A Novel Micropayment Scheme with Variable Denomination

Quan-Yu Zhao[1], Yi-Ning Liu[2], Gao Liu[1] and Chin-Chen Chang[3]
(Co-corresponding authors: Yi-Ning Liu, Chin-Chen Chang)
School of Mathematics and Computational Science, Guilin University of Electronic Technology, Guilin 541004, China[1]
Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China[2]
Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan[3]
(Email: ynliu2011@gmail.com, alan3c@gmail.com)
(Received Oct. 19, 2015; revised and accepted Jan. 23 & Mar. 6, 2016)

Abstract

Recently, the research of micropayment systems has attracted a lot of attention in the literature. Because of its special feature that large amount of transactions may occur while each transaction only deals with small value, not only security but also efficiency should be carefully considered in these systems, especially when considering that these systems may be implemented in resource constrained mobile devices. In this paper, a novel lightweight micropayment scheme with variable denomination is proposed. Compared with those existing schemes with fixed denomination, our proposed scheme not only guarantees security goals, but also dramatically reduces the computational cost as well as the storage burden.

Keywords: Computation cost, hash chain, micropayment, storage burden, variable denominations

1 Introduction

With the development of the electronic commerce [10], electronic payments (e-payment) are widely used, which usually involves at least three entities: Customer ($\boldsymbol{C}$) who pays for the goods, Vendor ($\boldsymbol{V}$) who provides the goods, and Bank ($\boldsymbol{B}$) who helps the money transfer between $\boldsymbol{C}$ and $\boldsymbol{V}$ [18].

The e-payment scheme can be classified into macropayment [5] and micropayment [1, 23]. The transaction value in macro-payment is large enough, its security is guaranteed using complicated cryptographic techniques [16, 17]. However, this is unacceptable for micropayment since the transaction value is tiny, maybe it is less than the transaction cost. Therefore, the main goal of micropayment should be lightweight besides the security requirements such as no forgery and privacy as well.

The PayWord was firstly proposed by Rivest and Shamir in [14], which used the hash chain to replace the public key signature to improve the efficiency [8]. Inspired by these works, a lightweight micropayment scheme was proposed to enhance the fairness [9]. In [12], three schemes denoted as SPayword, UPayword and PPayword were presented to improve the robustness. Moreover, the version for multiple vendors was proposed in [3], and the self-renewal micropayment scheme was designed in [11]. In [21], Yang and Teng employed multiple Payword chains to find the minimum hash chain in transaction. Micropayment scheme presented in [4] is an anonymous fair offline scheme for all entities. Moreover, three micropayment schemes [6, 7, 15] were designed for the application over the mobile network.

The PayWord scheme is generalized into two categories: Micropayment Scheme using a Single-Payword Chain (MSSC), and Micropayment Scheme using Multi-Payword Chain (MSMC). Although each value of a hash chain is assigned with the same denomination, the denomination of each value in one hash chain is different from that of the others. Therefore, $\boldsymbol{C}$ can use the value of a hash chain to purchase in the MSSC scheme, and two hash chains are employed for the transaction in the MSMC scheme. Obviously, it is more efficient to utilize the multi-Payword hash chains [13, 19, 20] to reduce the computation cost.

However, the hash chain with small denomination might be used faster than the one with large denomination in the MSMC scheme, which might result in the storage waste due to the redundancy of the hash value. To make the best use of the redundancy multi-Payword hash chain, MSRC was presented in [22]. In order to reduce the computation and storage burden, a novel micropayment scheme using a hash chain is proposed. Each value in the hash chain is taken weight by adopting a pre-defined function. Moreover, the weight is used as the denomination of this hash value. Therefore, the denominations of the hash chain are variable. As a result, the proposed scheme with variable denomination is more efficient than the schemes with fixed denomination.

The rest of this paper is organized as follows. In

Section 2, we introduce and analyze the MSRC scheme. Moreover, a variable denomination micropayment is proposed in Section 3. In Section 4, we give the analysis of presented scheme. The comparison of the micropayment schemes is given in Section 5. Finally, Section 6 concludes the paper.

2 Yang-Wu's MSRC

2.1 Micropayment Scheme to Return Changes (MSRC)

The MSRC scheme involves three participants, i.e., $\boldsymbol{C}$, $\boldsymbol{B}$ and $\boldsymbol{V}$, their corresponding public key and private key are (PK_C, SK_C), (PK_V, SK_V) and (PK_B, SK_B). Furthermore, $\{\cdot\}_{PK}$ and $\{\cdot\}_{SK}$ denote the encryption and signature using the public key and private key. The MSRC scheme consists of four phases: *Registration Phase*, *Transaction Phase*, *Redemption Phase and Remittance Phase*.

1) Registration Phase

Step 1. $\boldsymbol{C}$ sends a payment request $PR = \{ID_C, ID_V, n, r\}$ to $\boldsymbol{B}$, where ID_C and ID_D are the identities of $\boldsymbol{C}$ and $\boldsymbol{V}$, n and r are the length of payment hash chain and returning change hash chain, respectively.

Step 2. $\boldsymbol{B}$ generates three hash chains $(A - chain, A^{'} - chain, B - chain)$ with the denomination d_A, $d_A^{'}$ and d_B, which satisfy $d_A = d_A^{'} < d_B$,

$$\begin{aligned} A &= A_1 \| A_2 \\ &= (a_0, a_1, \cdots, a_n) \| (a_{n+1}, a_{n+2}, \cdots, a_{n+r}) \\ A^{'} &= A_1^{'} \| A_2^{'} \\ &= (a_0^{'}, a_1^{'}, \cdots, a_n^{'}) \| (a_{n+1}^{'}, a_{n+2}^{'}, \cdots, a_{n+r}^{'}) \\ B &= (b_0, b_1, \cdots, b_n) \end{aligned}$$

where $a_i = h(a_{i+1})$, $a_i^{'} = h(a_{i+1}^{'})$, for $i = n + r - 1, n + r - 2, \cdots, 0$, and $b_j = h(b_{j+1})$, for $i = n - 1, n - 2, \cdots, 0$, $h(\cdot)$ is a secure hash function such as SHA-256, and $\|$ is the concatenation operation.

Step 3. $\boldsymbol{B}$ sends $\{A, A_1^{'}, B\}_{PK_C}$ to $\boldsymbol{C}$, and sends $\{a_0, a_0^{'}, b_0, A_2^{'}\}_{PK_V}$ to $\boldsymbol{V}$.

Step 4. $\boldsymbol{C}$ derives the payment chain $\{A, A_1^{'}, B\}$ by decrypting $\{A, A^{'}, B\}_{PK_C}$ with SK_C.

Step 5. $\boldsymbol{V}$ obtains the root $\{a_0, a_0^{'}, b_0\}$ and the returning chain $A_2^{'}$ by decrypting $\{a_0, a_0^{'}, b_0, A_2^{'}\}_{PK_V}$ with SK_V.

2) Transaction Phase

$\boldsymbol{C}$ uses the e-cash $A - chain$ and $B - chain$ to pay for the goods. For simplicity, an example which the price of the goods is 27 dollars is used to illustrate this phase.

In the MSRC scheme, d_A = \$1 and d_B = \$10 were assumed. Then $\boldsymbol{C}$ should spend 2 hash values in $B - chain$ and 7 hash values in chain A_1 and $A_1^{'}$. That's to say the hash chain A_1 and $A_1^{'}$ with small denomination will be fetched away soon. When the hash chains A_1 and $A_1^{'}$ are exhausted, $\boldsymbol{C}$ must spend 3 hash values in $B - chain$, and $\boldsymbol{V}$ needs to spend 3 hash values in $A_2^{'}$, which saves the hash values with small denomination. The processes of $\boldsymbol{C}$ pays $M \times d_B$ money and $\boldsymbol{V}$ returns $m \times d_A$ money are shown in the following listing.

Step 1. $\boldsymbol{C}$ sends (b_M, M) to $\boldsymbol{V}$.

Step 2. $\boldsymbol{V}$ checks the equation $b_0 = h^M(b_M)$. If the equation holds, $\boldsymbol{V}$ transmits $(a_{n+m}^{'}, m)$ to $\boldsymbol{C}$ for returning $m \times d_A$ money, provides goods to $\boldsymbol{C}$, and stores b_M instead of b_0.

Step 3. $\boldsymbol{C}$ checks the equation $a_n^{'} = h^m(a_{n+m}^{'})$. If the equation holds, $\boldsymbol{C}$ ensures that $\{a_i, a_i^{'}\}$ $(i = n + 1, n + 2, \cdots, n + m)$ can be used for the payment.

3) Redemption Phase and Remittance Phase

If $\boldsymbol{V}$ wants to redeem the money from $\boldsymbol{B}$, he executes the following steps.

Step 1. $\boldsymbol{V}$ sends $\{ID_C, ID_V, a_{i_1}, a_{i_1}^{'}, b_{j_1}\}_{SK_V}$ to $\boldsymbol{B}$.

Step 2. $\boldsymbol{B}$ verifies $i_1 \leq n + r$ and $j_1 \leq n$. If i_1 and j_1 are valid, $\boldsymbol{B}$ remits the money $i_1 \times d_A + j_1 \times d_B$ to $\boldsymbol{V}$ and stores $(a_{i_1}, a_{i_1}^{'}, b_{j_1})$ simultaneously.

Step 3. In the next time, $\boldsymbol{B}$ receives the redemption request $\{ID_C, ID_V, a_{i_2}, a_{i_2}^{'}, b_{j_2}\}_{SK_V}$ from $\boldsymbol{V}$. Therefore, $\boldsymbol{B}$ remits the money $(i_2 - i_1) \times d_A + (j_2 - j_1) \times d_B$ to $\boldsymbol{V}$ and stores $(a_{i_2}, a_{i_2}^{'}, b_{j_2})$ when $i_2 \leq n + r$ and $j_2 \leq n$ hold.

2.2 Analysis of Yang-Wu's MSRC

With the ability of returning change, MSRC scheme is in favor of fabricating the hash chains with the length of the appropriate ratio, then the hash chain with the shortest length is often found to be used for purchasing the same goods. As the PayWord with the small denomination, $(a_i, a_i^{'})$ is utilized for the payment. Moreover, the chain with returning change cannot be used unless the hash chains with the small denomination are exhausted. Before the chain with returning change is used, the $\boldsymbol{V}$'s verification costs twice hash computations, and $\boldsymbol{C}$ needs to store twice hash chains A and $A^{'}$ with the same length. Therefore, the computation and storage burden should be paid attention to.

In order to reduce the computation and storage burden, a variable denomination micropayment scheme is presented using Hamming weight [2], in which only one hash chain is necessary and the denomination of values in it might not be the same.

3 Variable Denomination Micropayment (VDM) Scheme

In VDM scheme, the range of the transaction values is assumed in the set $D = \{1, 2, \cdots, s\}$, where s is the max transaction value. Moreover, the denomination $d_i = WF(a_i) mod P + 1$ is assigned to a_i, where $WF(x)$ is denoted as the weight function with the value of x such as Hamming weight function, or other method that can be defined as the value of some bits, P is a prime number. Then d_i is an positive integer in the range $[1, P]$. In this paper, Hamming weight is assigned to $WF(x)$, the binary representation of 9 is 1001, so $WF(9) = 2$.

The VDM scheme also consists of four phases: *Registration Phase*, *Transaction Phase*, *Redemption Phase and Remittance Phase*. Moreover, the details of each phase are described as follows.

1) Registration Phase

$\boldsymbol{C}$, $\boldsymbol{V}$ select random pseudonym identities ID_C and ID_V which must be different from that of others. The identities of them should be verified by $\boldsymbol{B}$.

Step 1. $\boldsymbol{C}$ sends a payment request $PR = \{ID_C, ID_V, n\}$ to $\boldsymbol{B}$ for getting a payment hash chain.

Step 2. $\boldsymbol{B}$ generates an $n - length$ hash chains,

$$A = (a_0, a_1, \cdots, a_n)$$

where $a_i = h(a_{i+1})$ for $i = n-1, n-2, \cdots, 0$.

Step 3. $\boldsymbol{B}$ sends $\{A\}_{PK_C}$ to $\boldsymbol{C}$, and sends $\{a_0\}_{PK_V}$ to $\boldsymbol{V}$.

Step 4. $\boldsymbol{C}$ derives the payment hash chain A by using the private key SK_C to decrypt $\{A\}_{PK_C}$.

Step 5. $\boldsymbol{V}$ obtains the root a_0 by decrypting $\{a_0\}_{PK_V}$ with SK_V.

2) Transaction Phase

Step 1. Before $\boldsymbol{C}$ sends the first purchase request, he chooses a_{m_1} which the total denominations satisfy $t_1 = \sum_{j=1}^{m_1} d_j - W_1$ and $|t_1| < P$, where W_1 is the true price of the first purchased goods. Moreover, $\boldsymbol{C}$ sends (a_{m_1}, m_1, t_1) to $\boldsymbol{V}$.

Step 2. $\boldsymbol{V}$ computes $t_1^{'} = \sum_{j=1}^{m_1} d_j - W_1$ and checks $t_1^{'} = t_1$; then checks $a_0 = h^{m_1}(a_{m_1})$ and $|t_1^{'}| < P$, If all checks hold, $\boldsymbol{V}$ provides $\boldsymbol{C}$ with the goods and stores (a_{m_1}, t_1).

Step 3. In the $i-th$ transaction, $\boldsymbol{C}$ selects $a_{m_1^{'}}$ which $t_{i-1}t_i < 0$ (where $t_i = t_{i-1} + \sum_{j=m_{i-1}^{'}+1}^{m_i^{'}} d_j - W_i$ $(i \geq 2)$) and $|t_i| < P$. Then $\boldsymbol{C}$ sends $(a_{m_i^{'}}, m_i^{'} - m_{i-1}^{'}, t_i)$ to $\boldsymbol{V}$, where t_i is the error between the paid money and the true price $\sum_{j=1}^{i} W_j$ of the purchased goods for the previous i times transactions, and $m_i^{'} - m_{i-1}^{'}$ is the number of hash operations in the $i-th$ transaction.

Step 4. $\boldsymbol{V}$ computes $t_i^{'} = t_{i-1}^{'} + \sum_{j=m_{i-1}^{'}+1}^{m_i^{'}} d_j - W_i$, provides $\boldsymbol{C}$ with the goods and stores $(a_{m_i^{'}}, t_i)$ if all of $t_i^{'} = t_i$, $a_{m_{i-1}^{'}} = h^{m_i^{'} - m_{i-1}^{'}}(a_{m_i^{'}})$ and $|t_i^{'}| < P$ are hold.

3) Redemption Phase and Remittance Phase

Step 1. $\boldsymbol{V}$ sends $\{ID_C, ID_V, i_1, a_{i_1}\}_{PK_B}$ to $\boldsymbol{B}$.

Step 2. $\boldsymbol{B}$ derives $\{ID_C, ID_V, i_1, a_{i_1}\}$ by using the private key SK_B. Furthermore, if $i_1 \leq n$ is valid, $\boldsymbol{B}$ recharges the money $Q_{i_1} = \sum_{j=1}^{i_1} d_j$ to $\boldsymbol{V}$ and stores a_{i_1}, where Q_i is the money $\boldsymbol{B}$ remits to $\boldsymbol{V}$ in the $i-th$ transaction.

Step 3. When $\boldsymbol{B}$ receives $\{ID_C, ID_V, i_2, a_{i_2}\}_{PK_B}$ in the next remittance period, where $i_2 \leq n$, then $\boldsymbol{B}$ calculates $Q_{i_2} = \sum_{j=i_1+1}^{i_2} d_j$, deposits the money Q_{i_2} to $\boldsymbol{V}$ and replaces a_{i_1} with a_{i_2}.

4 System Analysis

In this section, we analyze the security and efficiency of the proposed scheme and compare it with other schemes.

4.1 Security Analysis

The proposed micropayment scheme achieves the correctness, anonymity, un-traceability, fairness, and unforgeability.

Correctness.

In the VDM scheme, $|t_i| < P$ is checked by $\boldsymbol{V}$ in each transaction. Furthermore, the error rate between the error value t_i and the total of true price in the previous i times transactions is defined as $E_i = \frac{|t_i|}{\sum_{j=1}^{i} W_j} \times 100\%$. Obviously, E_i will reduce when the transaction amount increases. As a result, t_i is acceptable for $\boldsymbol{V}$ and $\boldsymbol{C}$ since the transaction amount is huge.

The AVISPA tool can analyze the security of the protocols, and check whether the scheme is safe or unsafe against the passive and active adversaries. Moreover, the AVISPA tool is used for the automated validation of security-sensitive protocols and applications. The simulation results are showed in Figure 1. From the results, it is evident that our scheme is secure against attack, and satisfies the confidentiality.

Anonymity and Un-traceability.

In the proposed scheme, $\boldsymbol{C}$ and $\boldsymbol{V}$ randomly select the pseudonym identities ID_C and ID_V , which need not have any relationship with the real identities of them, and must be different from that of any others. Therefore, nobody including $\boldsymbol{C}$, $\boldsymbol{V}$ and $\boldsymbol{B}$ can trace the real identity of the entities via the data exchange. Then the VDM scheme satisfies the anonymous and the un-traceability.

```
% OFM
% Version of 2006/02/13
SUMMARY
 SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
 /home/ubuntu/tools/avispa-1.1/testsuite
  /results/micropayment_MSSC.if
GOAL
 as_specified
BACKEND
 OFMC
COMMENTS
STATISTICS
 parseTime: 0.00s
 searchTime: 9.04s
 visitedNodes: 0 nodes
 depth: 1000000 plies
```

Figure 1: The simulation results of analysis using AVISPA

Fairness.

If the scheme is executed functionally, it is fair for all entities. The scheme adopts the prepaid method in the transaction, $\boldsymbol{C}$ must pay the money to $\boldsymbol{B}$ before sending the payment request. Therefore, the profit of $\boldsymbol{B}$ and $\boldsymbol{V}$ are protected. Moreover, $\boldsymbol{B}$ is responsible for creating the hash chain, so the risk of the $\boldsymbol{C}$'s overspending is eliminated. Furthermore, $\boldsymbol{C}$ pays the money to $\boldsymbol{V}$ after receiving the goods, so he takes no risk of unfairness.

Unforgeability.

Security plays a vital and central role in the e-commerce payment scheme. In this subsection, we analyze that the VDM scheme resists against the outsider and insider forgery attacks, it means that no forgery attacks exist in the proposed scheme.

No forgery attacks mean an attacker cannot fake the value of hash chain for payment by eavesdropping the information from the public channel.

Scenario 1. *It is unfeasible for an attacker to forge the hash chain to exchange money if the protocol is processed functionally.*

Proof. Primarily, the external malicious attacker cannot forge and spend the hash chain since hash chain is a strong cryptography one-way function and the information transferred among $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{V}$ is always encrypted by a public key. Meanwhile, the external malicious attacker obtains the information $(a_{m_i'}, m_i' - m_{i-1}', t_i)$ from the public channel, he cannot derive $(a_{m_i'+1}, a_{m_i'+2}, \cdots, a_n)$ to redeem money from $\boldsymbol{B}$ due to the feature of secure hash function. Moreover, the external attacker cannot obtain a_{i_1} from the eavesdropped $\{ID_C, ID_V, i_1, a_{i_1}\}_{PK_B}$. The external malicious attacker cannot use the identity of $\boldsymbol{V}$ to exchange money for himself, and the money can be remitted if and only if the identity of $\boldsymbol{V}$ can be verified. Therefore, the attacker cannot forge the hash chain for payment or obtain the illegal benefit from $\boldsymbol{B}$.

Secondly, the forgery attack from the internal $\boldsymbol{C}$ means that $\boldsymbol{C}$ attempts to use the hash chains to double spend or to overspend hash value. In the proposed scheme, $\boldsymbol{C}$ cannot spend more than the total money of the hash chain since he should pay the money to $\boldsymbol{B}$ in advance. Simultaneously, $\boldsymbol{B}$ will track the balance of the total money of the hash chain and avoid the overspending hash chain values. Moreover, the used hash chain will be destroyed by $\boldsymbol{B}$ and will not be allowed to double spend. Obviously, $\boldsymbol{C}$ cannot launch this attack.

Finally, the forgery attack from the internal $\boldsymbol{V}$ implies that $\boldsymbol{V}$ sends $\{ID_C, ID_V, i_1, a_{i_1}\}_{PK_B}$ to $\boldsymbol{B}$ for exchanging more money. If $i_1 \leq n$, $\boldsymbol{V}$ can redeem money Q_{i_1} from $\boldsymbol{B}$, but he cannot get a_i $(i > i_1)$. Therefore, it is impossible for the internal $\boldsymbol{V}$ to redeem more money from $\boldsymbol{B}$. If $\boldsymbol{V}$ receives a_i $(i \leq n)$, he cannot obtain a_j $(j > n)$ to redeem money. Hence, he cannot use the received a_i to redeem the illegal money from $\boldsymbol{B}$. □

4.2 Efficiency Analysis

The $\boldsymbol{C}$, $\boldsymbol{V}$ and $\boldsymbol{B}$ involved in the proposed VDM only needed their (PK_C, SK_C), (PK_V, SK_V) and (PK_B, SK_B). In the registration phase, the private key computation SK_C is executed once by $\boldsymbol{C}$ and the private key computation SK_V is carried out once by $\boldsymbol{V}$. Moreover, $\boldsymbol{B}$ executes the corresponding public key computation PK_C and PK_V once. However, the private key computation SK_B is executed once by $\boldsymbol{B}$ and the corresponding public key computation PK_B is used once by $\boldsymbol{V}$ in each redemption phase and remittance phase. No certificates are needed in the transaction. Moreover, the information stored by the entities is much more simple and entities can be implemented the protocol as soon as possible. Compared with the other schemes which used the complicated algorithm, the proposed scheme is highly efficient.

5 Comparison

In this subsection, some important micropayment schemes (MSSC, MSMC, MSRC and VDM) are further discussed: (1) the number of hash and encryption operation, (2) the storage burden. The comparison of the computation and storage burden among the MSSC, the MSMC, the MSRC and the proposed schemes will be summarized. Furthermore, only the hash function computation is considered since the complicated computations such as the signature and encryption operations are the same in these schemes. In addition, $d = \frac{d_B}{d_A}$ is defined due to the two chains with different denomination d_A and d_B, where $d_A < d_B$ in all micropayment schemes.

The computation cost in MSRC scheme is approximate

with that of MSMC scheme. Afterwards, the computation cost in VDM scheme is much smaller than that of the above schemes. Therefore, the computation costs of MSSC, MSRC schemes are compared with that of our VDM scheme.

The error rate E_i is controlled within 5%, and the hash operation amount of previous s transactions in MSSC, MSRC schemes are compared with that of the VDM scheme, which is shown in Figure 2.

As depicted in Figure 2, the computation cost in VDM scheme is significant smaller than that of others. We compare the storage burden requirement of other schemes with VDM scheme in Table 1. Obviously, our storage burden is also lighter than that of the other schemes.

6 Conclusions

By using the Hamming weight function in micropayment scheme, the VDM scheme is presented to reduce the hash operations in the transaction phases. Each hash value in VDM scheme is taken weight by adopting the Hamming weight function. Moreover, the weight is used as the denomination of this hash value. Therefore, the denominations of the hash chain are variable. Compared with the fixed denomination schemes, the proposed scheme not only ensures the privacy, anonymity, un-traceability, fairness, unforgeability and authentication, but also is more efficient with respect to the computation and storage burden. Therefore, the presented scheme is suitable for the secure mobile devices communication.

Acknowledgments

Project supported by the Natural Science Foundation of China (No. 61363069, 61301166), Guangxi Natural Science Foundation (No. 2014GXNSFAA118364), Innovation Project of Guangxi Graduate Education (No. YCSZ2013069), the High Level Innovation Team of Guangxi Colleges and Universities, and the Program for Innovative Research Team of Guilin University of Electronic Technology.

References

[1] V. Daza and F. Lombardi, "Frodo: Fraud resilient device for off-line micropayment," *IEEE Transactions on Dependable and Secure Computing*, DOI: 10.1109/TDSC.2015.2432813.

[2] C. Ding and J. Yang, "Hamming weights in irreducible cyclic codes," *Discrete Mathematics*, vol. 313, no. 4, pp. 434–446, 2013.

[3] A. Esmaeeli and M. Shajari, "Mvpayword: Secure and efficient payword-based micropayment scheme," in *Applications of Digital Information and Web Technologies*, pp. 609–614, 2009.

[4] C. Fan, Y. Liang, and C. Wu, "An anonymous fair offline micropayment scheme," in *International Conference on Information Society*, pp. 377–381, 2011.

[5] T. Fun, L. Beng, and M. Razali, "Review of mobile macro-payment schemes," *Journal of Advances in Computer Networks*, vol. 1, pp. 323–328, 2013.

[6] A. Isern-Dey, M. Payeras-Capell, and M. Mut-Puigserver, "Micropayment proposal with formal verification using coloured petri nets and performance analysis on the android platform," *International Journal of Business Intelligence and Data Mining*, vol. 8, no. 1, pp. 74–104, 2013.

[7] N. Kiran and G. Kumar, "Implication of secure micropayment system using process oriented structural design by hash chaining in mobile network," *International Journal of Computer Science Issues*, vol. 9, pp. 329–339, 2012.

[8] Y. Liu, L. Hu, and H. Liu, "A micropayment scheme based on weighted multi-dimensional hash chain," *Journal of Electronics (China)*, vol. 23, pp. 791–794, 2006.

[9] Y. Liu and J. Yan, "A lightweight micropayment scheme based on lagrange interpolation formula," *Security and Communication Networks*, vol. 6, pp. 995–960, 2013.

[10] J. Lo, H. Lu, T. Sun, and M. Hwang, "Improved on date attachable electronic cash," *Applied Mechanics and Materials*, vol. 284-287, pp. 3444–3448, 2013.

[11] J. Meng and Y. Yang, "Self-renewal hash chains micropayment protocol based on payword," *Computer Engineering*, vol. 35, pp. 63–65, 2009.

[12] Y. Mu, V. Varadharajan, and Y. Lin, "New micropayment schemes based on paywords," *Lecture Notes in Computer Science*, vol. 1270, pp. 283–293, Springer, 1997.

[13] S. Quan, "Multi-dimensional hash chains and application to micropayment schemes," *Lecture Notes in Computer Science*, vol. 3969, pp. 218–228, Springer, 2006.

[14] R. Rivest and A. Shamir, "Payword and micromint: Two simple micropayment schemes," *Lecture Notes in Computer Science*, vol. 1189, pp. 69–87, Springer, 1997.

[15] L. Rotger, M. Capell, and M. Puigserver, "Anonymous, fair and untraceable micropayment scheme: Application to LBS," *IEEE Latin America Transactions*, vol. 10, pp. 1774–1784, 2012.

[16] K. R. Santosh, C Narasimham, and P Shetty, "Cryptanalysis of multi-prime rsa with two decryption exponents," *International Journal of Electronics and Information Engineering*, vol. 4, no. 1, pp. 40–44, 2016.

[17] G Sharma, S Bala, and A. Verma, "An improved rsa-based certificateless signature scheme for wireless sensor networks," *International Journal of Network Security*, vol. 18, no. 1, pp. 82–89, 2016.

[18] M. Silvio and L. Ronald, "Micropayment revisited," *Lecture Notes in Computer Science*, vol. 2271, pp. 149–163, Springer, 2002.

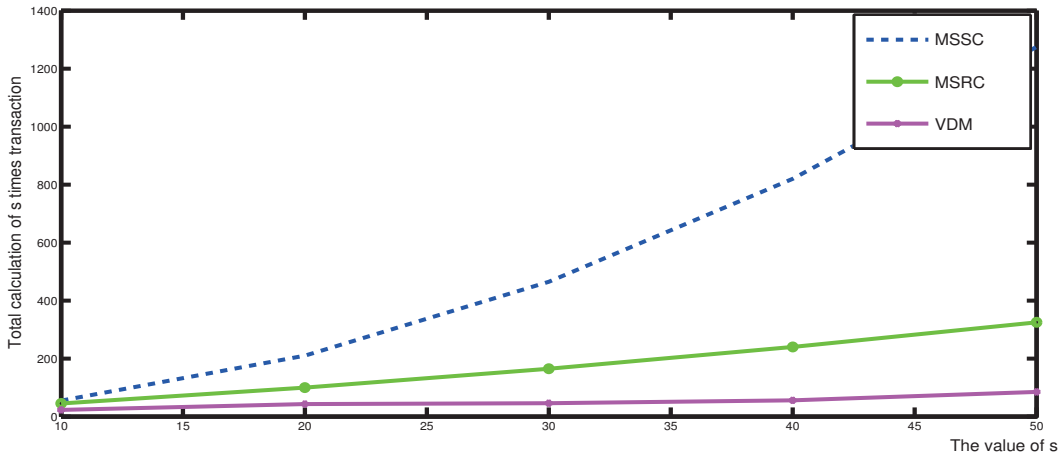

Figure 2: Total hash operation comparison of s times transaction

Table 1: Storage burden comparison of related works with ours

	MSSC	**MSMC**	**MSRC-I**	**MSRC-II**	**MSRC-III**	**VDM**
Encryption key	No	No	$\boldsymbol{C}$ need	No	$\boldsymbol{C}$ and $\boldsymbol{V}$ need	No
$\boldsymbol{C}$*'s storage burden*	$(d+1)n$	$2n$	$2n$	$3n+r$	$2n+1$	$\frac{22n}{1+P}$
$\boldsymbol{V}$*'s storage burden*	1	2	$r+2$	$r+3$	$\geq r+2$	2
Total storage burden	$(d+1)n+1$	$2n+2$	$2n+r+2$	$3n+2r+3$	$\geq 2n+r+3$	$\frac{22n}{1+P}+2$

[19] H. Wang, J. Ma, and J. Sun, "Micro-payment protocol based on multiple hash chains," in *Second International Symposium on Electronic Commerce and Security*, pp. 71–74, 2009.

[20] H. Wang, Z. Wang, and Y. Bo, "Micropayment system based on self-updatable two-dimensional hash chain," *Computer Engineering*, vol. 37, no. 18, pp. 272–274, 2011.

[21] C. Yang and H. Teng, "An efficient method for finding minimum hash chain of multi-payword chains in micropayment," in *IEEE International Conference on E-Commerce*, pp. 45–48, 2003.

[22] C. Yang and C. Wu, "Msrc: Micropayment scheme with ability to return changes," *Mathematical and Computer Modelling*, vol. 58, no. 1-2, pp. 96–107, 2013.

[23] S. Yen, H. Lin, Y. Chen, J. Hung, and J. Wu, "Paystar: A denomination flexible micropayment scheme," *Information Sciences*, vol. 259, pp. 160–169, 2014.

Quan-yu Zhao is currently pursuing his M.S. degree in Guilin University of Electronic Technology, Guilin, China. He received the B.S. degree in Mathematics and Applied Mathematics from Huainan Normal University, Anhui, China, in 2014. His research interests focus on e-voting, micropayment, e-lottery, group key transfer, and oblivious transfer.

Yi-Ning Liu is currently a professor in Guilin University of Electronic Technology, Guilin, China. He received the B.S. degree in Applied Mathematics from Information Engineering University, Zhengzhou, China, in 1995, the M.S. in Computer Software and Theory from Huazhong University of Science and Technology, Wuhan, China, in 2003, and the Ph.D. degree in Mathematics from Hubei University, Wuhan, China, in 2007. His research interests include the analysis of information security protocol, the smart grid, and e-voting.

Gao Liu is now pursuing his M.S. degree in Guilin University of Electronic Technology, Guilin, China. He received the B.S. degree in Applied Mathematics from Yibin University, Sichuan, China, in 2013. His research interests focus on e-voting, micropayment, e-lottery, and smart grid.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His first degree is Bachelor of Science in Applied Mathematics and master degree is Master of Science in computer and decision sciences. Both were awarded in National Tsing Hua University. Dr. Chang served in National Chung Cheng University from 1989 to 2005. His title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from Feb. 2005. He is a Fellow of IEEE and a Fellow of IEE, UK. His research interests include database design, computer cryptography, image compression and data structures.