payment tool. As a TPP, it also provides the guarantee service of transactions on Taobao website, which is the largest e-commerce platform in China. Alipay can monitor consumption choices, obtain product name, buyer's delivery address, and purchase time. Further, it can build up a fine-grained and comprehensive consumption profile. In addition, the payment using PayPal or Taobao usually links to a credit card number or a bank account, which serves as a persistent identifier. Using the persistent identifier, PayPal or Taobao can link multiple transactions of different shopping websites to one buyer. So, it is a more serious privacy leaks issue than browsing web.

Another different type is Bitcoin, which was created in a 2008 academic paper by a still unknown person using the name Satoshi Nakamoto [3]. It is a cryptographically secure decentralized peer-to-peer electronic payment system. It provides anonymity but does not use a TPP. The trust in Bitcoin is implemented by a public digital ledger called blockchain. This ledger keeps records for all coin transactions, which is obtained by all Bitcoin network nodes. The job of keeping the system running is left to a volunteer workforce known as Bitcoin miners. The calculations are so intense that computers used for calculations emit the heat keeping the room at a high temperature. Therefore, high energy consumption becomes a main weakness of Bitcoin.

Moreover, since Bitcoin is a decentralized model with an uncontrollable environment, there exists security threats: (1) 51% hashpower attack. 51% hashpower attacker controls more than 50% of computational power and he can weakens the effectiveness of Bitcoin protocol. He can start computing from the recent block checkpoint. The calculation of new block chain has been hidden until its length is two more than the original. According to Bitcoin network protocol at present, the original block chain will be replaced by a new one. (2) DDOS attacks. Their majority targets are large mining pools. Through exhausting the network resources, a large number of mining nodes are forced to go offline. (3) Anonymity

Attack. Nowadays, there's a steady shift toward seeing cryptocurrency as a tool for prosecuting crimes [4]. As we know, the public blockchain reveals all the transaction data to any user connected to the Bitcoin network. Its associated data can create a forensic trail that consists of entire financial history information. In [5], the graph method using heuristics was offered. First, by traversing the transaction graph, the relationship between various input and output addresses can be inferred. And using these relations, an address graph is generated. Next, by using the address graph along with a number of heuristics, an entity graph is generated, where the grouping address probably belongs to the same user. Further, researchers follow the money by a robust blockchain analysis.

Our work In order to address the aforementioned issues and unify strong anonymity, security and efficiency in mobile payment, we propose a strong anonymous mobile payment against a curious third-party provider (SATP). In the scheme, a ticket as a new payment tool is constructed based on partially blind signature. Unlike Paypal and Alipay, SATP can resist privacy attacks from service providers, other users and TPP. Unlike Bitcoin, it has not a decentralized ledge to be maintained. So it avoids high energy consumption and 51% hashpower attack in Bitcoin system. Also, all transaction data are transmitted securely, instead of being stored in the public blockchain. Therefore, SATP offers better privacy guarantees and has lower computation cost than Bitcoin.

A possible payment scene is shown in Fig. 1. In the scene, a mobile phone, a PDA, a vehicle terminal or other mobile devices are used to buy some services or digital goods such as e-book, music, and games. And a new payment tool, Ticket, is required to be installed. Then, customers use Ticket to make payment. During the transactions, customers (users), service providers and TPP have different security and privacy requirements. More concretely, for a user, he is worried that his identity is known by TPP, a service provider and other users. In practice, there was a survey administered at an International Airport in a major U.S. city, and the results show that 16% of users are worried about privacy disclosure, including sharing (selling,

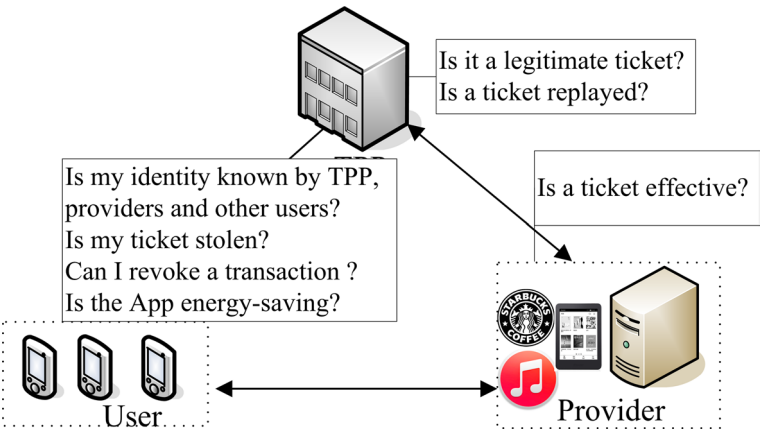

Fig. 1 Payment scenario in SATP

renting) personal information to other companies, and being contacted by the company without providing consent [6]. When a user makes payment by a ticket, he is worried whether his ticket is stolen. Sometimes, he expects that he can revoke the transaction if the service provider has not offered the request services. In addition, system performance is also an important issue. Since a mobile terminal has lower power compared with the PC, the solution with high computation and communication costs is not suitable for mobile payment. For the TPP, it wants to know whether the payment is authorized. It should distinguish a legitimate ticket from a forged ticket sent by a malicious user or a service provider. Also, it wants to ensure that the ticket has not been replayed because a message is easily tapped and copied under the wireless network environment. For the service provider, it wants to know whether the ticket is effective.

The remainder of this paper is organized as follows. Firstly, we introduce related works in Sect. 2, which will emphasize the motivation of our work. We then describe secure framework of our solution in Sect. 3 and the proposed SATP scheme in details in Sect. 4. We analyze security and performance in Sects. 5 and 6, respectively. Finally, we conclude the paper in Sect. 7.

2 Related works

Some research works have been conducted in recent years, and their concerned issues focus on security, privacy and efficiency in mobile payment.

Security countermeasures SET and i-kp protocols are secure electronic payment protocols. The SET protocol [7] uses digital signature technology and complicated transaction process to protect payment messages. *i*-kp [8] is a family of secure protocols based on public key cryptosystem, including 1-kp, 2-kp and 3-kp, and their security levels are gradually increasing. The 1-kp protocol is the simplest and cannot provide non repudiation for messages from users and merchants. In the 2-kp protocol, the user can confirm the authenticity of the message from the merchant. The 3-kp protocol provides multiparty security and non repudiation for messages from payment gateways, users, and merchants. The two protocols both have security properties: confidentiality, integrity and authorization. They are based on public key infrastructure (PKI). First, participants need to apply for public key certificates from certification center (CA). During the transaction process, a large number of operations, such as encryption, decryption, and digital signatures and verifications, are required, as well as transmission of certificates. So communication and computation overheads of the protocols [7, 8] are high. Since mobile terminals are resource-constrained devices, unlike web browsing of personal computer, these protocols are not suitable for them. More studies [9] have considered the following methods to provide authentication: password, symmetric and asymmetric cryptography. Han et al. [10] proposed an identity-based plaintext checkable encryption scheme where anyone can check whether a cipher text is the encryption of a plaintext under a specific identity; then, it was incorporated into the m-commerce scenario, and an

accountable m-commerce (AMEC) scheme was given; AMEC does not provide anonymity because the identities of a user and a merchant are public.

Privacy countermeasures Using symmetric-key operations, Isaac et al. [11] designed and implemented a lightweight secure payment protocol in VANETs, where a merchant cannot communicate directly with his financial institution to process a payment request. The scheme prevents a merchant from knowing the identity of a client, but it cannot prevent an issuer from knowing it. Isern-Deya et al. [12] presented a new payment scheme to access location-based services. The scheme implements a fair exchange between a provider and a user, and it provides refund services. It achieves user anonymity, but it cannot ensure payer's anonymity in face of a bank. Bitcoin is a cryptocurrency and popular payment system. Its transactions take place between users, without a TPP, and transaction data are recorded in a public distributed ledger called a blockchain. Bitcoin is anonymous, meaning that funds are not tied to real-world entities directly but rather Bitcoin addresses. However, the public blockchain reveals all the transaction data to any user connected to the Bitcoin network. Nowadays, graph methods using heuristics have been offered to follow the money through a robust blockchain analysis. From this point of view, Bitcoin does not provide strong anonymity.

Efficiency countermeasures In order to decrease the computational cost, symmetric key cryptography and certificate less signature (CLS) are used in mobile payment. Sekhar et al. [13] designed a secure lightweight mobile payment protocol, where two parties use the shared key to communicate secretly and either side can deny his own behavior; it has repudiation. Gong et al. [14] presented a CLS scheme to pursue robustness and efficiency. Under the hardness of the ECDLP, Yeh et al. [15] introduced a new CLS scheme that is more efficient than those in the past; but it does not provide user anonymity.

The schemes [7, 8] based on PKI require digital certificates, and their computation and communication costs are generally high due to a large number of encryption, decryption, signature and verification operations, as well as transmission of certificates. Identity-based cryptosystem (IBC) is presented by Shamir [16], where the public key certificate is not required. IBC has clear advantage over general PKI schemes in communication costs. On the other side, a blind signature method provides good privacy protection of a message since an issuer does not know the message he signs. But a blind signature may cause illegal use of signature by a malicious applicant. Partially blind signature was proposed, where common information is embedded into the signature by the signer. Therefore, a partially blind signature based on IBC is used in our scheme to solve strong anonymity and security issues of mobile payment.

Partially blind signature There have been some research results about partially blind signature. Zhang et al. [17] proposed a partially blind signature scheme from bilinear pairings. The signature result is short; but it lacks randomness. Chow et al.

[18] proposed a randomized partially blind signature scheme that includes four steps: challenge, blindness, signature, and verification. Li et al.'s scheme [19] has existential unforgeability under the computational Diffie–Hellman assumption in the standard model. It also requires four steps, the same as Chows's. The above partial blind protocols [17–19] do not provide the function of revocation.

So we design a new partially blind signature protocol, which decreases steps and provides revocation function. Based on the protocol, a secure and efficient mobile payment scheme is proposed. The main contributions are: (1) it ensures confidentiality and integrity of payment data, and non-repudiation of payment operation. (2) It offers strong privacy guarantees: no matter TPP, a payee or a malicious user, no one knows the identity of a payer from transaction messages. (3) When a user has not obtained the related service, TPP can refund the money to the user. It means that transactions can be revoked.

3 Overview of the SATP scheme

In the Kerberos protocol, the concept of a ticket is introduced as an access credential of a client. We use the concept as a payment credential. In our scheme, a ticket is similar to an e-voucher, but it has a wider application range. When a user pays for the service, he sends the ticket to the service provider. Then, the provider returns the related service if it accepts the ticket. When the provider needs to cash the ticket, it sends the ticket to TPP. Further, payment model, security requirements and some related cryptographic primitives are presented.

3.1 Payment model

The involved main parties in SATP are as follows and the notations are described in Table 1.

- User (U): A user is equipped with a smart phone, a PDA or a tablet. He can access the Internet to buy services. His identity, public key and private key are denoted as ID_U, Q_U and D_U, respectively. ID_U may be his phone number or email address.
- Service provider (P): A service provider sells various services. Its identity, public key and private key are denoted as ID_S, Q_S and D_S, respectively.
- Third-party payment provider (TPP): TPP is an independent agency to protect the interests of both trading parties. It manages payment accounts and provides online money transfer services. Its identity, public key and private key of TPP are denoted as ID_T, Q_T and D_T, respectively.
- Private key generator (PKG): U, P and TPP need to obtain their private keys from PKG. During the system initialization, PKG randomly picks a number $a \in Z_q^*$ as its private key, computes and broadcast its public key $P_{pub} = aP$. Any entity needs to register at the PKG and then gets the private key.

Table 1 Notation

Notation	Description
(Q_U,D_U), (Q_P,D_P), (Q_T,D_T)	Public and private key of U, P and TPP, respectively
(ID_U, V_U)	Identity and pseudonym of U
$\alpha=(V_U, ID_P)$	Pseudonym of U and identity of P
$\beta=(N, t)$	Denomination and expiration date of a ticket
$Ticket$ (α, β, R, S)	Signature (R, S) of (α, β)
$TList_T$, $TList_P$	Used ticket list of TPP and P, respectively
$E(\cdot)$, $D(\cdot)$, K	AES encryption and decryption algorithms, session key
Enc(·), Dec(·)	Encryption and decryption algorithms using IBC
Signcrypt(·), Verifydecrypt(·)	Sign then encryption, verify then decryption using IBC
H_1, H_2, H_3, H_4, H_5, H_6	Collision-resistant hash functions
G_1, G_2	q-order additive and multiplicative group, respectively
P	Generator of G_1
P_{pub}, a	Public and private key of PKG, respectively

- Ticket: A ticket includes U's pseudonym, P's identity, denomination and expiration date of a ticket, and a signature. When paying, U sends a ticket to P. Then, P uses the ticket for cashing. A success payment requires a valid ticket. The ticket is valid means it is not expired, unused and has a right signature.

Payment process in SATP is as follows (Fig. 2).

1. U sends a request for a service to P; P responds the price of the service to U.
2. U sends a request for a ticket to TPP; TPP responds a ticket to U. More specifically, U's identity requires to be verified. Then, TPP deducts the same amount money as the amount requested by U from U's account. Next, TPP embeds a common information into the ticket and issues the ticket using partially blind signature.
3. U Shows the ticket to P; P provides the requested service.
4. P cashes the ticket at TPP before the expiration date in order to withdraw money.

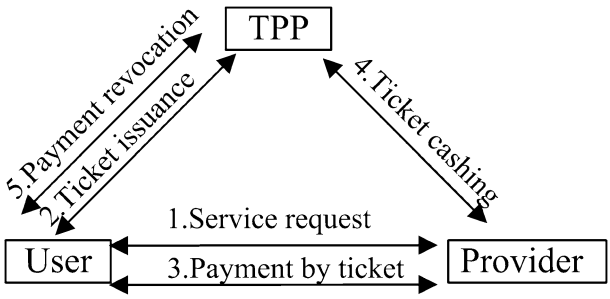

Fig. 2 Payment framework using a ticket

5. (Optional) U may cancel the payment by revoking the ticket if he does not obtain the service.

3.2 Security requirements

We assume that PKG is honest, and it neither disclose nor tamper private keys. We also assume that TPP does not initiate an active attack: it does not forge a ticket for itself or other entities, and it does not maliciously transfer money from A's account to B's account. But TPP is curious. Besides account data, TPP might want to get more information such as purchase behavior of users. To achieve this, it might even conspiracy with merchants. On the other hand, we assume that U and P might initiate active and passive attacks for their own benefits. They might eavesdrop on the communication to obtain trading information or forge a signature to obtain illegal money. So the following security and privacy properties are required.

Authentication It is computationally infeasible for an attacker to apply for, use or cash a ticket successfully when he impersonates a user or a service provider. For example, TPP issues a ticket to U only if U passes identity authentication and U's cash account balance is greater than the denomination of the ticket.

Strong anonymity It is computationally infeasible for attackers to restore U's identity, even under the collusion of P and TPP.

Partial blindness It is computationally infeasible for TPP to obtain *U's* identity by associating the signature result with the specific signing process. Meanwhile, the common information is embedded by TPP and cannot be removed from the signature.

In the completely blind signature protocol, the signer digitally signs the file without knowing the content of the file, and the signer cannot associate the signature process with the final signature result. Completely blind signature protects requester privacy, but easily causes the signature used illegally, i.e., a requester provides an illegal file with malicious content to the signer, and yet the signer blindly signs the file. In the partially blind signature protocol, a message is divided into two parts: one is the blind content; the other is the common important information known by a signer and requester, which is embedded in the signature by the signer, such as the amount of money and deadline.

Unforgeability It is computationally infeasible for P and other users to forge a valid ticket that belongs to U.

3.3 Identity-based cryptography

Identity-based cryptography (IBC) does not require digital certificate, and its communication overhead is less than the traditional public key cryptosystem. Based on Boneh-Franklin IBC [20], the following encryption and signcryption algorithms are proposed to ensure confidentiality and authentication of transactions.

First, define a bilinear map $e : G_1 \times G_1 \rightarrow G_2$, where G_1 and G_2 are q-order additive and multiplicative group, respectively. And P is a generator of G_1.

Let $H_1 : \{0,1\}^* \to G_1$, $H_2 : \{0,1\}^* \times \{0,1\}^* \times G_1 \to Z_q^*$, $H_3 : G_2 \to \{0,1\}^*$, $H_4 : \{0,1\}^l \times \{0,1\}^* \to Z_q^*$, $H_5 : \{0,1\}^* \times G_1 \to Z_q^*$ and $H_6 : \{0,1\}^* \to \{0,1\}^l$ be collision-resistant hash functions. PKG generates the public system parameters $\{G_1, G_2, H_1, H_2, H_3, H_4, H_5, H_6, P, P_{pub}\}$ and sends $D_{ID} = aQ_{ID}$ to the applicant securely. Here, P_{pub} and a are public and private keys of *PKG*, respectively; *ID* is the applicant's identity, $Q_{ID} = H_1(ID)$ and D_{ID} are his public and private keys, respectively. Through the key distribution, TPP, P and U obtain the private keys D_T, D_P and D_U, respectively.

1. Encrypt and decrypt

For confidential communication, encryption and decryption are designed. Let $r \in Z_q^*$ be chosen randomly and define $Enc(Q_{ID}, m) = (c_1, c_2)$

$$(c_1, c_2) = (rP, H_3(w) \oplus m)) \tag{1}$$

where $w = e(Q_{ID}, P_{pub})^r$ and Q_{ID} is the public key of the receiver.

Define

$$Dec(D_{ID}, c_1, c_2) = c_2 \oplus H_3(w) \tag{2}$$

where $w = e(D_{ID}, c_1)$ and D_{ID} is the private key of the receiver.

2. Signcrypt and verifydecrypt

A sends the message m to B using signcrypt and verifydecrypt functions as follows.

Let $r \in Z_q^*$ be chosen randomly, and define $Signcrypt(D_A, Q_B, m) = (c, R, S)$

$$(c, R, S) = (H_3(w) \oplus m, rP, (r + H_2(m, ID_B, R))^{-1} D_A) \tag{3}$$

where $w = e(Q_B, P_{pub})^r$, Q_B is the public key of B, D_A is the private key of A, $(r + H_2(m, ID_B, R))^{-1}$ is the inverse, and (c, R, S) is the signed and encrypted result of the message m. We assume that $r + H_2(m, ID_B, R)$ is invertible in Z_q^*; otherwise, we choose r continuously until the invertibility condition is satisfied.

Further, define $Verifydecrypt(Q_A, D_B, c, R, S) = (m \text{ or } \perp)$. Here, if

$$e(R + H_2(m, ID_B, R)P, S) = e(P_{pub}, Q_A) \tag{4}$$

then

$$w = e(D_B, R),\ m = H_3(w) \oplus c \tag{5}$$

and return m; otherwise, return $\perp$.

4 Description of the SATP scheme

The SATP scheme is composed by six phases: registration, service request, ticket issuance, payment by ticket, ticket cashing and payment revocation (optional). The

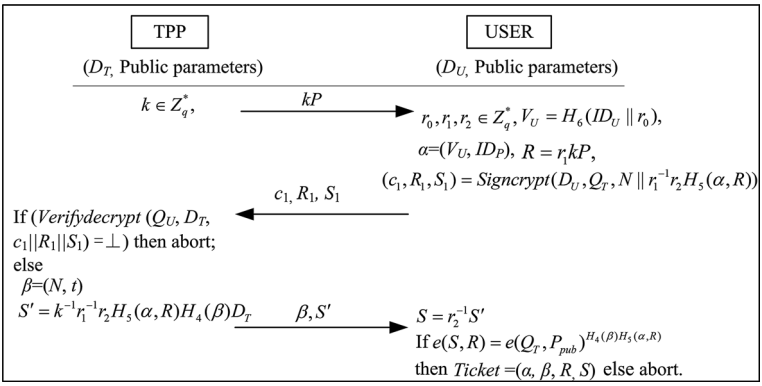

Fig. 3 Message exchange during ticket issuance

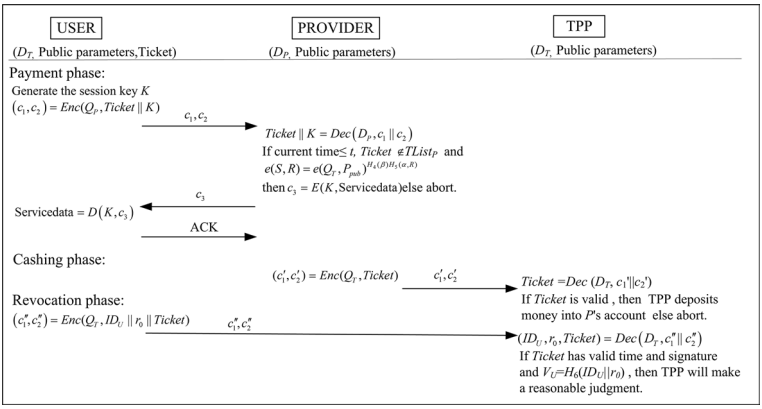

Fig. 4 Messages exchange during three phases: payment, cashing and revocation

registration protocol is to ensure that users and merchants have their accounts on the TPP. The service request protocol returns the price of services. The two protocols are relatively simple. For the remaining four protocols, the ticket issuance protocol is described in Fig. 3. Messages exchange processes during payment, cashing and revocation phases are demonstrated in Fig. 4.

4.1 Registration

TPP works as a debit system. *U* and *P* should setup their accounts on TPP. If *U* applies for a ticket with the denomination amount of *N* successfully, *U*'s account balance will be reduced by *N*. If *P* cashes the ticket with the same denomination successfully, then *P*'s account balance will be increased by *N*.

4.2 Service request

1. First, U sends a request to P. This request may be from the practical requirements when buying music and other electronic products, or ordering some services such as navigation services.
2. Then, P returns the service price N. Then, N is as the denomination amount of the ticket.

4.3 Ticket issuance

During the issuance of a ticket, U sends a request for a ticket for payment to TPP; TPP responds a ticket to U.

1. TPP chooses invertible element $k \in Z_q^*$ computes kP and sends it to U.
2. U receives kP and generates his pseudonym

$$V_U = H_6(ID_U||r_0) \tag{6}$$

where ID_U is the real identity of U and r_0 is a random number in Z_q^*. Further, he generates

$$\alpha = (V_U, ID_p) \tag{7}$$

where ID_p is the identity of P.

3. U randomly chooses invertible elements $r_1, r_2 \in Z_q^*$, computes $R = r_1 kP$, executes the algorithm $Signcrypt(D_U, Q_T, N||r_1^{-1} r_2 H_5(\alpha, R))$, gets (c_1, R_1, S_1) and sends them to TPP. Here, U's pseudonym and P's identity are blinded.
4. TPP receives (c_1, R_1, S_1) and executes the algorithm $Verifydecrypt$ $(Q_U, D_T, c_1, R_1, S_1)$. If the result is $\perp$, then it refuses and stops; else it gets $N||r_1^{-1} r_2 H_5(\alpha, R)$ and continues.
5. TPP extracts N from the message and determines the common information

$$\beta = (N, t) \tag{8}$$

where t is the deadline for the ticket cashing.

6. TPP makes a signature

$$S' = k^{-1} r_1^{-1} r_2 H_5(\alpha, R) H_4(\beta) D_T \tag{9}$$

Next, it deducts the amount N from U's account and sends (β, S') to U.

7. U removes the blind factor r_2 from S', and gets

$$S = r_2^{-1} S' = k^{-1} r_1^{-1} H_5(\alpha, R) H_4(\beta) D_T \tag{10}$$

U checks the equation

$$e(S, R) = e(Q_T, P_{pub})^{H_4(\beta)H_5(\alpha,R)} \tag{11}$$

If the equality does not hold, U aborts. Otherwise, U obtains the ticket

$$Ticket = (\alpha, \beta, R, S) \tag{12}$$

Here, (R, S) is the signature of the message (α, β). Equation (11) is obvious established because $e(S, R) = e(k^{-1} r_1^{-1} H_5(\alpha, R) H_4(\beta) D_T, r_1 kP) = e(Q_T, P_{pub})^{H_4(\beta)H_5(\alpha,R)}$.

Steps 3 and 4 are used to authenticate U's identity when U sends a request to TPP. In steps 5 and 6, TPP embeds the common information into the ticket and issues the ticket. Indeed, TPP knows the common information β, but β may be owned by different users. Meanwhile, TPP does not know α, R and S. So TPP cannot establish the association between (α, β, R, S) and a unique U. And the signature has blindness. Since β is embedded into the signature by the signer, the signature has partially blindness.

4.4 Payment by ticket

During the payment phase, U sends the ticket to P securely; P provides the service requested by U.

1. U randomly generates the 128-bit session key K for service data encryption, encrypts $Ticket||K$ using P's public key Q_P, gets the cipher $(c_1,c_2) = Enc(Q_P, Ticket||K)$ and sends it to P. Here, Enc is the public key cryptography algorithm based on IBC.
2. P receives (c_1,c_2), executes the algorithm Dec (D_P, c_1, c_2) and gets $Ticket||K$.
3. P checks whether the ticket is within the validity period. If passed, P determines whether the ticket has been in the used ticket list $TList_P$. If it is not used, then continues; else aborts.
4. P further decides whether the validity of the signature on the ticket. He checks (11). If not passed, P refuses to provide the service and aborts; otherwise, P stores the ticket in $TList_P$ and continues.
5. P executes AES algorithm, gets the cipher of service data $c_3 = E$ (K, Servicedata) and sends it to U.
6. U opens the cipher using the session key K, obtains the related service and sends an acknowledgement to P.

In Step 4, the size of $TList_P$ will expand with the increase of the number of the users. To reduce the storage and search costs, P may periodically delete the expired tickets. P checks the deadline of the ticket before P determines whether the ticket is in $TLIST_P$. So the deletions will not affect the validity checking. In Step 5, the service data is transmitted securely using the AES algorithm. The key exchange

between U and P is completed by Step 1. U sends the chosen key K to P using the $Enc()$ with the public key Q_P of P. So the session key K is shared by U and P.

4.5 Ticket cashing

P cashes the ticket at TPP before the deadline t, and further withdraws money.

1. Similarly to Sect. 4.4, P encrypts the ticket using TPP's public key Q_T, and sends it to TPP.
2. TPP decrypts it and gets the ticket. If it passes the validity checking (time, signature, and usage), TPP deposits the same denomination amount of money to P's account.

4.6 Payment revocation (optional)

U sends the ticket to P, but P does not provide U with the service. Under the circumstances, the scheme provides the function of transaction revocation. If there is a dispute between U and P, TPP will make a reasonable judgment and the specific process is as follows.

1. U encrypts his identity ID_U, the initial secret r_0 and the ticket, and then sends the cipher to TPP.
2. TPP executes the algorithm Dec (D_P, c_1'', c_2'') and gets the parameters $ID_U||r_0||Ticket$. If $Ticket$ is valid, TPP further checks whether ID_U and r_0 satisfy $H_6(ID_U, r_0) = V_U$.

Indeed, once the revocation protocol is executed, TPP can link U's identity ID_U with his pseudonym V_U and anonymity is spoiled.

3. If the checks all pass, then TPP refunds the money to U. But if other conditions pass except that the ticket is used, both U and P should provide relevant evidences to TPP. Specifically, U provides the evidence that the service has not been obtained; P provides the evidence that the service has been offered. If P is dishonest, money is transferred from P's account to U's account. Otherwise, it will remain unchanged. Furthermore, a dishonest buyer or seller will be blacklisted.

As a last note, the checks for ticket validity in Sects. 4.4, 4.5 and 4.6 include three aspects. (1) Time is valid. Adjusting t can control the number of valid tickets and reduce the storage and search overheads. But when t is small, P needs to cash the ticket in time, which improves real-time requirements for transactions. (2) Check the signature to ensure that the ticket is issued by TPP. (3) Determine that the ticket has not been used by judging $Ticket \notin TList_P$ and $Ticket \notin TList_T$. Here, $TList_P$ and $TList_T$ are the used ticket list of P and TPP, respectively.

5 Security analysis

Definition 1 (*Discrete logarithm (DL) problem*). Given $P \in G_1$, xP, find $x \in Z_p^*$.

Definition 2 (*DL assumptions*) The advantage of any probabilistic polynomial time algorithm in solving the DL problem is negligibly small, i.e., DL problem is assumed to be difficult to solve.

Proposition 1 *Our scheme has strong anonymity as long as none of U's tickets are revoked even in the face of the TPP, but not the PKG.*

Proof

(1) In the ticket issuance protocol, the user is anonymous in the face of TPP. U randomly chooses three random numbers r_0, r_1, r_2. And then he sends $r_1^{-1} r_2 H_5(\alpha, R)$ to TPP. We assume H_5 and H_6 are random oracles, the signer is difficult to obtain $\alpha = (V_U, ID_P)$ from $r_1^{-1} r_2 H_5(\alpha, R)$ and further difficult to obtain ID_U from $V_U = H_6(ID_U \| r_0)$. The signer cannot link a valid signature (α, β, R, S) to the view $r_1^{-1} r_2 H_5(\alpha, R)$ because of the randomness of r_1 and r_2. Meanwhile, the signature contains the common information, which is embedded by the signer and cannot be removed from the signature. Hence the ticket issuance protocol has the partial blindness property.
(2) During the payment phase, U sends the ticket to P; during the ticket cashing phase, P sends the ticket to TPP. The ticket includes *U's* pseudonym $V_U = H_6(ID_U || r_0)$. Assume H_6 is a random oracle and it has good one-way property, P does not know *U's* real identity, the same as *TPP*. So the user is anonymous in the face of P and *TPP*.
(3) During the ticket revocation phase, U sends $ID_U || r_0 || Ticket$ to TPP in order to prove that he is the person who applies for the ticket. After the revocation, his identity is leaked.

If there is no revocation, the identity of U is not known by P, *TPP* and any eavesdropper. The scheme has strong anonymity.

Proposition 2 *Our scheme has the existential unforgeability against adaptive chosen messages attacks under the random oracle model and DL assumption.*

Proof

(1) The blind signature protocol in Sect. 4.3 has the existential unforgeability. In order to prove the conclusion, assume that the challenger C receives an instance (S, xS) of the DL problem, his goal is to compute x. Let $\mathcal{A}$ be a probabilistic polynomial Turing machine to find a valid signature. C gives $\mathcal{A}$ public parameters, runs $\mathcal{A}$ as subroutine and acts as $\mathcal{A}'s$ challenger. The simulation is as described below.

- $H_1(ID_i)$: C checks whether there is a tuple (ID_i, Q_i) in list L_1. If it exists, C returns Q_i to $\mathcal{A}$. Otherwise, C chooses a random number x and returns $Q_i = xP$. Then, add (ID_i, Q_i) to L_1.
- $H_2(m, ID_i, R)$: C checks whether there is a tuple (m, ID_i, R, h_2) in L_2. If it exists, C returns h_2. Otherwise, C chooses a random number $h_2 \in Z_q^*$, adds (m, ID_i, R, h_2) to L_2 and returns h_2.
- $H_3(w)$: C checks whether there is a tuple (w, h_3) in L_3. If it exists, C returns h_3. Otherwise, C chooses a random l bits number h_3, adds (R, h_3) to L_3 and returns h_3.
- $H_4(\beta)$: C checks whether there is (β, h_4) in L_4. If it exists, C returns h_4. Otherwise, C chooses randomly a number $h_4 \in Z_q^*$, adds (β, h_4) to L_4 and returns h_4.
- $H_5(\alpha, R)$: C checks whether there is (α, R, h_5) in L_5. If it exists, C returns h_5. Otherwise, C chooses randomly a number $h_5 \in Z_q^*$, adds (α, R, h_5) to L_5 and returns h_5.
- $H_6(ID||r_0)$: C checks whether there is ($ID||r_0$, h_6) in L_6. If it exists, C returns h_6. Otherwise, C chooses randomly a l bits number h_6, adds ($ID||r_0$, h_6) to L_6 and returns h_6.
- Extract (ID_i): If $ID_i = ID_T$, return stop simulation. Otherwise, get (ID_i, Q_i) through H_1 and return $D_i = aQ_i$.
- BSign (m, ID_i): $ID_i \neq ID_T$: Get the private key D_i by running extract oracle. Choose invertible elements $r_1, k \in Z_q^*$. Compute $R = r_1 kP$. Get a tuple (β, h_4) through H_4 and (α, R, h_5) through H_5. After signing blindly and removing blind factor, output $S = k^{-1} r_1^{-1} H_4(\beta) H_5(\alpha, R) D_T$.Finally, return the result (α, β, R, S).
- $ID_i = ID_T$: Compute $R = H_5(\alpha, R) P_{pub}$ and $S = H_4(\beta) Q_T$. Then return the result (R, S).
- BVerify (α,β, R, S, ID_T):

Verify $e(S, R) = e(Q_T, P_{pub})^{H_4(\beta) H_5(\alpha, R)}$. If the verification passes, C returns true. Otherwise, C returns false.

For each forged signature of (α, β, R, S), where $R = H_5(\alpha, R) P_{pub}$. We observe that collisions of H_5 queries happen with negligible probability. Therefore, the above simulator cannot be distinguished from the legitimate signer. And then, it follows from the forking lemma [21] that if $\mathcal{A}$ is a sufficiently efficient forger, then we can construct a machine $\mathcal{A}'$ that outputs two signed messages (R, h_5, S) and (R, h_5', S') with $h_5 \neq h_5'$.

Finally, we construct a machine C' to solve the DL problem as follows.

1. C' runs $\mathcal{A}'$ to obtain two distinct forgeries, suppose they are $(\alpha, \beta, R, h_5, S)$ and $(\alpha, \beta, R, h_5', S')$
2. Because (R, h_5, S) and (R, h_5', S') satisfy $e(S, R) = e(Q_T, P_{pub})^{h_4 h_5}$ and $e(S', R) = e(Q_T, P_{pub})^{h_4 h_5'}$, respectively. It means that $e(h_5^{-1} S, R) = e(Q_T, P_{pub})^{h_4}$ and $e(h_5'^{-1} S', R) = e(Q_T, P_{pub})^{h_4}$. And $e(h_5^{-1} S, R) = e(h_5'^{-1} S', R)$. So $h_5^{-1} S = h_5'^{-1} S'$. Then $S' = h_5' h_5^{-1} S$. Thus,$x = h_5' h_5^{-1}$ as a solution of the DL problem: given S, xS, find x. It is in contradiction with the DL assumption.

Therefore, if there is an adversary who can succeed in such existential forgery attack with non-negligible advantage, that means there is an algorithm to solve the DL problem with non-negligible advantage. The scheme is secure against any existential forgery under chosen message attack under the random oracle model and DL assumption.

2. In Sect. 3.3, signcryption is constructed. Like the above proof, it has also the existential unforgeability against adaptive chosen messages attacks.

Proposition 3 *The scheme provides the revocation of payment.*

Proof When the user has not obtained the service, the scheme provides the function of revocation of the transaction. If *Ticket* is valid and U offers the right values of ID_U and r_0, then TPP refunds the money to U. But if the ticket is used, both U and P are required to provide relevant evidences to TPP, and TPP makes a reasonable decision to refund the money to U or P.

Since V_U is generated by the user ID_U independently, only the real user can provide the right ID_U and r_0, and pass authentication successfully. Other entities do not know $ID_U||r_0$, and they cannot pass authentication.

From the security analysis it can be seen that our scheme has unforgeability, revocable payment and anonymity even in the face of TPP. Confidentiality is obvious because symmetric encryption is used to protect the privacy of service data and IBE is used to protect the security of the ticket.

As mentioned in Sect. 1, PayPal as well as Alipay is the prevailing payment system with TPP. PayPal and Alipay both offer the function of transaction revocation, but they have serious privacy leakage issues.

Further, we compare our scheme with existing research works that are intended to ensure security and privacy of mobile payment. The comparison results of security features are shown in Table 2. Isaac et al. [11] used symmetric cryptography to achieve message confidentiality and message integrity. Because of no direct communication between the merchant and his bank, the merchant does not know the identity of the client, but the issuer knows it. So the scheme does not

Table 2 Secutity features comparisons

	Confidentiality	Unforgeability	Revocable payment	Strong anonymity
Isaac et al.'s scheme [11]	Yes	Yes	No	No
Han et al.'s scheme [10]	Yes	Yes	No	No
Bitcoin [3]	No	Yes	No	No
Isern et al.'s scheme [12]	Yes	Yes	Yes	No
Our scheme	Yes	Yes	Yes	Yes

have strong anonymity. Based on bilinear pairing, Han et al. [10] made a digital signature and encryption of messages; the scheme is unforgeability, confidentiality, and accountability; but it does not have anonymity. The two schemes [10, 11] do not consider revocable payment. As a popular payment system without TPP, Bitcoin provides anonymity. For Bitcoin, all transaction data are recorded in a public distributed blockchain, which can be obtained by any user connected to the Bitcoin network. So Bitcoin has not confidentiality. Moreover, some robust blockchain analysis methods based on graph and heuristics have been provided, and they can follow the money. And thus, Bitcoin has not strong anonymity. Meanwhile, it has the obvious weaknesses: high energy consumption and no function of payment revocation. Isern's scheme [12] used the idea of micro payment, where payment has been made using a fair exchange protocol and the forgery of e-coin is not possible because the coin creation requires the knowledge of the bank's private key. And the user can withdraw the money that is not spent. He is anonymous to a service provider, but not to the bank.

6 Performance analysis

For convenience to evaluate the computation costs of the scheme, we ignore some operations such as a hash function and a multiplication operation since they are quite light in terms of load. Since AES algorithm requires 94 μs to perform encryption with packet size of 1024 bytes, the same with decryption [22], we also ignore it. Then we focused on some time-consuming operations defined in the following notations. T_P denotes the time of executing a bilinear map operation. When executing a bilinear map operation, all exponentiations in G_2 can be transformed into scalar multiplications in G_1 to get a fast implementation. So we use T_{G1} to represent the time of executing a scalar multiplication or an exponentiation operation.

6.1 Performance of the SATP scheme

Table 3 shows computation and communication costs of our SATP scheme during four phase: ticket issuance, payment by ticket, ticket cashing and payment revocation. In the table, $|G_1|$, $|Data|$ and $|\beta|$ are the length of the element in G_1, the service data and the common data, respectively. l_1, l_2, l_3 and l_4 are the lengths of the messages $ID_T||N||r_1^{-1}r_2H_5(\alpha, R)$, $Ticket||K$, $Ticket$ and $ID_U||r_0||Ticket$, respectively.

Table 3 Computation and communication costs of the SATP scheme

	Ticket issuance	Payment	Ticket cashing	Payment revocation
Computation costs	$9\ T_{G1}+6T_P$	$4T_{G1}+4T_P$	$4T_{G1}+4T_P$	$4T_{G1}+4T_P$
Communication costs	$3\|G_1\|+l_1+\|\beta\|$	$\|G_1\|+l_2+\|Data\|$	$\|G_1\|+l_3$	$\|G_1\|+l_4$

6.2 Performance comparisons with our schemes

In Isaac et al.'s scheme [11], a merchant does not communicate with a bank directly, and a user is as a gateway. Han et al.'s solution [10] is to ensure the confidential user-to-user communication; there is no TPP in the system. Also, Bitcoin system has no TPP. So the architectures of the three secure mobile payment schemes are different from ours. We shall compare our scheme with Isern et al.'s scheme [12] since its structure is similar to ours.

To achieve the similar security level of 1024 bits RSA signature, Chen et al. [23] proposed $|q| = 20$ bytes and $|G_1| = 20$ bytes in a bilinear map; it requires 4.5 ms to perform a bilinear map operation and 0.6 ms to perform scalar multiplications in G_1 on an Intel Pentium 4 processor with the clock speed of 3.4 GHz. For elliptic curve digital signature algorithm (ECDSA), if the key is 28 bytes, then ECDSA signature is 53 bytes, the point on the elliptic curve is 29 bytes and public certificate is 84 bytes; it requires 0.8 and 4.2 ms to perform ECDSA signature and verification, respectively, on a Centrino machine with the clock speed set at 1.5 GHz [24]. In our scheme, we assume that $|G_1| = 20$ B (i.e., bytes), $|q| = 20$ B, $|ID_U| = 10$B, $|V_U| = 20$B, $|\beta| = 8 + 2 = 10$B, and $|Data| = 1024$B. In Isern's scheme, besides the same length of *Data*, we also assume that the payment credential is 20 bytes and the certificate is 84 bytes if the key is 28 bytes.

Figures 5 and 6 show computation and communication costs comparisons with during different phases, respectively. We observe that our scheme requires more computation costs. It is because we adopt signcrypt and encrypt algorithm based on IBC that uses the time-consuming bilinear map operations. In Isern et al.'s scheme, signature, verification, encryption and decryption are mainly scalar multiplication operations that are lighter than bilinear map operations.

On the other hand, Isern et al.'s scheme used public key cryptography, where public key certificates greatly increase the communication overhead. Our scheme uses IBC, where certificates are not required. It maintains better communication performance than Isern et al.'s [12] during all four phases, while providing higher security level, especially in the aspect of strong anonymity.

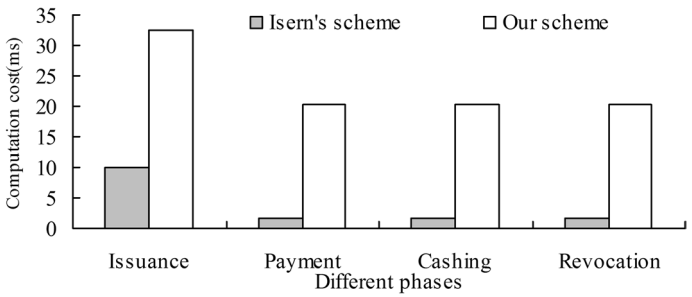

Fig. 5 The computation cost comparisons in different phases

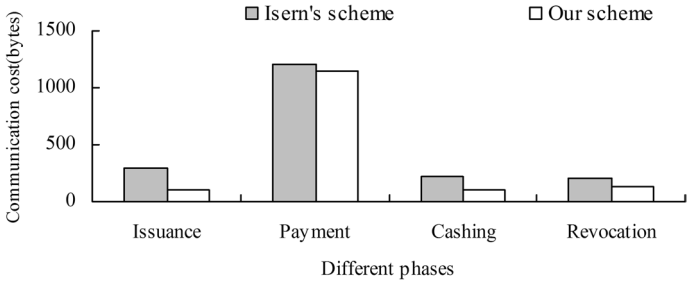

Fig. 6 The communication cost comparisons in different phases

7 Conclusion

Security, anonymity and efficiency are the most concerned issues in mobile payment. Among prevailing payment systems, PayPal has a serious privacy leakage issue, the same as Alipay. Bitcoin provides anonymity. But high energy consumption and security threats become its weaknesses. And it does not have strong anonymity.

We propose a SATP scheme, where a ticket, as a way of payment, is partially blindly signed by TPP. No matter TPP, a payee or a malicious user, no one can know the identity of a payer from transaction messages. Moreover, if the payer does not receive the requested service, he can revoke the payment. Our mobile payment scheme has confidentiality, unforgeability, strong anonymity and revocation. Since no public key certificate is required, it has clear communication advantage. Security analysis and performance analysis show that SATP has high security and it can be applied in mobile payment efficiently.

Acknowledgements This work was supported by the Major Research Project for Social Science Innovation and Development of Anhui Province (Grant No. 2017ZD005), the Visiting Scholar Projects of Anhui Province for Excellent Young and Middle-aged Backbone Talents (Grant No. gxfxZD2016305), and the Natural Science Foundation of Anhui Province (Grant No. 1608085MF141). We would like to thank the anonymous referees and Editors for their valuable comments and suggestions.

References

1. Isaac, J. T., & Zeadally, S. (2014). Design, implementation, and performance analysis of a secure payment protocol in a payment gateway centric model. *Computing, 96*(7), 587–611.
2. Preibusch, S., Peetz, T., Acar, G., & Berendt, B. (2016). Shopping for privacy: Purchase details leaked to PayPal. *Electronic Commerce Research and Applications, 15,* 52–64.
3. Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.
4. Bohannon, J. (2016). The Bitcoin Busts. *Science, 351*(6278), 1144–1146. https://doi.org/10.1126/science.351.6278.1144.
5. Conti, M., Lal, C., & Ruj, S. (2017). A survey on security and privacy issues of Bitcoin. arXiv preprint arXiv:1706.00916.
6. Miyazaki, A. D., & Fernandez, A. (2001). Consumer perceptions of privacy and security risks for online shopping. *Journal of Consumer Affairs, 35*(1), 27–44.

7. Specification, S. S. E. T. (1997). Book 3: Formal protocol definition. In *SET Secure Electronic Transaction LLC, Version* (p. 1).
8. Bellare, M., Garay, J. A., Hauser, R., Herzberg, A., Krawczyk, H., Steiner, M., et al. (2000). Design, implementation, and deployment of the iKP secure electronic payment system. *IEEE Journal on Selected Areas in Communications, 18*(4), 611–627.
9. Pukkasenung, P., & Chokngamwong, R. (2016). Review and comparison of mobile payment protocol. *Advances in parallel and distributed computing and ubiquitous services* (pp. 11–20). Singapore: Springer.
10. Han, J., Yang, Y., Huang, X., Yuen, T. H., Li, J., & Cao, J. (2016). Accountable mobile E-commerce scheme via identity-based plaintext-checkable encryption. *Information Sciences, 345,* 143–155.
11. Isaac, J. T., Zeadally, S., & Cámara, J. S. (2012). A lightweight secure mobile payment protocol for vehicular ad-hoc networks (VANETs). *Electronic Commerce Research, 12*(1), 97–123.
12. Isern-Deya, A. P., Magdalena Payeras-Capella, M., Mut-Puigserver, M., & Ferrer-Gomila, J. L. (2012). Anonymous, secure and fair micropayment system to access location-based services. In *Trustworthy ubiquitous computing* (pp. 227–247).
13. Sekhar, V. C., & Sarvabhatla, M. (2012). Secure lightweight mobile payment protocol using symmetric key techniques. In *International Conference on Computer Communication and Informatics* (pp. 1–6).
14. Gong, P., & Li, P. (2015). Further improvement of a certificateless signature scheme without pairing. *International Journal of Communication Systems, 27*(10), 2083–2091.
15. Yeh, K. H. (2017). A secure transaction scheme with certificateless cryptographic primitives for IoT-based mobile payments. *IEEE Systems Journal, 99,* 1–12.
16. Shamir, A. (1984). Identity-based cryptosystems and signature schemes. In *Crypto* (vol. 84, pp. 47–53).
17. Zhang, F., Safavi-Naini, R., & Susilo, W. (2003). Efficient verifiably encrypted signature and partially blind signature from bilinear pairings. In *Indocrypt* (vol. 2904, pp. 191–204).
18. Chow, S., Hui, L., Yiu, S., & Chow, K. (2005). Two improved partially blind signature schemes from bilinear pairings. *Information security and privacy* (pp. 355–411). Berlin: Springer.
19. Li, F., Zhang, M., & Takagi, T. (2013). Identity-based partially blind signature in the standard model for electronic cash. *Mathematical and Computer Modelling, 58*(1), 196–203.
20. Boneh, D., & Franklin, M. (2001). Identity-based encryption from the Weil pairing. In *Annual international cryptology conference* (pp. 213–229). Berlin: Springer.
21. Pointcheval, D., & Stern, J. (1996). Security proofs for signature schemes. In *Eurocrypt* (vol. 96, pp. 387–398).
22. Wang, N. W., Huang, Y. M., & Chen, W. M. (2008). A novel secure communication scheme in vehicular ad hoc networks. *Computer Communications, 31*(12), 2827–2837.
23. Chen, L., Ng, S. L., & Wang, G. (2011). Threshold anonymous announcement in VANETs. *Selected Areas in Communications, 29*(3), 605–615.
24. Calandriello, G., Papadimitratos, P., Hubaux, J. P., & Lioy, A. (2007). Efficient and robust pseudonymous authentication in VANET. In *International workshop on vehicular ad hoc networks, Vanet 2007, Montréal, Québec, Canada* (pp. 19–28). OAI.