Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 6953160, 21 pages
https://doi.org/10.1155/2018/6953160

Research Article

A Secure Fair Exchange for SMS-Based Mobile Payment Protocols Based on Symmetric Encryption Algorithms with Formal Verification

Chalee Thammarat and Werasak Kurutach

Faculty of Information Science and Technology, Mahanakorn University of Technology, 140 Moo 1, Cheumsampan Road, Nong Chok, Bangkok 10530, Thailand

Correspondence should be addressed to Chalee Thammarat; chalee23@gmail.com

Received 30 August 2017; Revised 18 January 2018; Accepted 17 April 2018; Published 5 July 2018

Academic Editor: Javier Aguiar

Information security and fair exchange are essential to creating trust among all the parties participating in any sale transaction. However, implementing them in any mobile commerce is challenging due to the limitation of resources on mobile devices. Numerous m-commerce protocols that have been proposed so far still lack those two important aspects. In this paper, we propose mobile payment (m-payment) protocols, a crucial part of m-commerce, that incorporate both information security and fair exchange while retaining their own lightweight property. To allow convenience of use, the proposed protocols can be implemented on the existing Short Message Service (SMS) infrastructure. Our approach is based on the secure session key generation technique to enhance information security under lightweight conditions and involves a trusted third party to guarantee fair exchange without information disclosure. We have formally proven that our protocols are more effective and efficient than others in terms of fairness, security, and lightweight properties. In addition, the soundness and completeness of the protocols have been analyzed and proven using BAN logic and an automated security protocol proof tool named Scyther.

1. Introduction

Currently, mobile commerce (m-commerce) is one of the most popular methods for buying goods or services on the Internet. It has had a major impact on the growth of the world economy as well as improving the quality of life in the human society. One major factor that drives the success of m-commerce is mobile payment (m-payment), which allows the exchange of goods or services and money between a purchaser and a vendor. Therefore, m-payment is a crucial mechanism and needs to be trustworthy from the perspectives of all parties involved. There are two essential issues that can create trust in a sale transaction based on the m-payment, i.e., fair exchange and information security. These will provide all involved parties with the confidence that no one can take advantage of the others when conducting any sale transaction. In addition, they can prevent fraud. Nowadays, SMS-based transactional payments are a major model of mobile payments. This kind of payment is simple for purchasers to use. They can pay for goods or services via a plain text message sent from a mobile phone. However, the message is transmitted through the wireless network and, hence, can be eavesdropped by a malicious person. Even though the Global System for Mobile Communication (GSM) uses cryptographic techniques of A5/1 and A5/2 between the mobile device and the base station subsystem (BSS), the messages are still vulnerable [1–4]. During the past few years, many researchers have investigated protocols for mobile payments with fair exchange [1, 3, 5–9]. Unfortunately, they still lack some of the important properties of information security, e.g., mutual authentication, nonrepudiation, strong fairness (due to no involvement of a trusted third party), and undisclosed information to the trusted third party (if any).

In this paper, we review a number of existing techniques for ensuring fairness. More importantly, we propose protocols for mobile payments that satisfy the crucial properties of information security. The secure session key generation technique is employed to accomplish a security requirement.

Consequently, this results in the lightweight preservation of our protocols. Furthermore, our approach can be implemented on the current SMS infrastructure to retain its ease of use. In this method, the online TTP (trusted third party or TTP) model is chosen because the TTP is required to keep some evidences to be used when a dispute arises. Moreover, the online TTP must not be able to access or disclose information from those evidences in any case or for their own benefits. In contrast, in an offline TTP model, the information needs to be disclosed to the TTP in order to make a dispute resolution possible.

2. Related Works

Generally, there are two types of fairness protocols. One requires the involvement of a trusted third party (TTP), but the other does not [10–12]. The former can be further divided into three categories: inline [10, 13, 14], online [7, 10, 12, 15–17], and offline [10, 11, 18–24]. In our work, we will emphasize the fairness protocols that involve the online TTP. In this section, we will briefly describe previous works related to this topic.

A number of mobile payment protocols have been proposed [1, 3, 5–9, 25] to secure mobile payment based on SMS text messaging. Saxena and Chaudhari [1] suggested EasySMS, which offers end-to-end secure communication through SMS between mobile users and is able to prevent several attacks, such as SMS disclosure, air modification, replay attacks, man-in-the-middle attacks, and impersonation attacks. This protocol deploys symmetric cryptography as well as hash functions. Pourali et al. [5] proposed a secure SMS model of e-commerce payment, based on Elliptic Curve Cryptography (ECC). The authors argue that their model satisfies the properties of confidentiality, integrity, authentication, and nonrepudiation. The paper provides various types of e-payment (electronic payment) business models, which are e-payment methods of mobile payment schemes. The model is suitable for SMS-based mobile payment. Kisore and Sagi [6] proposed a secure SMS protocol for a digital cash system, which is a protocol based on ECC with public key algorithms in the Cryptographic Message Syntax (CMS), key agreement protocol, and Advanced Encryption Standard (AES). The proposed digital cash system satisfied confidentiality and authentication. However, the protocols [5, 6] lack bilateral authentication and some important properties of security, for example, message authentication and recipient authentication. This is because each message is encrypted with its own public key, which cannot verify the sender of the message. Bojjagani and Sastry [7] proposed SSMBP, a unique protocol for SMS which is based on mobile banking, the payment framework, and Elliptic Curve Digital Signature Algorithm (ECDSA), used for generating and verifying digital signatures and also for encryption and decryption of SMS. The Elliptic Curve Integrated Encryption Scheme (ECIES) is used to satisfy confidentiality, integrity, authentication and nonrepudiation. This protocol provides secure SMS communication between customers and banks through a mobile phone banking application and deploys both symmetric and asymmetric cryptography, including hash functions. There are five parties involved in this protocol: a payer, a payee, an issuer bank, an acquirer bank, and a payment gateway. However, the session key shared between the issuer bank and the payer (SKB), the session key shared between the issuer bank and the payment gateway (SKPG), and the session key shared between the acquirer bank and the payment gateway (SKAB) need to be transmitted over the network using static parameters that could possibly be intercepted by attackers, and the protocol is prone to brute force attacks. Rongyu et al. [8] proposed PK-SIM, a security framework, offering solutions for the development of secure mobile business applications using SMS to provide point-to-point security between the PK-SIM card and the Secure Access Gateway (SAG). There are parties involved in this protocol: a PK-SIM card for storing security credentials, a SAG for receiving and sending secure SMS messages, a trusted third party, the Certification Authority (CA) for providing a public key certification service and a mobile operator for providing the communication infrastructure for the SMS. Both symmetric and asymmetric cryptography, including hash functions, are deployed in this protocol. There are three subphases: the authentication phase, the session key establishment phase, and the communication phase. Rongyu et al. argue that their framework satisfies the requirements of authentication, integrity, and confidentiality. However, the session key UAKey (the primary key between the PK-SIM card and the SAG) needs to be transmitted over the network and, therefore, could possibly be intercepted by attackers. In other words, it is prone to brute force attacks. Saxena and Chaudhari [3] proposed SecureSMS, a new secure and optimal choice for a secure SMS messaging protocol. This protocol provides end-to-end SMS security and is able to prevent various threats and attacks such as replay attacks, man-in-the-middle attacks, SMS spoofing, and SMS disclosure. The system is based on symmetric cryptography, including the hash function. The framework satisfies authentication, confidentiality, integrity, and nonrepudiation of the messages. There are three parties involved in this protocol: The Mobile Station (MS) sends the SMS, and this is carried out by the Authentication Server (AS), with the help of the Certification Authority/Registration Authority (CA/RA). The authors argue that their protocol satisfies message mutual authentication between the MS and AS. However, within the messages, ID_MS, T1, T3, and ExpT are sent in clear text and, therefore, can be easily intercepted and modified by an attacker. Minta and Panchami [9] proposed an efficient encryption protocol for securely transmitting a confidential SMS from one mobile phone to another, which serves the cryptographic goals of confidentiality, authentication, and integrity of the messages. This protocol prevents various attacks such as man-in-the-middle attack, replay attack, SMS disclosure, and over-the-air modification. The protocol is composed of four parties: Mobile Station 1 (MS1), Mobile Station 2 (MS2), an Authentication Server (AS), and a Certification Authority (CA). The framework proposed by Bojjagani and Sastry [25] provides end-to-end SMS communication between the customer and the bank through a mobile application. The main objective of the framework is to design and develop a security framework for SMS banking. This framework is validated using formal methods utilizing a model-checking tool called Scyther.

Unfortunately, most of the mobile payment systems that have thus far been proposed still lack fairness. For example, some systems allow disclosure of information to the trusted third party, and some do not even involve a trusted third party. In addition, many researchers have proposed mobile payment protocols with weaknesses in some areas such as confidentiality, integrity, nonrepudiation, and mutual authentication. Mutual authentication can prevent both replay attacks and man-in-the-middle attacks, which is critical for information security. This paper introduces mobile payment protocols that enhance fairness and provide security properties such as confidentiality, integrity, nonrepudiation, and mutual authentication. In addition, the proposed protocols are lightweight and have improved security since they deploy the secure session key generation technique and the session keys will not be reused.

3. The Proposed Mobile Payment Protocols

In this section, we apply the fairness model in [28] to mobile payment protocols to ensure strong fairness. Later in this paper we will show that the proposed protocols satisfy strong fairness. Our protocols are composed of two subprotocols: Purchase Credit Request and Making Payment.

3.1. Notations and Assumptions. The protocols comprise four parties: a client or C, a merchant or M, a mobile operator or O (payment gateway), and a trusted third party or TTP. Clients need to install the proposed software on their mobile devices. The client who establishes an account with a mobile operator pays monthly fees.

(i) ID_A is the identity of A.

(ii) $\{DK_{AB}, K_{AB}, m_{AB}\}$ are the key distribution parameters that are shared between the parties. Readers may find more details about the key distribution parameters from [29]. Our protocols used parameters to generate the secure session key generation technique proposed in [29].

(iii) K_{AB} stands for a long-term key.

(iv) DK_{AB} stands for a distributed key.

(v) m_{AB} is a random number. It is utilized to indicate the number of keys that will be produced.

(vi) SK_{ABj} where j = 1, . . ., m are the session keys shared between party A and party B.

(vii) $h(M, SK_{ABj})$ is a Message Authentication Code (MAC) of a message M and a key SK_{ABj}.

(viii) $h(M)$ is a hash value of a message M.

(ix) $\{M\}_{SKABj}$ is a symmetrically encrypted message of a message M with a key SK_{ABj}.

(x) CL_T is a credit limit up to which the client is allowed to purchase goods or services.

(xi) SN is a serial number for a top-up cash card purchased offline, which infers the credit limit in the client's account.

(xii) CL_{RM} is the remaining credit in the client's account.

(xiii) $OI = \{TID, Price, OD\}$. TID is Transaction ID. $Price$ is the price of goods or services, and OD is an order description containing details of the goods or services purchased.

The mobile operator establishes a Transport Layer Security (TLS) session and shares this $\{K_{OTTP}, DK_{OTTP}, m_{OTTP}\}$ with TTP. Both the mobile operator and TTP create a set of session keys SK_{OTTPj} by using the secure session key generation technique proposed in [29], where j = 1, . . ., m, while implementing the system. The proposed protocols assume that the session keys between parties are the same.

3.2. The Fairness Model for Internet Transactions. Thammarat et al. [28] proposed a fairness model for Internet transactions. The model supports a mobile payment protocol. This model still lacks some of the conditions to fully ensure fairness. Therefore, we extend the proposed model in [28] to complete the fairness protocols. The symbolic notations and modal operators can be defined as follows.

Let S denote a payment system defined as $S = \{C, M, PG\}$, where C is a client who buys goods or services, M is a merchant, and PG is a payment gateway which acts as the financial institution for both C and M. In the payment system, C acts as the payer and M acts as the payee. In addition, let V be a verifier, an external party not involved in a transaction but trusted by all parties, and let TTP denote a trusted third party. TTP is not involved in the transaction itself but keeps all transaction data for later verification. R denotes any party.

(i) *P authorized X*: the party P has authorization to perform an action X.

(ii) *P CanProve X to R*: the party P is able to prove to the party R that the statement X is true without revealing any information which is considered to be secret to R.

(iii) *Payment-order(C, M)* is the interaction between the client and the merchant when the client requests to purchase goods or services from the merchant.

(iv) *Debit(C, PG)* is the interaction between the payment gateway and the client regarding the deduction of the payment token, requested by the client, from the client's account.

(v) *Credit(M, PG)* is the interaction between the payment gateway and the merchant in order to transfer the payment token to the merchant account.

(vi) *Log(C, TTP)* is the interaction between TTP and the client, enabling TTP to keep all the transactions occurring between the client and related parties and send them to the client.

We define a new operator called "*satisfies*", which refers to the situation where a party satisfies the result given at the end of the transaction. For example, *"C satisfies payment-order(C, M)"* means that the client C has satisfied the results of the transaction occurring between the client and the merchant.

Table 1: Advantages and disadvantages of the types of fairness.

Type of Fairness	Advantage	Disadvantage
No Fairness	(i) no cost of *TTP*	(i) very high number of messages (ii) need for reliable network (iii) unable to have dispute resolution
Weak Fairness	(i) low number of messages	(i) need for reliable network (ii) full *TTP*
Strong Fairness	(i) semi-TTP (ii) no need for reliable network	(i) high number of messages (ii) cost of *TTP*

The conditions of fairness are as follows: Strong fairness: (1) *TTP* is involved. (2) There is no information disclosure to *TTP*. Weak fairness: (1) *TTP* is involved. (2) There is information disclosure to *TTP*. No fairness: (1) *TTP* is not involved. The details of each condition of fairness can be explained as follows:

(i) *TTP* is involved: *TTP* keeps all the transactions and provides the transaction occurring between the parties as evidence when a dispute arises.

(ii) *TTP* is not involved: there is no *TTP* to keep all transactions and no evidence can be sent to the parties. When a dispute arises, parties cannot have dispute resolution. It is not possible to ensure fairness that is not strongly an involvement of *TTP* [21, 30].

(iii) Information disclosure to *TTP*: *TTP* can read all transactions occurring between the two parties. *TTP* may sell the information about parties to the competitors.

(iv) No information disclosure to *TTP*: there is no *TTP* involved. *TTP* cannot read all transactions occurring between parties and therefore cannot sell the information about parties to the competitors.

Table 1 shows the advantages and disadvantages of the types of fairness. Strong fairness has more advantages than the other types, for example, semi-TTP (semitrusted third party), and it does not need a reliable network. In this paper, we will apply Thammarat's fairness model for Internet transactions to existing SMS mobile payment protocols for strong fairness. Note that a semi-TTP is one that can misbehave on its own but will not collude with any of the participating parties.

Client's Fairness. The following requirements must be satisfied from the client's point of view to achieve the goal of fairness:

M CanProve (C authorized payment-order(C, M)) to V $\bigwedge$
PG CanProve (C authorized debit(C, PG)) to V $\bigwedge$
C CanProve (PG authorized debit(PG, C)) to V $\bigwedge$
C CanProve (M authorized payment-order(M, C)) to V $\bigwedge$
TTP CanProve (C authorized Log(C, TTP)) to V $\bigwedge$
C CanProve (TTP authorized Log(TTP, C)) to V
$\longrightarrow$ *C satisfies payment-order(C, M, TTP)*

In each client's opinion, the client will satisfy the transactions only if he/she gets both a *payment-order* response from the merchant and a *debit* response from the payment gateway. However, the responses must be from the previous transaction made by the client. All requests are sent to the trusted third party.

Merchant's Fairness. For a transaction to be satisfactory in the merchant's opinion, the following requirements must be satisfied:

M CanProve (C authorized payment-order(C, M)) to V $\bigwedge$
PG CanProve (M authorized credit(M, PG)) to V $\bigwedge$
M CanProve (PG authorized credit(PG, M)) to V $\bigwedge$
C CanProve (M authorized payment-order(M, C)) to V $\bigwedge$
TTP CanProve (M authorized Log(M, TTP)) to V $\bigwedge$
M CanProve (TTP authorized Log(TTP, M)) to V
$\longrightarrow$ *M satisfies credit (M, PG, TTP)*

It can be seen from the above statements that the merchant will satisfy a transaction if he/she has delivered or committed to deliver goods or services to clients as a result of receiving a *payment-order* request from the client. The payment has to be processed by the payment gateway. The payment gateway must transfer the amount requested by the merchant to the account of the merchant, prior to the delivery of goods or services to the client.

Payment Gateway's Fairness. For a transaction to be satisfactory in the payment gateway's opinion, the following requirements must be satisfied:

PG CanProve (C authorized debit (C, PG)) to V $\bigwedge$
PG CanProve (M authorized credit (M, PG)) to V $\bigwedge$
R CanProve (PG authorized payment-clearing (PG, C, M)) to V $\bigwedge$
TTP CanProve (PG authorized Log(PG, TTP)) to V $\bigwedge$
PG CanProve (TTP authorized Log(TTP, PG)) to V
$\longrightarrow$ *PG satisfies payment-clearing (PG, C, M, TTP)*

It can be seen that the payment gateway will satisfy a transaction if the payment gateway has performed *payment-clearing* as a result of the requests from the client and merchant. Specifically, the payment gateway must perform actions according to the *debit* and *credit* requests made by both the client and the merchant. The payment gateway transfers or commits to deduct the amount requested by the client and transfer the amount requested by the merchant to the account of the merchant.

3.3. Background Concept on the Secure Session Key Generation Technique. In this section, we will explain the concept of the secure session key generation technique that is used in a phase of our proposed protocols. The topics that are extensively discussed in symmetric encryptions are the secure session key generation techniques. Many researchers have presented a secure session key generation technique used in their own

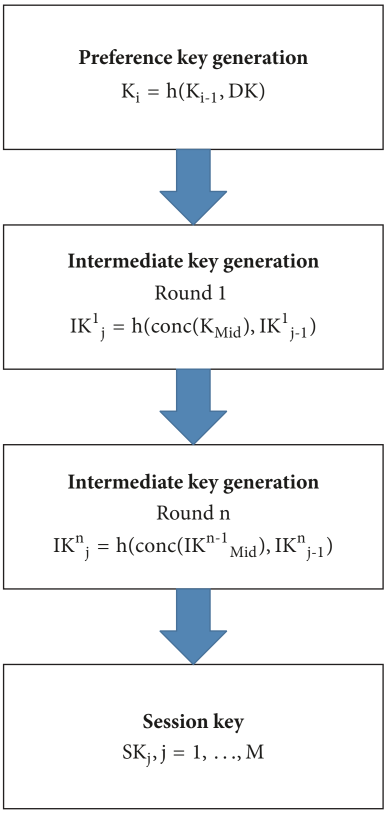

Preference key generation
$K_i = h(K_{i-1}, DK)$
Intermediate key generation
Round 1
$IK^1{}_j = h(conc(K_{Mid}), IK^1{}_{j-1})$
Intermediate key generation
Round n
$IK^n{}_j = h(conc(IK^{n-1}{}_{Mid}), IK^n{}_{j-1})$
Session key
$SK_j, j = 1, \ldots, M$

Figure 1: Session key generator.

papers [29, 31–35]. Among all the mentioned approaches, [29] proposed a secure session key generation technique that prevents key compromise attacks. Moreover, with the secure session key generation technique the parties do not need to transmit a key over the network. In our paper, we use [29] in introducing our protocols. The methodology of [29] is demonstrated below.

Let us assume that party A and party B share $\{K_{AB}, DK, m\}$, where K_{AB} stands for a long-term key, DK stands for a distributed key, and m is a random number. m is utilized to indicate the number of keys that will be produced. The *conc(M1, M2, M3)* shows the concatenation of each of the messages *M1*, *M2,* and *M3*, respectively. The secure session key generation technique process is shown in Figure 1.

After sharing $\{K_{AB}, DK, m\}$, party A and party B create a set of preference keys K_i, where i = 1, . . ., m, as follows: $K_i = h(K_{i-1}, DK)$, where $K_0 = K_{AB}$. The set of K_i will be applied as an origin to invigorate session keys. K and DK can be removed from the system.

Next, party A and party B generate sets of intermediate keys to increase the difficulty for cryptanalysis. Moreover, this makes it more difficult to find the preference key if the session key is compromised. In each round, a new set of intermediate keys is produced. Note that the higher the number of rounds performed, the greater the security of the system. The intermediate key generation is performed as follows: $IK^x{}_j = h(conc({}^{IKx\text{-}1}{}_{Mid}), IK^x{}_{j\text{-}1})$, where x specifies the number of rounds and j specifies the number of intermediate keys that are generated, j = 1, . . ., m. $IK^{x\text{-}1}{}_{Mid}$ stands for the set of $\{IK^{x\text{-}1}{}_{Mid1}, IK^{x\text{-}1}{}_{Mid2}, IK^{x\text{-}1}{}_{Mid3}\}$. $IK^x{}_{Mid1} = mid(IK^x{}_1, IK^x{}_{rm})$ and rm is the remaining number of intermediate keys in the set of $IK^x{}_j$. $IK^x{}_{Mid2} = mid(IK^x{}_{Mid1}, IK^x{}_{rm})$. $IK^x{}_{Mid3} = mid(IK^x{}_1, IK^x{}_{Mid2})$. $IK^1{}_{Mid1} = K_{Mid1}$, $IK^1{}_{Mid1} = K_{Mid2}$, and $IK^1{}_{Mid1} = K_{Mid3}$. The generation of K_{Mid1}, K_{Mid2}, and K_{Mid3} is the same as that of $IK^x{}_{Mid1}$, $IK^x{}_{Mid2}$, and $IK^x{}_{Mid3}$, respectively. $IK^x{}_{j\text{-}1} = \varphi$. The output of the last round of intermediate key generation is considered as session keys SK_j, where j = 1, . . ., m, which is shown below: $IK^n{}_1 = SK_1, IK^n{}_2 = SK_2, \ldots, IK^n{}_m = SK_m$. Party A and party B can then use SK_j as a credential to secure transactions as, say, an encryption key or as an input to Message Authentication Codes.

It can be obviously seen that the session key is created purely offline. Each party can produce a set of session keys used to secure communication among themselves with no requirements to exchange credentials through the network, as a new session key is not transferred over the network.

3.4. Registration Phase. It is assumed that mobile devices have the software based on installation of our protocols. Once the software is loaded, the client needs to log on to the payment system via a secure channel, e.g., TLS (Transport Layer Security). The protocol details are as follows:

(1) The client establishes a TLS session and shares $\{K_{CO}, DK_{CO}, m_{CO}\}$ with the mobile operator. The client and the mobile operator create a set of session keys SK_{COj} by using the secure session key generation technique proposed in [29], where j = 1, . . ., m.

(2) The client establishes a TLS session and shares $\{K_{CTTP}, DK_{CTTP}, m_{CTTP}\}$ with *TTP*. The client and *TTP* create a set of session keys SK_{CTTPj}, using the secure session key generation technique proposed in [29], where j = 1, . . ., m.

3.5. Purchase Credit Request Phase. In this section, before the client pays for goods or services, the client purchases a top-up cash card. The client runs the client software from his/her device. Next, he or she fills in the serial number of the top-up cash card and sends the following to the mobile operator:

M1: C ⟶ O: $ID_C, SN, T_1, h(SN, CL_T, T_1, SK_{COj}), \{h(ID_C, SN, CL_T, T_1)\}_{SKCTTPj}$

M2: O ⟶ TTP: $\{h(ID_C, SN, CL_T, T_1)\}_{SKCTTPj}, \{h(CL_T, T_1, T_2)\}_{SKOTTPj}$

M3: TTP ⟶ O: $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}, \{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKOTTPj+1}$

M4: O ⟶ C: $T_2, h(CL_T, T_1, T_2, SK_{COj+1}), \{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}$

In message M1, the client sends message ID_C, SN, T_1, $h(SN, CL_T, T_1, SK_{COj})$, $\{h(ID_C, SN, CL_T, T_1)\}_{SKCTTPj}$ to the mobile operator. The message $h(SN, CL_T, T_1, SK_{COj})$ is considered as Message Authentication Code (MAC), which is an authentication token between the client and the mobile operator. Moreover, in order to guarantee information correctness and to defend against erroneous information alteration or destruction, the message $\{h(ID_C, SN, CL_T, T_1)\}_{SKCTTPj}$ will be transmitted to *TTP* via the mobile operator. Note that T_1

Table 2: Symbols and abbreviations.

Symbol	Definition	Bits
ID_{MS1}/ID_{MS2}/IMSI	International Mobile Subscriber Identity of Mobile station	80
MAC/H	Message Authentication Code/Hash function	160
ReqNo/Seq	Request number	8
DK	Delegation key	128
Order_msg	Product order	160
Price/P	Cost of product	64
Withdraw_Req_Msg/ Withdraw_Resp_Msg	The customer pays the desired price from his bank	64
Ack	A validation message to the customer	64
FS	Financial server identification	128
Name	Name of payer	160
NationalID	National Identification	104
RandomPublicKey	The ephemeral public key	128
PayeeMobNo	Mobile phone number of payee	80
Value	Digital currency	80
$ID_{PR}/ID_{PE}/ID_C/ID_M$	Identification of Payer/ Payee/Client/Merchant	80
Payee_Id	Identification of Payee	80
BI	Bank information	128
Amt	Amount in payer's bank account	64
$SK_B/SK_{PG}/SK_{IB}/$ $SK_{PGIB}/SK_{PGAB}/SK_{CO}/SK_{CTTP}/SK_{OTTP}/$ $SK_{MTTP}/SK_{CM}/SK_{MO}/SK_{AB}$/K/Passkey/UAKey	Session Key	128
$TS_{PG}/TS_{IB}/TS_B/TS_{PR}/$ $T/T_1/T_2/T_3/T_4/T_5$	Timestamp	80
$N_{PG}/N_{IB}/N_{PR}/N_B$	Nonce	128
Payment_Notify	Payment acknowledgment	128
PI_{PR}	Payment Information of payer	128
TID	Transaction ID	128
ExpT/Expiry/KeyExpiry	Expiry of primary key	64
N_S/N_C	Random number	128
$Cert_{SAG}$	Certificate of Security Access Gateway	40
ID_ME	Mobile phone Number	80
C_{ME}	Certificate of mobile phone	40
SN	Serial number of the top-up cash card	112
CL_T	Credit limit	40
CL_{RM}	Remaining credits	40
OI/Order_Message	Order information	160
(Yes/No)	Purchase credit request/Message status of merchant verifies the goods or services	24
(Accept/Reject)/Req_Msg	Payment status	48

denotes the timestamp for the time when the purchase credit is requested and to prevent a replay attack.

In message M2, upon receiving this message from the client, the mobile operator sends the message $\{h(ID_C, SN, CL_T, T_1)\}_{SKCTTPj}$, $\{h(CL_T, T_1, T_2)\}_{SKOTTPj}$ to *TTP*.

In message M3, *TTP* decrypts this message with SK_{CTTPj} and SK_{OTTPj} and keeps the hash value. *TTP* encrypts the hash value of $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}$ with $SK_{CTTPj+1}$and encrypts the hash value of $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKOTTPj+1}$ with $SK_{OTTPj+1}$. Then, *TTP* sends all the messages to the mobile operator. Note that T_2 denotes the timestamp when the purchase credit is sent to a client and to prevent a replay attack.

In message M4, the mobile operator generates message $h(CL_T, T_1, T_2, SK_{COj+1})$, which is an authentication token between the client and the mobile operator. Moreover, this

Table 3: Cryptographic operation functions.

Functions	Definition
$\{\}_{DK}$	Encryption with delegation Key
$\{\}_{SK}$	Encryption with symmetric key encryption
$\{\}_{Pub}$	Encryption with asymmetric key encryption
f1	Cryptographic function to generate DK
f2	Message Authentication Code function
f3	Hash function

Table 4: Message length of our protocols.

Message	Purchase Credit (Bytes)	Making Payment (Bytes)
M1	96	126
M2	64	80
M3	96	64
M4	88	96
M5	-	192
M6	-	64
M7	-	112
Total (Bytes)	344	734

message is created in order to guarantee information correctness and to defend against erroneous information alteration or destruction. Then, the mobile operator sends message $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}$ to the client.

In the above messages, the client supplies the serial number of the top-up cash card together with the requested credit limit to the mobile operator. Mobile operators can infer the credit limit from the serial number, given that the serial number is from a prepaid cash card. Note that serial numbers can also track the credit limit up to which the client is allowed to buy goods or services from the mobile operator. Note that an attacker cannot modify the serial number when the serial number is transmitted in clear text as it is key-hashed in $h(SN, CL_T, T_1, SK_{COj})$, which is a MAC with SK_{COj} that is shared between the client and the mobile operator. After receiving the demand from the client, the mobile operator adds the credit limit value to the client's account and sends a message to the client. The mobile operator acknowledges the increment of the client's credit limit. During this phase, the client utilizes a top-up cash card in case they do not have enough credit to make the payment phase.

3.6. Making Payment Phase. At the beginning of the protocol, the client and the merchant need to establish accounts with their mobile operator. The mobile operator will deduct a certain amount of value that is purchased by the client when he/she opens the account. There are two major functions on the client's side: application to operate searching and Making Payment for goods or services. The merchant is the seller of the goods or services on a mobile portal operated by the mobile operator. The protocol details are as follows:

(1) The client establishes a TLS session and shares $\{K_{CM}, DK_{CM}, m_{CM}\}$ with the merchant. The client and the merchant create a set of session keys SK_{CMj} by using the secure session key generation technique proposed in [29], where j = 1, . . ., m.

(2) The merchant establishes a TLS session and shares $\{K_{MO}, DK_{MO}, m_{MO}\}$ with the mobile operator. The merchant and the mobile operator create a set of session keys SK_{MOj}, using the secure session key generation technique proposed in [29], where j = 1, . . ., m.

(3) The merchant establishes a TLS session and shares $\{K_{MTTP}, DK_{MTTP}, m_{MTTP}\}$ with *TTP*. They both establish a set of session keys SK_{MTTPj} using the secure session key generation technique proposed in [29], where j = 1, . . ., m.

(4) The client browses lists of goods or services via the application on his or her device. When the goods or services have been added to the cart, the client can make a payment by the protocol described below:

M1: C $\longrightarrow$ O: $ID_C, T, \{ID_M, OI, T, h(OI, SK_{CMj})\}_{SKCOj}, \{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}$

M2: O $\longrightarrow$ M: $\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj}$

M3: M $\longrightarrow$ O: $\{Yes/No, h(Yes, OI, SK_{CMj+1})\}_{SKMOj+1}, \{h(Yes, OI)\}_{SKMTTPj}$

M4: O $\longrightarrow$ TTP: $\{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}, \{h(Accept/Reject, OI, CL_{RM})\}_{SKOTTPj}, \{h(Yes, OI)\}_{SKMTTPj}$

Otherwise, the mobile operator terminates the client's request.

M5: TTP $\longrightarrow$ O: $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKCTTPj+1}, \{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKOTTPj+1}, \{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKMTTPj+1}$

M6: O $\longrightarrow$ M: $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKMTTPj+1}$

M7: O $\longrightarrow$ C: $\{Accept/Reject, Yes, CL_{RM}, h(Yes, OI, SK_{CMj+1}), h(OI, SK_{MOj+1})\}_{SKCOj+1}, \{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, IO)\}_{SKCTTPj+1}$

In the case of message M1, the client sends message $ID_C, T, \{ID_M, OI, T, h(OI, SK_{CMj})\}_{SKCOj}, \{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}$ to the mobile operator. The message $\{ID_M, OI, T, h(OI, SK_{CMj})\}_{SKCOj}$ is encrypted with SK_{COj}, which is the session key shared between the client and the mobile operator for the mobile operation to verify authenticity of the client. It can be noted that the mobile operator does not know the session key SK_{CMj} in order to construct the message $h(OI, SK_{CMj})$ since it cannot generate this message by itself. This message $h(OI, SK_{CMj})$ is considered as a Message Authentication Code (MAC), which is used to guarantee information correctness and to defend against erroneous information alteration or destruction. Moreover, it enables the merchant to verify authenticity of the client. The message $\{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}$ will be transmitted to *TTP* via the mobile operator. It should be noted that *T* denotes the timestamp when the payment is requested and to prevent a replay attack.

In the case of message M2, upon receiving the message from the client, the mobile operator sends message $\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj}$ to the merchant to verify the authenticity of the mobile operator. The hash value of $h(OI, SK_{CMj})$ indicates that the mobile operator does not know the session key SK_{CMj} that is used to construct the message $h(OI, SK_{CMj})$ since it cannot generate this message by itself.

Table 5: SMS overhead of our protocols.

Session	SMS Overhead		
	Bytes	% 7-Bit ASCII (160 Byte)	% 8-Bit ASCII (140 Byte)
Purchase Credit			
M1	96	60	68.75
M4	88	55	62.85
Making Payment			
M1	126	78.75	90
M7	112	70	80

Table 6: Comparison of cryptographic operations.

	[1]	[5]	[6]	[7]	[8]	[3]	[9]	[25]	Purchase Credit	Making Payment
Symmetric Encryption	6	-	-	3	2	5	9	3	4	10
Asymmetric Encryption	-	4	3	4	4	-	-	5	-	-
Message Authentication Code	3	-	-	3	-	-	-	2	2	4
Hash Function	-	-	-	-	4	2	1	-	2	3
Number of Messages	9	6	5	6	6	7	9	6	4	7
Number of Parties	4	4	2	5	3	3	4	5	3	4

Table 7: Comparison of energy consumption.

	AES (1.21 μJ/byte)	RSA (546.5 μJ/byte)	HMAC (1.16 μJ/byte)	SHA1 (0.76 μJ/byte)	Total (μJ/byte)
[1]	7.26	0	3.48	0	10.74
[5]	0	2,186	0	0	2,186
[6]	0	1,639.5	0	0	1,639.5
[7]	3.63	2,186	3.48	0	2,193.11
[8]	2.42	2,186	0	3.04	2,191.46
[3]	6.05	0	0	1.52	7.57
[9]	10.89	0	0	0.76	11.65
[25]	3.63	2,732.5	2.32	0	2,738.5
Purchase Credit	4.84	0	2.32	1.52	8.68
Making Payment	12.1	0	4.64	2.28	19.02

Table 8: Communication complexity.

Protocol	Communication cost (bits)
[1]	(1)+(2)+(3)+(4)+(5)+(6)+(7)+(8)+(9)=(80+80+160+24)+(80+80+80+80+160+160)+(80+80+80)+(80)+(80+160+64)+ (80+24)+(80+24+64+128)+(80)+(24+80)=2192∗n
[5]	(1)+(2)+(3)+(4)+(5)+(6)= (160)+(64+160+80)+(64+48)+(64)+(64)+(64)=768∗n
[6]	(1)+(2)+(3)+(4)+(5)=(128)+(160+80+104+160+64+128)+(128)+(128)+(80+80+128+128)=1496∗n
[7]	(1)+(2)+(3)+(4)+(5)+(6)=80+128+80+128+80+128+80+128)+(128+128+128+80+128+128+128+80)+(80+64+128+128+80+160) +(80+64+128+160)+(128+64+80+128+160)+(128+80+128)=3728∗n
[8]	(1)+(2)+(3)+(4)+(5)= (40+80+128+24)+(80)+(40+160)+(128+128+64+160)+(128)=1160∗n
[3]	(1)+(2)+(3)+(4)+(5)= (80+80+160)+(80+80)+(80)+(80+160+64)+(80)=944∗n
[9]	(1)+(2)+(3)+(4)+(5)= (80+80+160)+(80+80)+(80)+(80+160+64)+(80)=944∗n
[25]	(1)+(2)+(3)+(4)+(5)+(6)=(80+128+128+80+80+128+128+80)+(128+128+128+80+80+80+128+128+128+80+80+80) +(128+80+128+128+80+128)+(160+128+128+128+128+80)+(160+128+128+128+128+80) +(128+80+128)=4592∗n
Making Payment	(1)+(2)+(3)+(4)+(5)+(6)+(7)=80+80+80+160+80+160+160)+(160+160+160+80)+(24+160+160)+(160+160+160) +(160+160+160+160+160+160+160+160+160)+(160+160+160) +(48+24+40+160+160+160+160+160)=5016∗n

Table 9: Computational complexity.

Protocol	Computational cost
[1]	f2, f2, $\{\}_{SK}$, $\{\}_{SK}$, f2, f1, $\{\}_{DK}$, f1, $\{\}_{SK}$, $\{\}_{SK}$, $\{\}_{DK}$=11∗n
[5]	$\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{Pub}$=4∗n
[6]	$\{\}_{Pub}$, f3, $\{\}_{Pub}$, $\{\}_{SK}$=4∗n
[7]	$\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{Pub}$, f2, $\{\}_{SK}$, f2, $\{\}_{SK}$, f2, $\{\}_{SK}$=10∗n
[8]	f3, $\{\}_{Pub}$. f3, $\{\}_{Pub}$, f2, $\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{SK}$=8∗n
[3]	f2, $\{\}_{SK}$, $\{\}_{SK}$, f1, f2, f1, $\{\}_{DK}$=7∗n
[9]	f2, $\{\}_{SK}$, $\{\}_{SK}$, f1, f2, f1, $\{\}_{DK}$=7∗n
[25]	$\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{Pub}$, $\{\}_{SK}$, f2, $\{\}_{SK}$, f2, $\{\}_{SK}$=10∗n
Making Payment	f2, $\{\}_{SK}$, f3, $\{\}_{SK}$, f3, $\{\}_{SK}$, f2, $\{\}_{SK}$, f3, $\{\}_{SK}$, f3, $\{\}_{SK}$, $\{\}_{SK}$, $\{\}_{SK}$, $\{\}_{SK}$, f2, $\{\}_{SK}$=17∗n

In the case of message M3, after receiving the message from the mobile operator, the merchant verifies the goods or services. If these are valid, a *Yes* message will be sent to the mobile operator, but if they are invalid a *No* message will be sent to the mobile operator and then the mobile operator will terminate the client's request. The merchant then sends the message $\{Yes/No, h(Yes, OI, SK_{CMj+1})\}_{SKMOj+1}$, $\{h(Yes, OI)\}_{SKMTTPj}$ to the mobile operator. The message that contains $\{Yes/No, h(Yes, OI, SK_{CMj+1})\}_{SKMOj+1}$ is for the mobile operator to verify the authenticity of the merchant. It should be noted that the mobile operator does not know the session key SK_{CMj+1} that is used to construct the message $h(Yes, OI, SK_{CMj+1})$ since it cannot generate this message by itself. This message contains $h(Yes/No, OI, SK_{CMj+1})$ in order to guarantee information correctness and to defend against erroneous information alteration or destruction. The message $\{h(Yes, OI)\}_{SKMTTPj}$ will be transmitted to *TTP* via the mobile operator.

In the case of message M4, upon receiving this message from the merchant, the mobile operator checks the credit balance and compares it with the requested amount. If the client has enough credit, the mobile operator will reply with an *Accept* message to the client. If not, the mobile operator will reply with a *Reject* message, and then the client has to return to the Purchase Credit Request Phase. The mobile operator then sends message $\{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}$, $\{h(Accept/Reject, OI, CL_{RM})\}_{SKOTTPj}$, $\{h(Yes, OI)\}_{SKMTTPj}$ to *TTP*.

In the case of message M5, after receiving the message from the mobile operator, *TTP* decrypts the message $\{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}$ with SK_{CTTPj}, $\{h(Accept/Reject, OI, CL_{RM})\}_{SKOTTPj}$ with SK_{OTTPj} and $\{h(Yes, OI)\}_{SKMTTPj}$ with SK_{MTTPj}, and it keeps the decrypted message to itself. Then *TTP* encrypts the message $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKCTTPj+1}$ with $SK_{CTTPj+1}$, $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKOTTPj+1}$ and encrypts the message with $SK_{OTTPj+1}$ and $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKMTTPj+1}$. *TTP* also encrypts the message with $SK_{MTTPj+1}$ and sends all the messages to the mobile operator.

In the case of message M6, the mobile operator forwards the message $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKMTTPj+1}$, which is encrypted by *TTP* with $SK_{MTTPj+1}$ to the merchant.

In the case of message M7, the mobile operator encrypts message one $\{Accept/Reject, Yes, CL_{RM}, h(Yes, OI, SK_{CMj+1}), h(OI, SK_{MOj+1})\}_{SKCOj+1}$ with SK_{COj+1}, while the message $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKCTTPj+1}$ is encrypted by TTP with $SK_{CTTPj+1}$ to the client. The message that contains $\{Accept/Reject, Yes, CL_{RM}, h(Yes/No, OI, SK_{CMj+1}), h(OI, SK_{MOj+1})\}_{SKCOj+1}$ is for the client to verify the authenticity of the mobile operator. It can be seen that the mobile operator cannot deny that the message has originated from him/her because only the mobile operator possesses both SK_{MOj+1} and SK_{COj+1}. This means that the mobile operator is the only one that can generate this message. The hash value of $h(Yes/No, OI, SK_{CMj+1})$ notes that the mobile operator does not know the session key SK_{CMj+1} that is used to construct the message $h(Yes, OI, SK_{CMj+1})$ and the client to verify the authenticity of the merchant, since it cannot generate this message by itself.

3.7. Dispute Resolution Phase.

After the transaction is complete, if the client is not satisfied with the transaction, he/she can request a dispute resolution with the verifier. The dispute resolution protocols are composed of two subprotocols: Purchase Credit Request Phase and Making Payment Phase. Consider the protocols below.

3.7.1. Purchase Credit Request.

The client sends the hash value $h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)$ of the transaction to the verifier. Upon receiving the hash value from the client, the verifier sends the requested hash value of *TTP*. After receiving the hash value from *TTP*, the verifier compares the hash value from the client with a hash value from *TTP*. If the hash values do not match, the verifier rejects the client's request; if not, the verifier sends a notification of dispute resolution to the mobile operator to transfer the amount to the client's account. Note that the verifier stands for the external party and is a party that is not relevant to the particular transaction.

3.7.2. Making Payment.

The client sends the hash value $h(ID_C, ID_M, OI, T), h\ (OI, CL_{RM}), h(Yes/No, OI)$ of the transaction to the verifier. Upon receiving the hash value from the client, the verifier sends the requested hash value of *TTP*. After receiving the hash value from *TTP*, the verifier compares the hash value from the client with a hash value from *TTP*. If the hash values do not match, the verifier rejects the client's request; if not, the verifier sends a notification of dispute resolution to the mobile operator to transfer the amount to the client's account and sends notification to the merchant. Note that the verifier stands for the external party and is a party that is not relevant to the particular transaction.

It is obvious that, in the Purchase Credit Request Phase (Section 3.5) and the Making Payment Phase (Section 3.6), all messages received by the *TTP* are only the hash values of the transactional data. Therefore, it is not possible for the *TTP* to

Table 10: Storage complexity.

Protocol	Storage cost
[1]	(1)+(2)+(3)+(4)=(80+80+160+24)+(80+160+64)+(80+24=104)+(24+80)=856∗n
[5]	(1)+(2)+(3)+(4)=(160)+(64+160+80)+(64+48)+(64)=640∗n
[6]	(1)+(2)+(3)= (128)+(160+80+104+160+64+128)+(80+80+128+128)=1240∗n
[7]	(1)+(2)+(3)=(80+128+80+128+80+128+80+128)+(128+128+128+80+128+128+128+80)+(80+64+128+128+80+160)=2400∗n
[8]	(1)+(2)+(3)=(40+80+128+24)+(128+128+64+160)+(128)=880∗n
[3]	(1)+(2)+(3)=(80+80+160)+(80+160+64)+(80)=704∗n
[9]	(1)+(2)+(3)=(80+80+160)+(80+160+64)+(80)=704∗n
[25]	(1)+(2)+(3)=(80+128+128+80+80+128+128+80)+(128+128+128+80+80+80+128+128+128+80+80+80)+ (128+80+128+128+80+128)=2752∗n
Making Payment	(1)+(2)=(80+80+80+160+80+160+160)+(48+24+40+160+160+160+160+160)=1712∗n

Table 11: Comparison of mobile payment protocols.

Protocol	Type of Fairness	Uses TTP	TTP Type
[1]	No Fairness	No	Without TTP
[5]	No Fairness	No	Without TTP
[6]	No Fairness	No	Without TTP
[7]	No Fairness	No	Without TTP
[8]	No Fairness	No	Without TTP
[3]	No Fairness	No	Without TTP
[9]	Weak Fairness	Yes	Online
[25]	No Fairness	No	Without TTP
Purchase Credit	Strong	Yes	Online
Making Payment	Strong	Yes	Online

access meaningful information of the transaction. This makes our protocols have strong fairness.

4. Discussion

4.1. Security Analysis. (1) Brute force attacks: for the completion of a transaction of our protocols, the session keys change every time and are not reused; therefore, it is difficult for the attacker to find the correct session key that is shared between parties. In addition, according to [29], a brute force attack is difficult to complete by applying an offline key generation technique.

(2) Replay attack prevention: the attacker intersects the message and resends an old message. Our protocols are used only once with a fresh timestamp, so a replay attack can be prevented and is difficult to accomplish. For further details about this technique, the reader is referred to [29].

(3) Message integrity: the integrity of the message is the most important security property that any party can ensure so that the message guarantees information correctness and is defended against erroneous information alteration or destruction during the transmission. With our protocols, each message comprises a Message Authentication Code (MAC) value that ensures the recipient of the message, and the same recent key can be used to generate a new MAC. The received MAC is compared with the calculated MAC.

(4) Mutual authentication: this is used to check who the originator and the receiver of a message are. Our protocols deploy MACs and symmetric cryptographic operations. Only the originator and the receiver who share the same key will be able to encrypt and decrypt their messages. As a result of our protocols, each message can be used to identify the originator and the receiver so that the client, the merchant, and the mobile operator ensure mutual authentication. Consider the message below:

M1: C $\longrightarrow$ O: ID_C, T, $\{ID_M, OI, T, h(OI, SK_{CMj})\}_{SKCOj}$, $\{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}$

It can be seen that the client and the mobile operator share the session key SK_{COj}, but the mobile operator cannot generate this message by himself or herself because the mobile operator cannot generate $h(OI, SK_{CMj})$, as the session key SK_{CMj} is shared only between the merchant and the client. Only the session keys SK_{CMj} and SK_{COj} are known by the client, therefore guaranteeing that the client generated this message.

(5) Man-in-the-middle attacks: an attacker cannot impersonate a party by intercepting message passes in these protocols. With the use of limited-use session keys and proper cryptographic operations, no confidential information is revealed and the reuse of authentication information is limited. In addition, the session keys used in our protocols are changed constantly using strong encryption techniques

Table 12: Comparison of fair exchange protocols.

Protocol	Type of Fairness	Use TTP	TTP Type
[13]	Weak	Yes	Inline
[26]	Weak	Yes	Online
[15]	Weak	Yes	Online
[21]	Weak	Yes	Offline
[23]	Weak	Yes	Offline
[24]	Weak	Yes	Offline
[17]	Weak	Yes	Online
[11]	No Fairness	No	Without TTP
Purchase Credit	Strong	Yes	Online
Making Payment	Strong	Yes	Online

Table 13: Comparison security properties.

	[1]	[5]	[6]	[7]	[8]	[3]	[9]	[25]	Purchase Credit Request	Making Payment
Confidentiality	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Integrity	Y	N	N	Y	Y	Y	Y	Y	Y	Y
Mutual authentication	Y	N	N	Y	Y	Y	N	Y	Y	Y
Non-repudiation	N	Y	N	Y	Y	N	N	Y	Y	Y
Brute force attack	N	Y	Y	N	Y	N	Y	N	Y	Y
Replay attack prevention	Y	N	Y	Y	Y	Y	Y	Y	Y	Y
Man-in-the-middle attack	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
MS disclosure	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Over the air modification	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Impersonation attack	Y	Y	N	Y	Y	Y	N	Y	Y	Y
SMS spoofing	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 14: Notations of BAN logic [27].

Notation	Definitions
X, Y	Statement
P, Q	Parties
$P\mid\equiv X$	P believes in X
$P \triangleleft X$	P sees X
$P\mid\sim X$	P once said X
$P\mid\Longrightarrow X$	P has the jurisdiction over X
$\#(X)$	The formula X is fresh
$P \overset{X}{\leftrightarrow} Q$	P and Q may use the shared key K to communicate
$P \overset{X}{\Longleftrightarrow} Q$	The formula Y is a secret known only to P and Q
$\{X\}_K$	The formula X is encrypted under the key K
$(X)_K$	The hash value of X using K as key

and therefore it is not possible for the transmitted message to be analyzed by an attacker who fraudulently feigns a party.

(6) Nonrepudiation of transactions: ordinarily, an encrypted message with the private key of a public key encryption operation provides a nonrepudiation property: a party cannot decline the transactions he or she has performed. However, an encrypted message with a symmetric key of symmetric key encryption operations cannot provide a nonrepudiation property. Nevertheless, nonrepudiation properties can be satisfied by encrypting messages with a symmetric key encryption. Being able to collect evidence from each message, our protocols demonstrate the actions that all parties have performed in each transaction. To see the nonrepudiation properties of our proposed protocols, see the message below:

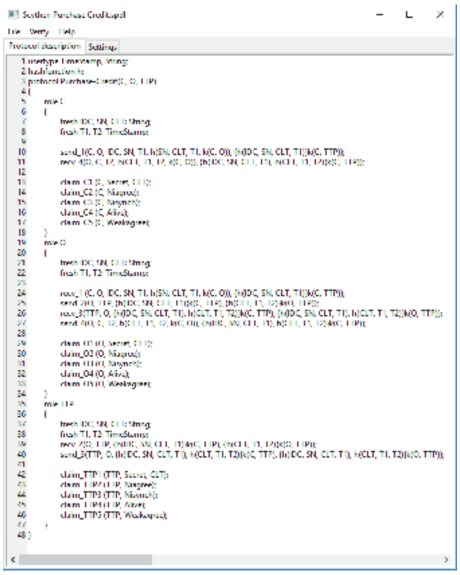
M1: C ⟶ O: $ID_C, T, \{ID_M, OI, T, h(OI, SK_{CMj})\}_{SKCOj}, \{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}$

It can be seen that the client cannot decline that the message he/she generated originated from him/her. This is because only the client possesses both SK_{CMj} and SK_{COj}. This indicates that only the client can generate this message.

(7) Confidentiality: generally, symmetric key encryption operations provide confidentiality of the security properties of the messages. Every message of our protocols applies symmetric key encryption operations to lead the confidentiality of the messages. The same session keys shared between the sender and receiver will be able to encrypt and decrypt their messages.

(8) SMS disclosure: at present, the confidentiality and integrity of the message do not provide the transmission of the SMS message. Our protocols use symmetric key encryption and hash functions to satisfy confidentiality and integrity from SMS disclosure of all the messages.

(9) Over-the-air modification: although the global system for a mobile communication network uses cryptographic



```
usertype TimeStamp, String;
hashfunction h;
protocol Purchase-Credit(C, O, TTP)
{
    role C
    {
        fresh IDC, SN, CLT: String;
        fresh T1, T2: TimeStamp;

        send_1(C, O, IDC, SN, T1, h(SN, CLT, T1, k(C, O)), {h(IDC, SN, CLT, T1)}k(C, TTP));
        recv_4(O, C, T2, h(CLT, T1, T2, k(C, O)), {h(IDC, SN, CLT, T1), h(CLT, T1, T2)}k(C, TTP));

        claim_C1 (C, Secret, CLT);
        claim_C2 (C, Niagree);
        claim_C3 (C, Nisynch);
        claim_C4 (C, Alive);
        claim_C5 (C, Weakagree);
    }
    role O
    {
        fresh IDC, SN, CLT: String;
        fresh T1, T2: TimeStamp;

        recv_1 (C, O, IDC, SN, T1, h(SN, CLT, T1, k(C, O)), {h(IDC, SN, CLT, T1)}k(C, TTP));
        send_2(O, TTP, {h(IDC, SN, CLT, T1)}k(C, TTP), {h(CLT, T1, T2)}k(O, TTP));
        recv_3(TTP, O, {h(IDC, SN, CLT, T1), h(CLT, T1, T2)}k(C, TTP), {h(IDC, SN, CLT, T1), h(CLT, T1, T2)}k(O, TTP));
        send_4(O, C, T2, h(CLT, T1, T2, k(C, O)), {h(IDC, SN, CLT, T1), h(CLT, T1, T2)}k(C, TTP));

        claim_O1 (O, Secret, CLT);
        claim_O2 (O, Niagree);
        claim_O3 (O, Nisynch);
        claim_O4 (O, Alive);
        claim_O5 (O, Weakagree);
    }
    role TTP
    {
        fresh IDC, SN, CLT: String;
        fresh T1, T2: TimeStamp;
        recv_2(O, TTP, {h(IDC, SN, CLT, T1)}k(C, TTP), {h(CLT, T1, T2)}k(O, TTP));
        send_3(TTP, O, {h(IDC, SN, CLT, T1), h(CLT, T1, T2)}k(C, TTP), {h(IDC, SN, CLT, T1), h(CLT, T1, T2)}k(O, TTP));

        claim_TTP1 (TTP, Secret, CLT);
        claim_TTP2 (TTP, Niagree);
        claim_TTP3 (TTP, Nisynch);
        claim_TTP4 (TTP, Alive);
        claim_TTP5 (TTP, Weakagree);
    }
}
```

Figure 2: SPDL script of Purchase Credit Request Phase.

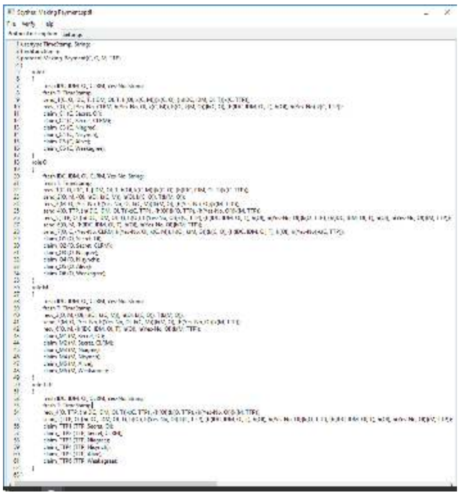
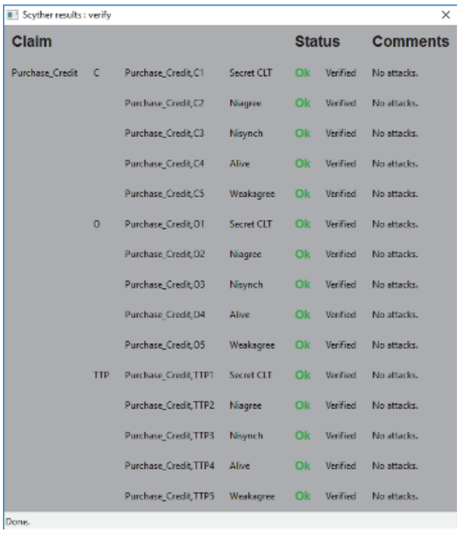


Figure 3: SPDL script of Making Payment Phase.



Claim				Status		Comments
Purchase_Credit	C	Purchase_Credit,C1	Secret CLT	Ok	Verified	No attacks.
		Purchase_Credit,C2	Niagree	Ok	Verified	No attacks.
		Purchase_Credit,C3	Nisynch	Ok	Verified	No attacks.
		Purchase_Credit,C4	Alive	Ok	Verified	No attacks.
		Purchase_Credit,C5	Weakagree	Ok	Verified	No attacks.
	O	Purchase_Credit,O1	Secret CLT	Ok	Verified	No attacks.
		Purchase_Credit,O2	Niagree	Ok	Verified	No attacks.
		Purchase_Credit,O3	Nisynch	Ok	Verified	No attacks.
		Purchase_Credit,O4	Alive	Ok	Verified	No attacks.
		Purchase_Credit,O5	Weakagree	Ok	Verified	No attacks.
	TTP	Purchase_Credit,TTP1	Secret CLT	Ok	Verified	No attacks.
		Purchase_Credit,TTP2	Niagree	Ok	Verified	No attacks.
		Purchase_Credit,TTP3	Nisynch	Ok	Verified	No attacks.
		Purchase_Credit,TTP4	Alive	Ok	Verified	No attacks.
		Purchase_Credit,TTP5	Weakagree	Ok	Verified	No attacks.

Done.

Figure 4: Output of Purchase Credit Request Phase.

techniques of A5/1 and A5/2 between the mobile device and the base station subsystem, they are still vulnerable [1–4]. Our protocols provide end-to-end security from the sender to the receiver by a strong encryption algorithm (such as AES).

(10) Impersonation attack: an attacker who impersonates a party cannot complete the attack because each message in our protocols provides mutual authentication of all parties of the transactions. Moreover, each message can be used to identify the sender and the receiver who share the same session keys. Our protocols can prevent impersonation attacks.

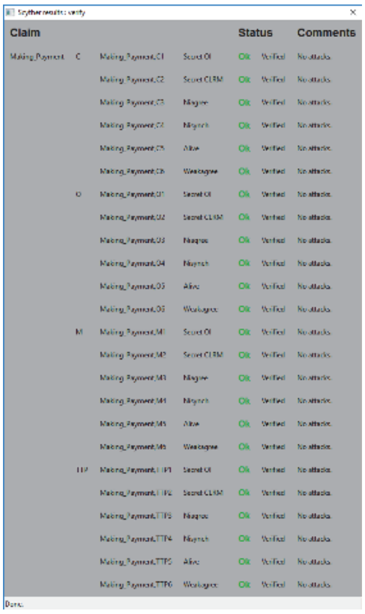
(11) SMS spoofing: an attacker can spoof the client by sending an SMS message with the correct headers via the Internet. Our protocols utilize a secure session key generation technique, in that only the sender and the receiver who share the same key will be able to encrypt and decrypt their messages. Moreover, our protocols on all parties provide mutual authentication. It can be seen that our protocols prevent SMS spoofing attacks.

4.2. Formal Security Verification of Our Protocols. The Scyther tool is used to verify the robustness and soundness of our protocols. The Scyther tool is a formal proofreader for security protocols. It is an automated security protocol analysis tool based on Security Protocol Description Language (SPDL), Scyther [36, 37]. SPDL provides three main protocol modelling features: roles, events, and claims. A lot of researchers have now utilized the Scyther tool to validate security protocols; see [25, 38–41]. The following security claims are verified in the analysis: secrecy of data (Secret), aliveness (Alive), weak agreement (Weakagree), noninjective agreement (Niagree), and noninjective synchronization (Nisynch) [37]. Our protocols are verified using the "Verification Claim" scheme in the Scyther tool.

Figures 2 and 3 display the SPDL script for verifying our protocol. Figures 4 and 5 show the output of the tests of our proposed protocols. Most of the claims are utilized to verify the security properties with a status (OK) for all the claims, and no attacks are discovered within bounds.

4.3. Performance Analysis of the Proposed Protocols. Our performance analysis follows the analysis of the mobile payment protocols [1, 3, 5–9, 25] and the fair exchange protocols [11, 13, 15, 17, 21, 24, 26] by focusing on several aspects related to the transaction performance, e.g., the number of cryptographic operations applied to each protocol and the number of messages passed in each protocol. We then show that our protocols have a greater performance than other existing SMS payment and fair exchange protocols. Our protocols utilize the symmetric encryption Advanced Encryption Standard (AES) 128 bits.

Tables 2 and 3 show symbols and abbreviations and cryptographic operation functions for use in this part of the paper in order to complete performance analysis.



```
Scyther results : verify                                                     ×
Claim                                                     Status      Comments
Making_Payment  C    Making_Payment,C1    Secret OI        Ok  Verified  No attacks.
                     Making_Payment,C2    Secret CLRM      Ok  Verified  No attacks.
                     Making_Payment,C3    Niagree          Ok  Verified  No attacks.
                     Making_Payment,C4    Nisynch          Ok  Verified  No attacks.
                     Making_Payment,C5    Alive            Ok  Verified  No attacks.
                     Making_Payment,C6    Weakagree        Ok  Verified  No attacks.
                O    Making_Payment,O1    Secret OI        Ok  Verified  No attacks.
                     Making_Payment,O2    Secret CLRM      Ok  Verified  No attacks.
                     Making_Payment,O3    Niagree          Ok  Verified  No attacks.
                     Making_Payment,O4    Nisynch          Ok  Verified  No attacks.
                     Making_Payment,O5    Alive            Ok  Verified  No attacks.
                     Making_Payment,O6    Weakagree        Ok  Verified  No attacks.
                M    Making_Payment,M1    Secret OI        Ok  Verified  No attacks.
                     Making_Payment,M2    Secret CLRM      Ok  Verified  No attacks.
                     Making_Payment,M3    Niagree          Ok  Verified  No attacks.
                     Making_Payment,M4    Nisynch          Ok  Verified  No attacks.
                     Making_Payment,M5    Alive            Ok  Verified  No attacks.
                     Making_Payment,M6    Weakagree        Ok  Verified  No attacks.
                TTP  Making_Payment,TTP1  Secret OI        Ok  Verified  No attacks.
                     Making_Payment,TTP2  Secret CLRM      Ok  Verified  No attacks.
                     Making_Payment,TTP3  Niagree          Ok  Verified  No attacks.
                     Making_Payment,TTP4  Nisynch          Ok  Verified  No attacks.
                     Making_Payment,TTP5  Alive            Ok  Verified  No attacks.
                     Making_Payment,TTP6  Weakagree        Ok  Verified  No attacks.
Done.
```

Figure 5: Output of Making Payment Phase.

Table 4 illustrates the message length of our protocols, which are considered as a lightweight cryptographic operation. Moreover, the message length of our protocols is less than the maximum message length of an SMS (160 bytes) in messages M1 and M4 (Purchase Credit Request Phase) and in messages M1 and M7 (Making Payment). Our protocols can be implemented in the current SMS infrastructure and therefore maintain ease of use.

Table 5 shows the SMS overhead of our protocols. The SMS overhead is calculated only with transactions between the client and the mobile operator. The client sends M1 (Purchase Credit Request Phase) to the mobile operator. The mobile operator sends M4 (Purchase Credit Request Phase) to the client. The client sends M1 (Making Payment) to the mobile operator. The mobile operator sends M7 (Making Payment) to the client. It can be seen that our protocols generate a minimum SMS overhead. Besides, the SMS overhead of our protocols is less than the maximum SMS overhead of an SMS (160 bytes).

Table 6 demonstrates the comparison of the cryptographic operations of our protocols with the proposed protocols in [1, 3, 5–9, 25]. Our protocols utilize different cryptography techniques such as a hash function and symmetric encryption. It can be inferred that our protocols use the minimum cryptographic operation.

The energy consumption of our protocols, according to [42], presents a framework for the energy consumption analysis of security protocols and cryptographic algorithms such as asymmetric, symmetric, hash function, and Message Authentication Code (MAC) algorithms. A number of frameworks have been proposed [43–45].

Table 7 shows the energy cost comparison of our protocols with other protocols [1, 3, 5–9, 25]. It can be seen that our protocols use a lower energy cost than the protocols proposed in [5–9, 25]. The protocols in [1, 3, 9] use less energy than our protocols. However, our protocols have a smaller number of messages than the protocols proposed in [1, 3, 9], so they take less time to accomplish every correlated transaction [1, 3, 9].

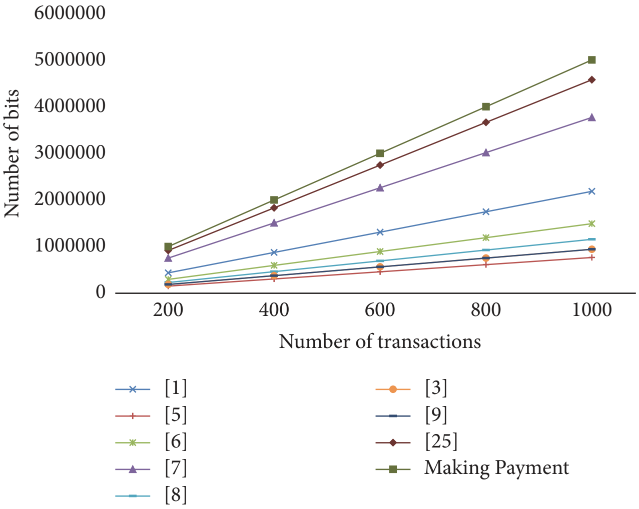


Figure 6: Communication overhead.

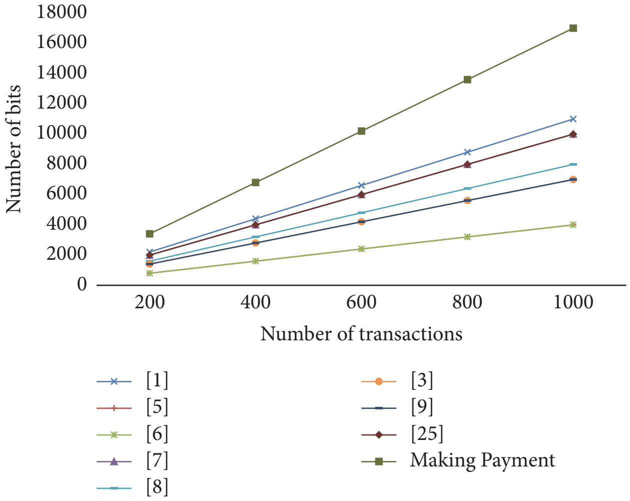


Figure 7: Computation overhead.

Table 8 and Figure 6 demonstrate comparisons of communication cost between the proposed protocols and existing protocols in [1, 3, 5–9, 25]. It can be seen that the proposed protocols has a higher communication cost than the protocols proposed in [1, 3, 5–9, 25]. Nevertheless, the SMS mobile payment protocols in [1, 3, 5–9, 25] are weak in terms of fairness according to the model described in Section 3.2. Moreover, in the case that one party misbehaves, the protocols proposed in [1, 3, 5–9, 25] cannot resolve any dispute because they do not provide the dispute resolution phase. Note that n is number of transactions.

From Table 9 and Figure 7, the protocols proposed in [1, 3, 5–9, 25] have lower computational cost than our protocols.

Table 10 and Figure 8 compare the storage costs of the protocols found in [1, 3, 5–9, 25] with the one of our protocols. It is clear that our protocols have a higher storage cost than

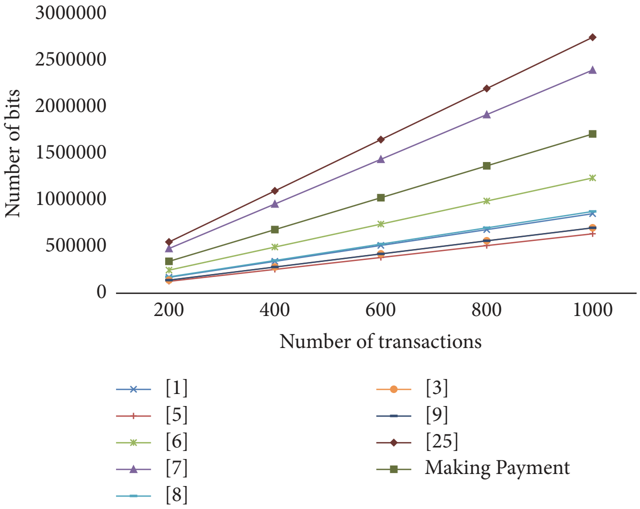


Figure 8: Storage overhead.

the protocols proposed in [1, 3, 5, 6, 8, 9] but lower than the ones in [7, 25].

Table 11 illustrates a fairness comparison between our proposed protocols and existing mobile payment protocols [1, 3, 5–8, 25]. As can be seen in Table 11, protocols proposed in [1, 3, 5–8, 25] have no trusted third party involved. Hence, those protocols do not have fairness according to the model presented in Section 3.2. Minta and Panchami [9] proposed an efficient encryption protocol for secure transmission of a confidential SMS. The Certification Authority and Registration Authority (CA/RA) are a trusted third party. It can read important information because CA/RA maintains a database containing the certificates of PK-SIM cards and Secure Access Gateway (SAGs) and Certificate Revocation Lists (CRLs). Thus, this protocol is weak in terms of fairness. In contrast, our proposed protocols satisfy the strong fairness as described Section 3.2.

Table 12 illustrates the comparison of the fairness aspect of different protocols. Mohammed [13] presented a transfer protocol ownership. The transaction server can read important information: a contract to be signed by both the buyer and the seller by the private key of the transaction server, the results of verification, buyer's account number, seller's account number, and amount. Hence, Mohammed's protocol is weak in terms of fairness according to the model described in Section 3.2. Chen [26] proposed a fair transaction model in mobile commerce, which is based on a personal trusted device. The customer of the business can read a considerable amount of information: the details of their purchases, the delivery address (which may be an email address or a physical address), dates, billing addresses, shipping addresses, the customer's own account, and the total price. Thus, Chen's protocol is also weak in terms of fairness. Alotaibi [15] proposed a new fair exchange protocol for trading goods online. The financial service provider can read all significant information: digital goods and invoices that are encrypted with temporary session keys of the financial service provider in the pre-exchange phase, the name of the financial institution of the customer, personal information about the customer, account details of the customer, and the total amount of money that the customer will pay for the digital goods. These are encrypted with the private key of the customer in the exchange phase. Therefore, Alotaibi's protocol is weak in terms of fairness. Mohammed [21] proposed fair exchange protocols between a customer and a merchant. The trusted third party can read important information: payments that are encrypted with the private key of the customer and digital goods that are encrypted with the session key of the merchant that is encrypted with the public key of the trusted third party. Therefore, Mohammed's protocol is also weak in terms of fairness. Alaraj [23] proposed a fair certified email protocol. The trusted third party can read a considerable amount of information: the mail that is encrypted with its sender's session key that is encrypted with the public key shared between the sender and the trusted third party. The trusted third party can decrypt messages with the private key shared between the trusted third party and the sender in the recovery subprotocol. Hence, Alaraj's protocol is weak in terms of fairness. Hinarejos [24] proposed an efficient and provable fair document exchange protocol. The trusted third party can read a considerable amount of information, such as parameters, which generate a session key to encrypt the document of party A and party B. Thus, Hinarejos's protocol is weak in terms of fairness. Liu [17] proposed a nonrepudiation protocol, which is based on a receiver-side smart card. The trusted third party can read significant information: a message to be exchanged between the customer and the vender. This message is encrypted with a session key generated by the trusted third party. Therefore, Liu's protocol is weak in terms of fairness. Paulin [11] proposed a fair nonrepudiation certified email protocol. This protocol either does or does not involve the trusted third party as the mediator between the senders and the receivers. In this way, Paulin's protocol has no fairness. Our protocols satisfy strong fairness according to the model presented in Section 3.2.

Table 13 shows a comparison of the security properties of the protocols found in the literature [1, 3, 5–7, 9, 25] and our protocols. It is clear that our protocols and [8] satisfy the necessary security properties, which are confidentiality, integrity, mutual authentication, nonrepudiation, brute force attack, replay attack prevention, man-in-the-middle attack and SMS disclosure, over-the-air modification, impersonation attack, and SMS spoofing. Note that Y and N mean "satisfying" and "unsatisfying".

5. Analysis of Fairness

In this section, we analyze our proposed protocols in terms of the fairness properties of the Purchase Credit Request Phase and the Making Payment Phase. We also provide some guidelines to prove the fairness of our proposed protocols. It can be seen that, to complete a transaction in the client's opinion, the *payment-order* response from the mobile operator must contain the amount requested by the client, and the *debit* response from the mobile operator must contain the amount where the client has requested the mobile operator to deduct from his/her account. In other words, the client not only has to receive payment responses, but also has to receive *payment-order* and *debit* from the mobile operator (on behalf of the mobile operator), respectively. All requests are sent to the trusted third party. According to the model stated in Section 3.2, our protocols then satisfy the fairness properties for all parties: the clients, the mobile operator, and the merchant.

Our analysis of fairness is derived from [46]. The reader may find more details about Kungpisdan's logic from [46].

5.1. Terms

(i) $\{P, Q, R, V\}$*:* a set of engaging parties

(ii) $\{X, Y\}$: a set of messages or message components in a protocol

(iii) $\{\varphi\}$: a set of statements derived from messages

(iv) $\xrightarrow{K} Q$: key K can be used to refer to Q

(v) $P \overset{K}{\longleftrightarrow} Q$: key K is the shared key between P and Q

(vi) *X-is-fingerprint-of-Y*: X is a fingerprint of Y

(vii) $\langle \mathrm{X} \rangle_K$: X applied with a single-key operation with key K. $\langle X \rangle_K$ can be any kind of message that is relevant to a single key, MAC (Message Authentication Code), or

(viii) even *h(K)*

(ix) *K-is-deriving-key-for-*$\langle X \rangle_K$: K can be used as a key to derive message X from a single-key cryptographic message

5.2. Formulae

(i) *P believes* φ: P believes that the statement φ is true.

(ii) *P sees X*: Some party has sent a message X to P and P is able to read X.

(iii) *P has X*: P possesses a message X. P can send X to other parties or use it for further computations.

(iv) *P says X*: P has sent message X.

(v) *P CanProve* φ *to Q*: P can prove to Q that statement φ is true.

(vi) *P authorized payment (P, Q, OI, datetime)*: P has authorization to make the payment amount *Price* to Q on *datetime*, the datetime of transaction.

(vii) *P authorized debit (P, Q, OI, datetime)*: P has authorization to request Q to deduct the amount *Price* from P's account on *datetime*, the datetime of transaction.

(viii) *P authorized credit (P, Q, OI, datetime)*: P has authorization to request Q to transfer the amount *Price* to P's account on *date*time, the datetime of transaction.

5.3. Axioms

5.3.1. Comprehensions. **C1:** *P sees X* $\longrightarrow$ *P believes P sees X*

5.3.2. Inference Rules. **M:** If φ is a theorem, then P believes φ is a theorem, where theorem is a formula, which can be derived from axioms alone.

5.3.3. Possessions. **H1:** *P sees X* $\longrightarrow$ *P has X.*

H2: *(P has* $X_1 \bigwedge \ldots \bigwedge$ *P has* X_n*)* $\longleftrightarrow$ *P has* $(X_1, \ldots, X_n)$, *where* $(X_1, \ldots, X_n)$ stands for a list of messages $X_1, X_2, \ldots, X_n$, respectively.

H3: *P has X* $\longrightarrow$ *P has h(X).*

H4: *(P has* $(\{X\}_K, K) \bigwedge$ *P believes* $P \overset{K}{\longleftrightarrow} Q)$ $\longrightarrow$ *P has X.*

5.3.4. Provability. **P2:** *[V-is-external-party* $\bigwedge$ *P has X* $\bigwedge$ *(V sees X* $\longrightarrow$ *V believes* φ*)]* $\longrightarrow$ *P CanProve* φ *to V.*

P3': If P has $\{M, X\}_K$ and a key K', P believes that K' is shared between Q and a party P', P can prove to V that K' can be used to decrypt $\{M, X\}_K$, and P can also prove to V that X is shared between Q and R, and then P can prove to V that Q has sent X to P'.

[P has $(\{M, X\}_K, K')$ $\bigwedge$ *P believes* $P' \overset{K}{\longleftrightarrow} Q$ $\bigwedge$ *P CanProve* $(K'$-is-decrypting-key-for-$\{M, X\}_K)$ *to V*

$\bigwedge$ *P CanProve* $(Q \overset{X}{\longleftrightarrow} R)$ *to V]*

$\longrightarrow$ *P CanProve (Q says* $(M, X, ID_{P'})$*) to V*

P6: *P CanProve (Q says* $(X_1, \ldots, X_n)$*) to V*

$\longleftrightarrow$ *[P CanProve (Q says* X_1*) to V* $\bigwedge \ldots \bigwedge$ *P CanProve (Q says* X_n*) to V]*

5.4. Initial Assumptions. The following assumptions are specific to our fairness protocols.

5.4.1. General Assumptions. **A1:** Every party believes that if V believes that K' is shared between parties Q and R, V has both $\langle X \rangle_K$ and K', and K' can be used to extract X from $\langle X \rangle_{K'}$, then V believes that $\langle X \rangle_K$ is shared between Q and R.

P believes [(V believes $Q \overset{K'}{\longleftrightarrow} R \bigwedge$ *V has* $(\langle X \rangle_K, K') \bigwedge X = \langle\langle X \rangle_K \rangle_{K'})$ $\longrightarrow$ *V believes* $Q \overset{\langle X \rangle_K}{\longleftrightarrow} R$*]*

Here Q and R stand for any two participants; assumption A1 together with axiom P2 is used for reason that if V believes that K is a secret shared between two parties, then any message that is applied with the single-key operation relevant to K must be considered a shared secret as well.

A2: Every party believes that if V believes that he/she does not generate $\langle X \rangle_K$ by himself/herself, K' is shared between P' and Q, V has both $\langle X \rangle_K$ and K', and K' can be used to extract X from $\langle X \rangle_{K'}$, then V believes that Q has sent X to party P'.

P believes (V believes P' sees $\langle X \rangle_K \bigwedge$ *V believes* $P' \overset{K'}{\longleftrightarrow} Q$ $\bigwedge$ *V has* $(\langle X \rangle_{K'}, K') \bigwedge X = \langle\langle X \rangle_K \rangle_{K'}$ $\longrightarrow$ *V believes Q says* $(X, ID_{P'})$*)*

A3: Every party believes that if V has messages X and Y and message X is *h(Y),* then V believes that X is the fingerprint of Y.

P believes [(V has (X, Y) $\bigwedge X = h(Y)$ *)* $\longrightarrow$ *V believes X-is-fingerprint-of-Y]*

A4: Every party believes that if V has key K' and the single-key cryptographic message $\langle X \rangle_K$ and K' can be used to derive $\langle X\rangle_K$, then V believes that K' is the deriving key for $\langle X\rangle_K$.

P believes (V has $\langle X\rangle_K, K')$ $\bigwedge X = \langle\langle X\rangle_K\rangle_{K'} \longrightarrow$ *V believes K'-is-deriving-key-for-*$\langle X\rangle_K)$

5.4.2. Possessions. **A5:** Each player has their own identities.
P believes *P* has ID_P, *C believes C has* ID_M,
M believes M has ID_C, *O believes O has* ID_M,
O believes O has ID_C.

5.4.3. Shared Secrets. **A6:** The client and the merchant believe that SK_{CM} and $\{N\}_{SKCM}$ are shared between them, and they also have SK_{CM}. The client and the mobile operator believe that SK_{CO} and $\{N\}_{SKCO}$ are shared between them, and they also have SK_{CO}. The merchant and the mobile operator believe that SK_{MO} and $\{N\}_{SKMB}$ are shared between them, and they also have SK_{MO}.

P believes $C \xleftrightarrow{SK_{CM}} M$, *P believes* $C \xleftrightarrow{\{N\}_{SKCM}} M$, *P believes P has* SK_{CM} *where P* denotes *C* and M, and *N* is a message.

Q believes $C \xleftrightarrow{SK_{CO}} O$, *P believes* $C \xleftrightarrow{\{N\}_{SKCO}} O$, *Q believes Q has* SK_{CO} *where Q denotes C and O.*

R believes $M \xleftrightarrow{SK_{MO}} O$, *R believes* $M \xleftrightarrow{SK_{MO}} O$, *R believes R has* SK_{MO} *where R* denotes *M* and *O*.

A7: Each player believes that the merchant does not receive SK_{CO}, the mobile operator does not receive SK_{CM}, and the client does not receive SK_{MO}.
P believes $\neg$ *M sees* SK_{CO}, *P believes* $\neg$ *O sees* SK_{CM},
P believes $\neg$ *C sees* SK_{MO}

5.4.4. Verifier's Beliefs. **A8:** Each player believes that the verifier is an external party to whom every participant is able to reveal its secrets.

P believes V-is-external-party

Q believes V believes $C \xleftrightarrow{SK_{CM}} M$, *Q believes V believes* $C \xleftrightarrow{\{N\}_{SKCM}} M$, *Q believes V believes* $C \xleftrightarrow{SK_{CO}} O$

Q believes V believes $C \xleftrightarrow{\{N\}_{SKCO}} O$, *Q believes V believes* $M \xleftrightarrow{SK_{MO}} O$, *Q believes V believes* $M \xleftrightarrow{\{N\}_{SKMO}} O$

Q believes $\neg V$ *believes* $C \xleftrightarrow{M_1} O$, *Q believes* $\neg V$ *believes* $C \xleftrightarrow{M_2} O$, *Q believes* $\neg V$ *believes* $M \xleftrightarrow{M_3} O$

Here Q denotes any participant, *N* denotes a message or a message component, M_1 denotes the message that is different from SK_{CM} and $\{N\}_{SKCM}$, M_2 denotes the message that is different from SK_{CO} and $\{N\}_{SKCO}$, and M_3 denotes the message that is different from SK_{MO} and $\{N\}_{SKMO}$.

5.4.5. Payment Information. **A9:** The client and the merchant believe that they possess the price and statement.

Q believes Q has (OI) where *Q* denotes *C* and *M*. Note that the merchant has *OI* because he/she has the information about goods or services such as the description and price.

5.4.6. Payment Authorizations. **A10:** Each player believes that he/she can prove to the verifier that if the client has sent the message containing ID_M identity, price, and the datetime of transaction, the client has authorization to order goods or services from the merchant.

P believes P CanProve (C says (ID_M*, OI, datetime)*
$\longrightarrow$ *C authorized payment-order(C, M, OI, datetime)) to V*

Each player believes that he/she can prove to the verifier that if the merchant has sent the message containing ID_C, price, and the datetime of the transaction, the merchant has authorization to issue the payment receipt to the client.

P believes P CanProve (M says (ID_C*, OI, datetime)*
$\longrightarrow$ *M authorized payment-order(C, M, OI, datetime)) to V*

Each player believes that he/she can prove to the verifier that if the mobile operator has sent the message containing ID_C*, OI,* and the datetime of transaction, the mobile operator has authorization to debit from the client's account.

P believes P CanProve (O says (ID_C*, OI, datetime)*
$\longrightarrow$ *O authorized debit(C, O, OI, datetime)) to V*

Each player believes that he/she can prove to the verifier that if the client has sent the message containing *OI* and the datetime of the transaction, then the client has authorization to request the mobile operator to deduct the requested amount from his/her account.

P believes P CanProve (C says (OI, datetime)
$\longrightarrow$ *C authorized debit(C, O, OI, datetime)) to V*

Each player believes that he/she can prove to the verifier that if the mobile operator has sent the message containing ID_M*, OI,* and the datetime of the transaction, then the mobile operator has authorization to transfer the requested amount to the account of the merchant.

P believes P CanProve (O says (ID_M*, OI, date)*
$\longrightarrow$ *O authorized credit(M, O, OI, datetime)) to V*

Each player believes that he/she can prove to the verifier that if the merchant has sent the message containing *OI* and the datetime of the transaction, then the merchant has authorization to request the mobile operator to transfer the requested amount to the merchant account.

P believes P CanProve (M says (OI, datetime)
$\longrightarrow$ *M authorized credit(M, O, OI, datetime)) to V*

5.5. The Goals of Analysis. To evaluate the fairness, we specify the goals regarding the fundamental interactions between pairs of engaging parties: the payment-order between the client and the merchant, the debiting between the client and the mobile operator, and the crediting between the merchant and the mobile operator. We state the goals G1-G6 by associating them with the abilities to prove the primitive interactions stated in Section 3.2 as follows:

G1: *M believes M CanProve (C authorized payment-order(C, M, OI, datetime)) to V*

G2: *C believes C CanProve (M authorized payment-order(M, C, OI, datetime)) to V*

G3: *O believes O CanProve (C authorized debit(C, O, OI, datetime)) to V*

G4: *C believes C CanProve (O authorized debit(O, C, OI, datetime)) to V*

G5: *O believes O CanProve (M authorized credit(M, O, OI, datetime)) to V*

G6: *M believes M CanProve (O authorized credit(O, M, OI, datetime)) to V*

5.6. Details of the Proof. Due to the limited space, we are not able to describe all details of our fairness analysis. However, we can provide some guidelines to prove G4 of the Purchase Credit Request Phase and G1 of the Making Payment Phase as follows.

G4: *C believes C CanProve (O authorized debit(O, C, OI, datetime)) to V*

Consider message 4 of Purchase Credit Request Phase,

M4: O ⟶ C: T_2, $h(CL_T, T_1, T_2, SK_{COj+1})$, $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}$

It can be transformed into:

C sees T_2, $h(CL_T, T_1, T_2, SK_{COj+1})$, $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}$

Detail of the proof can be shown as follows:

C sees T_2, $h(CL_T, T_1, T_2, SK_{COj+1})$, $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}$ (1)
1, C1, M: *C believes C sees* T_2, $h(CL_T, T_1, T_2, SK_{COj+1})$, $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}$. (2)
2, H1, H2, M: *C believes C has* T_2, $h(CL_T, T_1, T_2, SK_{COj+1})$, $\{h(ID_C, SN, CL_T, T_1), h(CL_T, T_1, T_2)\}_{SKCTTPj+1}$ (3)
3, A6, H2, M: *C believes C has* T_2, $h(CL_T, T_1, T_2, SK_{COj+1})$, CL_T, T_1, SK_{COj+1} (4)
A7, 4, A2, P2, M: *C believes V-is-external-party* $\bigwedge$ (5)
C believes C has T_2, $h(CL_T, T_1, T_2, SK_{COj+1})$, CL_T, T_1, SK_{COj+1} $\bigwedge$
C believes (V believes $(C \xleftrightarrow{SK_{COj+1}} O)$ $\bigwedge$
V believes C sees $h(CL_T, T_1, T_2, SK_{COj+1})$ $\bigwedge$
V has T_2, $h(CL_T, T_1, T_2, SK_{COj+1})$, CL_T, T_1, SK_{COj+1} $\bigwedge$
$h(CL_T, T_1, T_2, SK_{COj+1}) = h(CL_T, T_1, T_2, SK_{COj+1})$
⟶ *V believes O says* (CL_T, T_1, T_2)
⟶ *C believes C CanProve (O says* CL_T, T_1, T_2*) to V*
5, P6, M: *C believes C CanProve (O says* CL_T, T_1, T_2*) to V* (6)

The goal G4 is successfully proved. Thus, it can be concluded that the merchant is satisfied with the fairness. Note that the details of the analysis of goals G1–G3 and G5–G6 were successfully analyzed.

Consider message 2 of Making Payment Phase.

G1: *M believes M CanProve (C authorized payment-order(C, M, OI, datetime)) to V*

Consider message 2 of Making Payment Phase,

M2: O ⟶ M: $\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj}$

It can be transformed into:

M sees $\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj}$

Detail of the proof can be shown as follows:

M sees $\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj}$ (1)
1, C1, H1, H2, M: *M believes M has* $\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj}$ (2)
2, A6, H2, M: *M believes M has* $(\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj})$ (3)
A7, 3, A5, P2, M: *M believes M CanProve* $(SK_{MOj}$*-is-decrypting-key-for-*$\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj})$ *to V* (4)
3, A6, H4, M: *M believes M has* $(OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T)$ (5)
5, H2, M: *M believes M has* $h(OI, SK_{CMj})$ (6)
4, H3, M: *M believes M has* $(h(OI))$ (7)
7, A5, H2, M: *M believes M has* $(h(OI), SK_{CMj})$ (8)
A7, 6, 8, H2, A1, P2, M: *M believes M CanProve* $(C \xleftrightarrow{\boldsymbol{h}(\mathbf{OI},SK_{CMj})} M)$ *to V* (9)
3, A6, 4, 9, P3', M: *M believes M CanProve (C says* $(OI, h(OI, SK_{CMj}), T)$*) to V* (10)
10, P6, M: *M believes M CanProve (C says (OI, T)) to V* (11)
11, A9, M: *M believes M CanProve (C authorized payment-order(C, M, OI)) to V* (12)

The goal G1 is successfully proved. Thus, it can be concluded that the merchant is satisfied with the fairness. Note that the details of the analysis of goals G2–G6 were successfully analyzed.

6. Security Proof Based on BAN Logic

We use BAN logic model to prove the mutual authentication of our protocols. The BAN logic [27] is a well-known formal model, which is used to examine the security of mutual authentication. The literature in [47] has now utilized the BAN logic to verify the security of mutual authentication. The notations used in BAN logic are defined in Table 14.

Moreover, the proposed protocols use six rules of BAN logic to prove a secure mutual authentication between parties:

R1. Message-meaning rule: $P| \equiv P \overset{K}{\longleftrightarrow} Q, P \triangleleft \{X\}_K / P| \equiv Q| \sim X, P| \equiv P \overset{K}{\longleftrightarrow} Q, P \triangleleft (X)_K / P| \equiv Q| \sim X$.

R2. Nonce-verification rule: $P| \equiv \#(X), P| \equiv Q| \sim X / P| \equiv Q| \equiv X$.

R3. Jurisdiction rule: $P| \equiv Q \Longrightarrow X, P| \equiv Q| \equiv X / P| \equiv X$.

R4. Freshness rule: $P| \equiv \#(X) / P| \equiv \#(X, Y)$.

R5. Belief rule: $P| \equiv (X), P| \equiv Y / P| \equiv (X, Y)$.

R6. Decryption rules: $P| \equiv Q \overset{K}{\longleftrightarrow} P, P \triangleleft \{X\}_K / P \triangleleft X$.

We analyze both Purchase Credit Request and Making Payment Phases in order to complete a secure mutual authentication of all parties. Our protocols are listed as follows.

6.1. Purchase Credit Request Phase

6.1.1. Idealized Form

$$\textbf{M1: C} \longrightarrow \textbf{O:} \ (SN, CL_T, T_1, C \xleftrightarrow{SK_{COj}} O)$$

$$\textbf{M4: O} \longrightarrow \textbf{C:} \ (CL_T, T_1, T_2, C \xleftrightarrow{SK_{COj+1}} O)$$

6.1.2. Initial Assumptions

A1. $O| \equiv C \xleftrightarrow{SK_{COj}} O$, A2. $C| \equiv C \xleftrightarrow{SK_{COj+1}} O$,

A3. $O| \equiv \#(T_1)$, A4. $C| \equiv \#(T_2)$

6.1.3. The Goals of Analysis. G1. $O| \equiv (SN, CL_T, T_1, C \xleftrightarrow{SK_{COj}} O)$, G2. $C| \equiv (CL_T, T_1, T_2, C \xleftrightarrow{SK_{COj+1}} O)$

6.1.4. Details of the Proof. G1. $O| \equiv (SN, CL_T, T_1, C \xleftrightarrow{SK_{COj}} O)$

$O \triangleleft (SN, CL_T, T_1, C \xleftrightarrow{SK_{COj}} O)$ from message M1 1

1, R1: $O| \equiv O \triangleleft (SN, CL_T, T_1, C \xleftrightarrow{SK_{COj}} O)$ 2

2, R2, R4, A1, A3: $O| \equiv O$ *has* $(T_1, C \xleftrightarrow{SK_{COj}} O)$ 3

3, R5: $O| \equiv O$ *has* (SN, CL_T) 4

4: $O| \equiv (SN, CL_T, T_1, C \xleftrightarrow{SK_{COj}} O)$ 5

G2. $C| \equiv (CL_T, T_1, T_2, C \xleftrightarrow{SK_{COj+1}} O)$

$C \triangleleft (CL_T, T_1, T_2, C \xleftrightarrow{SK_{COj+1}} O)$ from message M4 1

1, R1: $C| \equiv C \triangleleft (CL_T, T_1, T_2, C \xleftrightarrow{SK_{COj+1}} O)$ 2

2, R2, A2: $C| \equiv C$ *has* $(CL_T, T_1, C \xleftrightarrow{SK_{COj+1}} O)$ 3

3, R4, R5, A4: $C| \equiv C$ *has* (T_2) 4

4: $O| \equiv (CL_T, T_1, T_2, C \xleftrightarrow{SK_{COj+1}} O)$ 5

The goals are successfully proved in Purchase Credit Request Phase. Thus, it can be concluded that all parties have satisfied a secure mutual authentication.

6.2. Making Payment Phase. **M1: C** $\longrightarrow$ **O:** $ID_C, T, \{ID_M, OI, T, h(OI, SK_{CMj})\}_{SKCOj}, \{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}$

M2: O $\longrightarrow$ **M:** $\{OI, h(OI, SK_{CMj}), h(OI, SK_{COj+1}), T\}_{SKMOj}$

M3: M $\longrightarrow$ **O:** $\{Yes/No, h(Yes, OI, SK_{CMj+1})\}_{SKMOj+1}, \{h(Yes, OI)\}_{SKMTTPj}$

M4: O $\longrightarrow$ **TTP:** $\{h(ID_C, ID_M, OI, T)\}_{SKCTTPj}, \{h(Accept/Reject, OI, CL_{RM})\}_{SKOTTPj}, \{h(Yes, OI)\}_{SKMTTPj}$

M5: TTP $\longrightarrow$ **O:** $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKCTTPj+1}, \{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKOTTPj+1}, \{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKMTTPj+1}$

M6: O $\longrightarrow$ **M:** $\{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, OI)\}_{SKMTTPj+1}$

M7: O $\longrightarrow$ **C:** $\{Accept/Reject, Yes, CL_{RM}, h(Yes, OI, SK_{CMj+1}), h(OI, SK_{MOj+1})\}_{SKCOj+1}, \{h(ID_C, ID_M, OI, T), h(Accept/Reject, OI, CL_{RM}), h(Yes, IO)\}_{SKCTTPj+1}$

6.2.1. Idealized Form. **M1: C** $\longrightarrow$ **O:** $\{IDM, OI, T, (OI, C \xleftrightarrow{SK_{CMj}} M)\}C \xleftrightarrow{SK_{COj}} O$

M2: O $\longrightarrow$ **M:** $\{OI, (OI, C \xleftrightarrow{SK_{CMj}} M), (OI, C \xleftrightarrow{SK_{COj+1}} O), T\}M \xleftrightarrow{SK_{MOj}} O$

M3: M $\longrightarrow$ **O:** $\{Yes/No, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)\}M \xleftrightarrow{SK_{MOj+1}} O$

M7: O $\longrightarrow$ **C:** $\{Accept/Reject, Yes, CL_{RM}, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M), (OI, M \xleftrightarrow{SK_{MOj+1}} O)\}C \xleftrightarrow{SK_{COj+1}} O$

6.2.2. Initial Assumptions. A1. $O| \equiv C \xleftrightarrow{SK_{COj}} O$, A2. $M| \equiv C \xleftrightarrow{SK_{CMj}} M$,

A3. $M| \equiv C \xleftrightarrow{SK_{MOj}} O$, A4. $O| \equiv C \xleftrightarrow{SK_{MOj+1}} O$,
A5. $C| \equiv C \xleftrightarrow{SK_{COj+1}} O$, A6. $C| \equiv C \xleftrightarrow{SK_{CMj+1}} M$,
A7. $O| \equiv \#(T)$.

6.2.3. The Goals of Analysis. G1. $O| \equiv \{ID_M, OI, T, (OI, C \xleftrightarrow{SK_{CMj}} M)\}C \xleftrightarrow{SK_{COj}} O$, G2. $M| \equiv (OI, C \xleftrightarrow{SK_{CMj}} M)$,

G3. $M| \equiv \{OI, (OI, C \xleftrightarrow{SK_{CMj}} M), (OI, C \xleftrightarrow{SK_{COj+1}} O), T\}M \xleftrightarrow{SK_{MOj}} O$, G4. $O| \equiv \{Yes/No, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)\}M \xleftrightarrow{SK_{MOj+1}} O$.

G5. $C| \equiv (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)$, G6. $C| \equiv \{Accept/Reject, Yes, CL_{RM}, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M), (OI, M \xleftrightarrow{SK_{MOj+1}} O)\}C \xleftrightarrow{SK_{COj+1}} O$.

6.2.4. Details of the Proof. G1. $O| \equiv \{ID_M, OI, T, (OI, C \xleftrightarrow{SK_{CMj}} M)\}C \xleftrightarrow{SK_{COj}} O$

$O \triangleleft \{ID_M, OI, T, (OI, C \xleftrightarrow{SK_{CMj}} M)\}C \xleftrightarrow{SK_{COj}} O$ from message M1 1

1, R1: $O| \equiv O \triangleleft \{ID_M, OI, T, (OI, C \xleftrightarrow{SK_{CMj}} M)\}C \xleftrightarrow{SK_{COj}} O$ 2

2, R6, A1: $O| \equiv O \textit{ has } \{ID_M, OI, T, (OI, C \xleftrightarrow{SK_{CMj}} M)\}C \xleftrightarrow{SK_{COj}} O$ 3

3, R4, A7: $O| \equiv O \triangleleft (ID_M, OI, T, (OI, C \xleftrightarrow{SK_{CMj}} M))$ 4

4: $O| \equiv \{ID_M, OI, T, (OI, C \xleftrightarrow{SK_{CMj}} M)\}C \xleftrightarrow{SK_{COj}} O$ 5

G2. $M| \equiv (OI, C \xleftrightarrow{SK_{CMj}} M)$

$M \triangleleft (OI, C \xleftrightarrow{SK_{CMj}} M)$ from message M1 1

1, R1: $M| \equiv M \triangleleft (OI, C \xleftrightarrow{SK_{CMj}} M)$ 2

2, A2: $M| \equiv M \textit{ has } (C \xleftrightarrow{SK_{CMj}} M)$ 3

3, R5: $M| \equiv M \textit{ has } (OI)$ 4

4: $M| \equiv (OI, C \xleftrightarrow{SK_{CMj}} M)$ 5

G3. $M| \equiv \{OI, (OI, C \xleftrightarrow{SK_{CMj}} M), (OI, C \xleftrightarrow{SK_{COj+1}} O), T\}M \xleftrightarrow{SK_{MOj}} O$

$M \triangleleft \{OI, (OI, C \xleftrightarrow{SK_{CMj}} M), (OI, C \xleftrightarrow{SK_{COj+1}} O), T\}M \xleftrightarrow{SK_{MOj}} O$ from message M2 1

1, R1: $M| \equiv M \triangleleft \{OI, (OI, C \xleftrightarrow{SK_{CMj}} M), (OI, C \xleftrightarrow{SK_{COj+1}} O), T\}M \xleftrightarrow{SK_{MOj}} O$ 2

2, R6, A3: $M| \equiv M \textit{ has } \{OI, (OI, C \xleftrightarrow{SK_{CMj}} M), (OI, C \xleftrightarrow{SK_{COj+1}} O), T\}M \xleftrightarrow{SK_{MOj}} O$ 3

3, R4, A7: $M| \equiv M \triangleleft (OI, (OI, C \xleftrightarrow{SK_{CMj}} M), (OI, C \xleftrightarrow{SK_{COj+1}} O), T)$ 4

4: $M| \equiv \{OI, (OI, C \xleftrightarrow{SK_{CMj}} M), (OI, C \xleftrightarrow{SK_{COj+1}} O), T\}M \xleftrightarrow{SK_{MOj}} O$ 5

G4. $O| \equiv \{Yes/No, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)\}M \xleftrightarrow{SK_{MOj+1}} O$

$O \triangleleft \{Yes/No, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)\}M \xleftrightarrow{SK_{MOj+1}} O$ from message M3 1

1, R1: $O| \equiv O \triangleleft \{Yes/No, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)\}M \xleftrightarrow{SK_{MOj+1}} O$ 2

2, R6, A4: $O| \equiv O \textit{ has } \{Yes/No, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)\}M \xleftrightarrow{SK_{MOj+1}} O$ 3

3: $O| \equiv O \triangleleft (Yes/No, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M))$ 4

4: $O| \equiv \{Yes/No, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)\}M \xleftrightarrow{SK_{MOj+1}} O$ 5

G5. $C| \equiv (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)$

$C \triangleleft (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)$ from message M3 1

1, R1: $C| \equiv C \triangleleft (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)$ 2

2, R2, A6: $C| \equiv C \textit{ has } (C \xleftrightarrow{SK_{CMj+1}} M)$ 3

3, R5: $C| \equiv C \textit{ has } (Yes, OI)$ 4

4: $C| \equiv (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M)$ 5

G6. $C| \equiv \{Accept/Reject, Yes, CL_{RM}, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M), (OI, M \xleftrightarrow{SK_{MOj+1}} O)\}C \xleftrightarrow{SK_{COj+1}} O$

$C \triangleleft \{Accept/Reject, Yes, CL_{RM}, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M), (OI, M \xleftrightarrow{SK_{MOj+1}} O)\}C \xleftrightarrow{SK_{COj+1}} O$ from message M4 1

1, R1: $C| \equiv C \triangleleft \{Accept/Reject, Yes, CL_{RM}, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M), (OI, M \xleftrightarrow{SK_{MOj+1}} O)\}C \xleftrightarrow{SK_{COj+1}} O$ 2

2, R6, A5: $C| \equiv C \text{ has } \{Accept/Reject, Yes, CL_{RM}, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M), (OI, M \xleftrightarrow{SK_{MOj+1}} O)\}C \xleftrightarrow{SK_{COj+1}} O$ 3

3: $C| \equiv C \triangleleft (Accept/Reject, Yes, CL_{RM}, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M), (OI, M \xleftrightarrow{SK_{MOj+1}} O))$ 4

4: $C| \equiv \{Accept/Reject, Yes, CL_{RM}, (Yes, OI, C \xleftrightarrow{SK_{CMj+1}} M), (OI, M \xleftrightarrow{SK_{MOj+1}} O)\}C \xleftrightarrow{SK_{COj+1}} O$ 5

The goals are successfully proved in the Making Payment Phase. Thus, it can be concluded that all parties have satisfied a secure mutual authentication.

7. Conclusion

In this paper, we have introduced protocols for mobile payments that satisfy the important attributes of both strong fairness and security of sale transactions. Those security properties include confidentiality, integrity, mutual authentication, and nonrepudiation. The notion of our approach is suitable for the existing SMS infrastructure. This is because hash functions have been employed, which keep the sizes of all messages small enough to fit the limitation of the SMS system. Moreover, our protocols have deployed the techniques of offline session key generation and distribution to create the transaction security while being able to retain the lightweight property. Based on our analysis, it has been proven that our protocols are more effective than others in terms of information security and strong fairness of the exchange. All of our protocols have been successfully analyzed using the Scyther and BAN logic tools to verify their completeness and soundness.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work has been funded by Faculty of Information Science and Technology, Mahanakorn University of Technology, Bangkok, Thailand.

References

[1] N. Saxena and N. S. Chaudhari, "EasySMS: A protocol for end-to-end secure transmission of SMS," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 7, pp. 1157–1168, 2014.

[2] M. Toorani and A. A. B. Shirazi, "SSMS - A secure SMS messaging protocol for the m-payment systems," in *Proceedings of the 13th IEEE Symposium on Computers and Communications, ISCC 2008*, pp. 700–705, mar, July 2008.

[3] N. Saxena and N. S. Chaudhari, "SecureSMS: A secure SMS protocol for VAS and other applications," *The Journal of Systems and Software*, vol. 90, no. 1, pp. 138–150, 2014.

[4] A. Biryukov, A. Shamir, and D. Wagner, *Real time cryptanalysis of A5/1 on a PC*, 2007, https://cryptome.org/a51-bsw.htm.

[5] A. Pourali, M. V. Malakooti, and M. H. Yektaie, "A secure SMS model in e-commerce payment using combined AES and ECC encryption algorithms," *Society of Digital Information and Wireless Communication*, p. 431, 2014.

[6] N. R. Kisore and S. Sagi, "A secure SMS protocol for implementing digital cash system," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics, ICACCI 2015*, pp. 1883–1892, ind, August 2015.

[7] S. Bojjagani and V. N. Sastry, "SSMBP: A secure SMS-based mobile banking protocol with formal verification," in *Proceedings of the 11th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob 2015*, pp. 252–259, are, October 2015.

[8] H. Rongyu, Z. Guolei, C. Chaowen, X. Hui, Q. Xi, and Q. Zheng, "A PK-SIM card based end-to-end security framework for SMS," *Computer Standards & Interfaces*, vol. 31, no. 4, pp. 629–641, 2009.

[9] M. Thomas and V. Panchami, "An encryption protocol for end-to-end secure transmission of SMS," in *Proceedings of the IEEE International Conference on Circuit, Power and Computing Technologies, ICCPCT 2015*, ind, March 2015.

[10] A. Alotaibi and H. Aldabbas, "A review of fair exchange protocols," *International Journal of Computer Networks & Communications*, vol. 4, no. 4, 2012.

[11] A. Paulin and T. Welzer, "A universal system for fair nonrepudiable certified e-mail without a trusted third party," *Computers & Security*, vol. 32, pp. 207–218, 2013.

[12] M. Jakobsson, "Ripping coins for a fair exchange," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 921, pp. 220–230, 1995.

[13] A. A. Mohammed, "Ownership transfer protocol," in *Proceedings of the Internet Technology and Secured Transactions*, 2010.

[14] R. H. Deng, "Practical protocols for certified electronic mail," *Journal of Network and Systems Management*, vol. 4, no. 3, pp. 279–296, 1996.

[15] A. Abdullah and H. Aldabbas, "Design and evaluation of a new fair exchange protocol based on an online TTP," *International Journal of Network Security & Its Applications*, vol. 4, no. 4, pp. 21–36, 2012.

[16] J. N. Luo, M. H. Yang, and S.-Y. Huang, "An Unlinkable Anonymous Payment Scheme based on near field communication," *Computers and Electrical Engineering*, vol. 49, pp. 198–206, 2016.

[17] J. Liu and L. Vigneron, "Design and verification of a non-repudiation protocol based on receiver-side smart card," *IET Information Security*, vol. 4, no. 1, pp. 15–29, 2010.

[18] W. Fan, H. Shu, E. Fife, and Q. Yan, "An enhanced-security fair E-payment protocol," in *Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering, CSIE 2009*, pp. 516–519, usa, April 2009.

[19] M. Ben-Or, O. Goldreich, S. Micali, and R. L. Rivest, "A fair protocol for signing contracts," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 36, no. 1, pp. 40–46, 1990.

[20] N. Asokan, M. Schunter, and M. Waidner, "Optimistic protocols for fair exchange," in *Proceedings of the 1997 4th ACM Conference on Computer and Communications Security*, pp. 6–17, April 1997.

[21] A. M. Alaraj and M. Munro, "Enforcing honesty in fair exchange protocols," *Advanced Information and Knowledge Processing*, vol. 52, pp. 451–479, 2010.

[22] A. M. Alaraj, "Purchase of physical products online," in *Proceedings of the 2012 International Conference on Multimedia Computing and Systems (ICMCS)*, pp. 937–940, Tangiers, Morocco, May 2012.

[23] A. M. Alaraj, "Fair certified email protocol," *International Journal of Computer & Technology*, vol. 10, no. 1, pp. 1255–1260, 2013.

[24] R.-J. Hwang and C.-H. Lai, "Provable fair document exchange protocol with transaction privacy for e-commerce," *Symmetry*, vol. 7, no. 2, pp. 464–487, 2015.

[25] S. Bojjagani and V. N. Sastry, "A secure end-to-end SMS-based mobile banking protocol," *International Journal of Communication Systems*, vol. 30, no. 15, Article ID e3302, pp. 1–19, 2017.

[26] C.-L. Chen, H.-Y. Lin, Y.-Y. Chen, and J.-K. Jan, "A fair transaction model in mobile commerce," in *Proceedings of the Signal Processing and Information Technology*, pp. 484–489, 2006.

[27] M. Burrows, M. Abadi, and R. Needham, "Logic of authentication," *ACM Transactions on Computer Systems*, vol. 8, no. 1, pp. 18–36, 1990.

[28] C. Thammarat, R. Chokngamwong, C. Techapanupreeda, and S. Kungpisdan, "A secure SMS mobile payment protocol ensuring fair exchange," in *IEE Proc. The 29th Circuit/Systems Computers and Communications*, pp. 163–166, Thailand, Phuket, 2014.

[29] S. Kungpisdan and S. Metheekul, "A secure offline key generation with protection against key compromise," in *Proceedings of the 13th World Multi-Conference on Systemics, Cybernetics and Informatics, WMSCI 2009, Jointly with the 15th International Conference on Information Systems Analysis and Synthesis, ISAS 2009*, pp. 63–67, usa, July 2009.

[30] H. Pagnia and F. C. Gärtner, "On the impossibility of fair exchange without a trusted third party," Technical Report TUD-BS-1999-02, Darmstadt University of Technology, Department of Computer Science, Darmstadt, Germany, 1999, pp. 1-15.

[31] O. Dandash, Y. Wang, P. D. Le, and B. Srinivasan, "Fraudulent internet banking payments prevention using dynamic key," *Journal of Networks*, vol. 3, no. 1, pp. 25–34, 2008.

[32] S. Kungpisdan, P. D. Le, and B. Srinivasan, "A limited-used key generation scheme for internet transactions," *Lecture Notes in Computer Science*, vol. 3325, 2005.

[33] H. H. Ngo, X. Wu, P. D. Le, C. Wilson, and B. Srinivasan, "Dynamic key cryptography and applications," *International Journal of Network Security*, vol. 10, no. 3, pp. 161–174, 2010.

[34] S. Kungpisdan, B. Srinivasan, and P. D. Le, "Lightweight mobile credit-card payment protocol," in *Progress in cryptology—INDOCRYPT 2003*, vol. 2904 of *Lecture Notes in Comput. Sci.*, pp. 295–308, Springer, Berlin, 2003.

[35] A. D. Rubin and R. N. Wright, "Off-line generation of limited-use credit card numbers," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 2339, pp. 196–209, 2002.

[36] C. Cremers, "The scyther tool: verification, falsification, and analysis of security protocols," in *Proceedings of the IEE on Computer Aided Verification*, pp. 414–418, Springer, Berlin/Heidelberg, Germany, 2008.

[37] C. Cremers and M. Sjouke, *Operational semantics and verification of security protocols*, Springer Science Business Media, 2012.

[38] C. Patsakis, K. Dellios, and M. Bouroche, "Towards a distributed secure in-vehicle communication architecture for modern vehicles," *Computers & Security*, vol. 40, pp. 60–74, 2014.

[39] Q. Cheng, S. Lu, and J. Ma, "Analysis and improvement of the Internet-Draft IKEv3 protocol," *International Journal of Communication Systems*, vol. 30, no. 9, Article ID e3194, 2017.

[40] M. Bilal and S. Kang, "Time-assisted authentication protocol," *International Journal of Communication Systems*, p. 16, 2017.

[41] Y. Ma, "NFC communications-based mutual authentication scheme for the internet of things," *International Journal of Network Security*, vol. 19, no. 4, pp. 631–638, 2017.

[42] N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha, "A study of the energy consumption characteristics of cryptographic algorithms and security protocols," *IEEE Transactions on Mobile Computing*, vol. 5, no. 2, pp. 128–143, 2006.

[43] D. Mishra, A. K. Das, A. Chaturvedi, and S. Mukhopadhyay, "A secure password-based authentication and key agreement scheme using smart cards," *Journal of Information Security and Applications*, vol. 23, pp. 28–43, 2015.

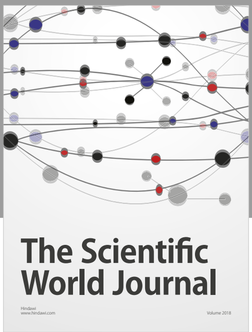
[44] R. Mishra and A. K. Barnwal, "A Privacy Preserving Secure and Efficient Authentication Scheme for Telecare Medical Information Systems," *Journal of Medical Systems*, vol. 39, no. 5, 2015.

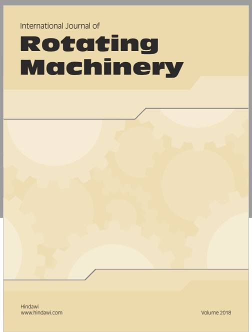
[45] L. Hu, L. Chi, H.-T. Li, W. Yuan, Y. Sun, and J.-F. Chu, "The classic security application in M2M: The authentication scheme of mobile payment," *KSII Transactions on Internet and Information Systems*, vol. 6, no. 1, pp. 131–146, 2012.

[46] S. Kungpisdan, "Accountability of centralized payment systems: Formal reasoning, protocol design and analysis," *IETE Technical Review*, vol. 27, no. 5, pp. 351–364, 2010.

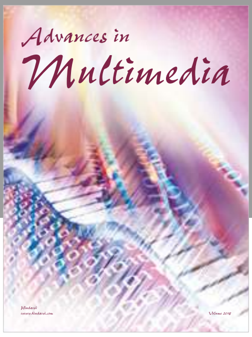
[47] Y. Chen, J. Martínez, P. Castillejo, and L. López, "A Privacy Protection User Authentication and Key Agreement Scheme Tailored for the Internet of Things Environment: PriAuth," *Wireless Communications and Mobile Computing*, vol. 2017, pp. 1–17, 2017.

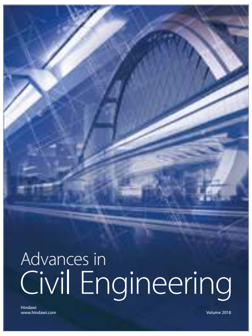
Journal of
Engineering

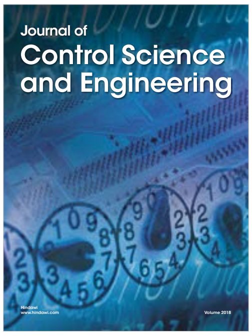

The Scientific World Journal


International Journal of
Rotating Machinery

Journal of
Sensors

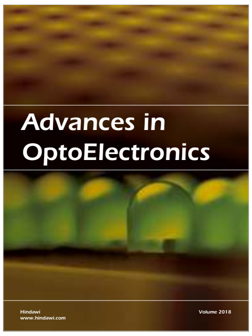

Advances in
Multimedia


Advances in
Civil Engineering


Journal of
Control Science and Engineering

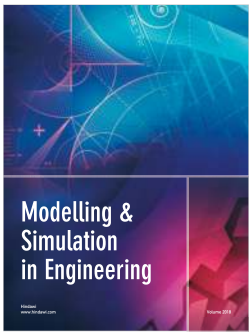
Journal of
Robotics

Journal of
Electrical and Computer Engineering


Advances in
OptoElectronics

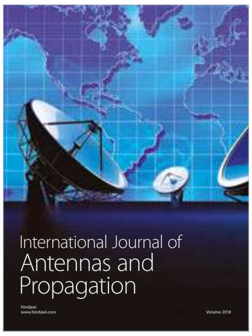
VLSI Design

International Journal of
Navigation and Observation

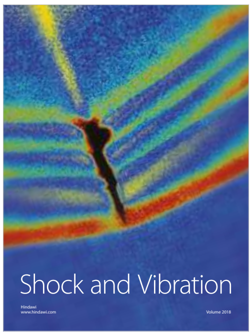

Modelling & Simulation in Engineering

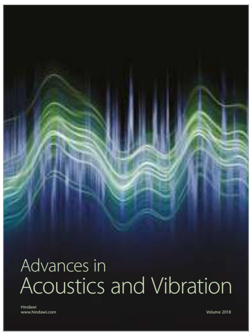
International Journal of
Aerospace Engineering

International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration