A Study of Micro-payment Based on One-Way Hash Chain

Min-Shiang Hwang[1] and Pei-Chen Sung[2]

(Corresponding author: Min-Shiang Hwang)

Department of Management Information Systems, National Chung Hsing University[1],
250 Kuo Kuang Rd., Taichung, Taiwan 402, R.O.C. (Email: cc.yang@nchu.edu.tw)
Department of International Business Management, WuFeng Institute of Technology[2],
NO.117, Sec. 2, Chian-Kuo Rd., Ming-Hsiung, Chiayi County, Taiwan 621-53, R.O.C.

(Invited Paper)

Abstract

Electronic payment systems have gradually become an important issue nowadays because of the popularity and importance of electronic commerce on the Internet. Security and convenience related topics are the most important issues that concern people. The electronic micropayment is one of the most popular research topics on electronic commerce. Recently, many efficient micropayment schemes, based on the usage of one-way hash chain, were developed. They enable more and more new applications for e-commerce. However, all existing micropayment schemes suffer a common drawback that a generated chain of electronic coins can only be spent at a specific merchant. This drawback limits the widespread application of existing micro-payment schemes. In this thesis, we introduce several micro-payment schemes based on one-way hash chain and review some literatures on supporting multiple payment. We propose a new micropayment scheme which achieve the following three goals: micro-payment multiple transactions, service providers, and anonymity.

Keywords: Blind signature, elliptic curve cryptosystem, micro-payment, one-way hash chain

1 Introduction

1.1 Electronic Payment

Electronic commerce has been rapidly expanding over the past decade. Customers can now purchase goods and services over the Internet. In an ideal electronic commerce, all of the steps of a transaction could be performed over the network. Information could be intercepted and tampered easily in an open network. Hence, how to build a secure and efficient environment for electronic payment is a key issue in electronic commerce development.

Many electronic payment systems have been developed. In general, electronic payment systems are classified into two types: macro-payment and micro-payment. The main difference between them is the amount of payment.

1.2 Difference Between Macro-payment and Micro-payment Schemes

Of the two types of payments, macro-payment schemes transfer larger sums of money for each transaction, so the security requirement is usually more rigorous. To date, public key cryptosystem is commonly used in macropayment for authentication and encryption. Besides public key cryptosystem, macro-payment schemes use on-line broker activities to detect double spending prior to the acceptance of a payment by the vendor. Both of the vendor's broker and the customer's broker connect to verify the transaction amount and perform on-line verification and redemption.

The computational and storage costs of micro-payment schemes are suitable for small payments. For example, purchasing a web page or downloading a paper. Compared with macro-payment scheme, the computational times of micro-payment schemes are less because it uses the one-way, collision-resistant hashing extensively but not public key cryptosystem. As a rough estimate, hashing is approximately about 100 times faster than the RSA signature verification, and about 10,000 times faster than the RSA signature generation [22]. Besides hashing, micro-payment schemes generally avoid on-line verification by the broker. This saves the broker on-line processing time and on-line storage requirements.

The security of micro-payment schemes is apparently not as efficient as that of the macro-payment schemes. However, if a micro-payment scheme is designed so that a customer only loses a few cents when his transaction is tampered with, and the cost of counterfeiting a coin is either computations or policies are higher than the value of the coin, then the security is considered to be adequate.

In short, the classification of both macro-payment and micro-payment schemes is based on processing time and storage requirements [19]. While macro-payment schemes are more concerned with the authenticity and privacy of data and therefore needs demanding encryption algorithms and on-line processing. Micro-payment schemes aim at providing a decent level of security for transactions with more economical time and storage requirement.

1.3 Micro-payment

A micro-payment scheme is an electronic payment system designed to allow efficient frequent payments of small amounts (e.g., less than one dollar or a few cents). In order to be efficient and keep the transaction cost very low, micro-payments minimize the communication and computation used. In contrast to macro-payment, micro-payment schemes aim to allow offline payment verification using lightweight cryptosystems. The systems do not require high transaction security, in order to increase efficiency. The cost of fraud is made more expensive than the possible value to be gained by cheating.

However, some security requirements are essential, such as authentication of the customer and merchant, protecting the integrity of transaction messages, and gaining non-repudiation of transaction processes. Due to these properties, a good micro-payment system not only performs the transactions accurately but meets the following requirements:

1) Good efficiency:
   The payment actions must be managed quickly and information goods delivered online.

2) Low cost:
   All the following loads should be minimized:

   a. Computational load:
      This cost should be comparable with the value to be paid. Therefore, the employment of public key cryptography, e.g., signature scheme, should be prevented or at least be kept as seldom as possible.

   b. Storage load:
      Since there will be a large amount of payment to be handled, it is not feasible to keep a record of each payment. This probably will make the cost of processing the payment by overloading it.

   c. Administrative load:
      This includes the minimization of interactions with the trusted third party (usually the bank) and the frequency of doing withdrawals and deposits.

3) Security:
   The identity of the users (customers or merchants) and the integrity of the transaction messages must be authenticated and protected.

4) Multiple transactions and service providers:
   A client should be able to do a number of micro-payment transactions with several different service providers on a single day.

5) Anonymity is necessary:
   People use money to purchase goods or a service and the merchant or bank/broker cannot associate their identity in real life. Digital cash should be the same as real money.

In order to satisfy these requirements, micro-payment systems often use efficient cryptographic techniques to ensure the security of transaction. The one-way hash function is a useful technique.

The most notable representatives of micro-payment schemes include those proposed in [1, 7, 9, 13, 16, 22, 23]. The fundamental cryptographic tool for most of these payment systems is a one-way hash chain which produces a one-way hash function. One-way hash chains have been extensively employed in the development of a special class of high-speed signature schemes called the one-time signature schemes [8, 17].

1.4 Hash Chain

Public key digital signature schemes have been widely used in electronic payment schemes. However, public key schemes are computationally expensive. Therefore, it may not be practical to request clients to sign each payment with a public key signature scheme.

Many notable representatives of micro-payment schemes make use of hash functions which generate a one-way hash chain which has been recognised widely by researchers ever since Lamport first proposed its use in one-time passwords [17]. They can be done much faster compared to public key schemes.

When the function $h(\cdot)$ in the iteration is instantaneous with a one-way hash function, such as MD5 and SHA, the result is a one-way hash chain as shows in Figure 1. Each element x_i is computed as $h^{n-r}x_n$.

This paper is organized as follows. In the next section, we briefly introduce what is electronic payment, and explain the difference between macro-payment and micro-payment. We also make a description of hash chains. In Section 2, we introduce the related works on micro-payment schemes. In Section 3, we present the proposed micro-payment scheme. In Section 4, we make an analysis of the proposed scheme. Finally, in Section 5, our conclusions and future works.

$$\boxed{W_0 = h^n(W_n) \leftarrow W_1 = h^{n-1}(W_n) \leftarrow W_2 = h^{n-2}(W_n) \leftarrow \ldots \leftarrow W_{n-1} = h^1(W_n) \leftarrow W_n}$$

Figure 1: One-way hash chain

2 Background Knowledge

2.1 Micro-payment

All micro-payment mechanisms consist of least three parties: the customer (or client), the merchant (or vendor) and a third party called the broker. The relationship of the three parties is shown in Figure 2.

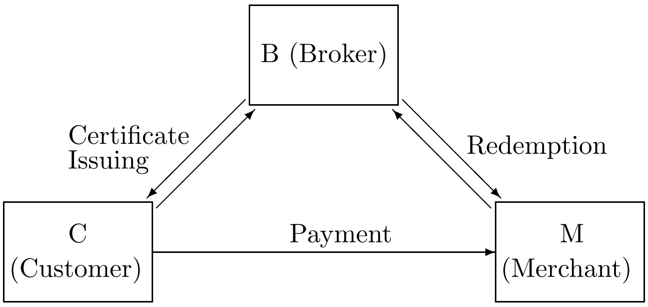

Figure 2: The relationship of the customer, the merchant and the broker

- The customer: who requests a certificate to the broker and makes purchases in the merchant.
- The merchant: who provides customers with information and contents according to their payments.
- The broker: who issues the certificate to customers and acts as a CA (Certificate Authority).

There are two ways of generating money for micro-payment schemes.

Money is created or certified by a broker:

A customer is assumed to buy micro-payment's money in bulk from a broker through a macro-payment protocol, and the broker debits the customer's account. To both the customer and the broker (whoever creates or certifies the money), this is a debit-based approach because the customer has to purchase a specific form of money in advance; whoever certifies the money will benefit from the "float". A return policy is required for a customer to refund or renew unused, expired money.

Money is generated by a customer:

The money generated by a customer may not need direct certification by a broker. In this case, no bulk purchase or macro-payment scheme is required. The payment scheme is credit-based to a customer, merchant, and broker because the customer's account will not be debited until the account will be paid at redemption.

Millicent was proposed in Digital Equipment Corporation [6]. It is a debit-based protocol to the customer, merchant, and broker. In this protocol, the payment message is called Scrip. To verify the Scrip, it is efficient in using symmetric encryption. However, the Scrip is merchant specific. First, customers must purchase the broker Scrip from the broker by using macro-payment protocol. When customers want to purchase something at a certain merchant site, customers must take the broker Scrip to change the specific merchant Scrip from the broker. The merchant Scrip can only be used at specific merchant site.

The PayWord and MicroMint were designed by Ronald Rivest and Adi Shamir [22]. PayWord is a credit-based protocol for the customer, merchant, and broker. It is based on a chain of hash values, called payword. Each payword can be represented as a specific value and has the same value in the same chain. Customers obtain a certificate issued by the broker. The certificate allows the legal customers to generate paywords. To verify the payword, the vendor only uses hash function and a signed commitment to honor payments of that chain. MicroMint uses k-way hash function collisions to mint coins. Although the security of MicroMint coins is lower, its performance is efficient. The customer generates coins efficiently and the merchant verifies these coins off-line.

Small Value Payment (SVP) is designed by Sern and Vaudenay [25] It uses a message authentication code (MAC) function and a special device, such as smart card, to verify the validity of payment message. Each merchant has a smart card which is issued by the broker. The smart card stores the broker's secret key to verify the identity of the customer. However, this protocol needs many communications between the customer and the merchant. The smart card must generate a random number which is sent to the customer, and wait for a response from the customer in each transaction. This is not suitable for frequent transaction, because other transactions must wait till the processing transaction is completed.

In [4], an experimental portable micro-payment system based on PayWord [22] has been reported. From the discussions given in the paper, it becomes clear for a general purposed portable device, while a small or moderately large value of n (called the length of the payment chain) would be acceptable, a larger n can cause an unacceptable lengthy delay in computation. On the other hand, a larger value of n reduces the required amount of computation for public key based signature which is actually the essence of developing PayWord-like micro-payment schemes.

To solve the above mentioned problem, some research workers develop efficient structures different from a simple one-way hash chain for good micro-payment schemes, especially for those to be implemented on a portable computing device.

Jutla and Yung proposed a structure called PayTree [7]. A PayTree offers a solution for a multi-merchant environment. That is to say, a PayTree can be spent among many different merchants. But there are two drawbacks when we use Paytree in practice. The first drawback is that the customer needs to store all the leaf nodes of a Paytree. These leaf nodes represent electronic coins bought by the customer from a bank. In a practical application, as the number of leaf nodes could be large, the customer may need to prepare a large amount of memory space to store all the node values. The second drawback is that double spending of a coin in the PayTree scheme cannot be avoided, although it can be detected afterwards. The reason is that the customer could pay the same coin to many different merchants.

Yen et al. proposed a new tree-based structure called an unbalanced one-way binary tree (UOBT) [28]. The major difference between UOBT and PayTree is that UOBT promotes merchant specific micro-payments in the conventional one-way hash chain structure. In a scheme based on UOBT, a secret random value is chosen as the root. This secret value is used to construct a tree from the root towards the lower levels in an unbalanced binary tree, such that to give a child node, no parent node can be derived from the child node. In [28], it was demonstrated that the UOBT approach could improve the performance of micro-payment schemes significantly.

UOBT is a 2-dimension one-way hash function chain that employs two different one-way hash functions. Lin et al. proposed a general micro-payment scheme based on an n-dimensions one-way hash function chain in 2002 [18]. In this scheme, the user can determine the trade off and choose the right number of dimensions for her-/himself.

2.2 Elliptic Curve Cryptosystem (ECC)

Elliptic curves have been extensively studied for over a hundred years [16]. Since the introduction of public key cryptographic systems by Whitfield Diffie and Martin Hellman [5], numerous public-key cryptographic systems have been proposed [16, 27]. All of these systems depend on the difficulty of mathematical problem for their security: integer factorization problem or discrete logarithm problem. In 1985, Lenstra succeeded in using elliptic curves for integer factorization. This result suggested the possibility of applying elliptic curves to public key cryptosystems [24]. Miller and Koblitz were the first to propose cryptosystems that employed elliptic curves. They didn't devise new cryptographic algorithms but they implemented existing public key cryptosystems using elliptic curves [15, 20], whose security rests on the discrete logarithm problem over the points on an elliptic curve. The greatest advantage of ECC is that its keys are smaller than those of the existing public key schemes with the same criterion of security and can be applied to the smart cards with restricted computational power and memory [30].

An elliptic curve is the set of solutions (x, y) to an equation for two numbers a, and b of the form:

$$y^2 = x^3 + ax + b \bmod p.$$

If (x, y) satisfies the above equation then P = (x, y) is a point on the elliptic curve.

An elliptic curve can also be defined over the finite field consisting of 2^m elements. Suppose P and Q are both points on the curve, then $P + Q$ will always be another point on the curve. The security of the ECC rests on the difficulty of the elliptic curve discrete logarithm problem. It can be stated as follows. P is a prime and a point on the curve. xPrepresents the point P added to itself x times. Suppose Q is a multiple of P for x namely $Q = xP$. The elliptic curve discrete logarithm problem is to determine x given P and Q.

2.3 Blind Signature

Another important technique is the blind signature. Blind signature is a kind of digital signature. Unlike a normal digital signature scheme, in a blind signature scheme, a signer signs a message without knowing what the message contains. That is, the message is blinded by a requester. After receiving the signed message from the signer, the requester can derive the valid signature of the message from the signer. Anyone can verify the blind signature using the public key of the signer. If the message and its signature are published, the signer can verify the signature, but he/she cannot link the message-signature pair [11]. Because of these two properties: blindness and untraceability, blind signatures are widely used in many e-commerce services, (e.g. electronic voting schemes and electronic payment systems).

The concept of the first blind signature scheme was introduced by Chaum [3]. This scheme was based on the factoring logarithm and the security depended on the RSA assumption. Camenisch et al. presented the blind signature based on the discrete logarithm problem [2]. In order to improve the efficiency of the blind signature, Fan et al. proposed a new scheme which was based on the difficulty of solving the square roots of quadratic residues [6]. In this study, we present a new blind signature scheme based on elliptic curve cryptography (ECC) [12, 15, 20, 26]. The security of ECC is based on the elliptic curve discrete logarithm problem (ECDLP) and was proven to provide greater efficiency than the factorization and discrete logarithm systems used by Vanstone [27].

We propose a new blind signature which is based on ECC. Notations in this article are listed as follows.
X_s: private key of the signer
Q_s: public key of the signer
k: randomly chosen number by the signer u
v: randomly chosen number by the requester

m: message which the requester wants to blind
$H(\cdot)$: A collision-free hash function

The procedure of the proposed scheme is shown in Figure 3 and as follows:

1) The requester obtains R' from the signer. That is, $R' = kP$.

2) The requester calculates $R = uR' + vP$, $e = H(R||m)$, and sends $e' = \frac{e}{u}$ to the signer.

3) The signer calculates $S' = X_s e' + k$ and sends it to the requester.

4) Upon receiving S', the requester calculates $S = S'u + v$ and checks the following equation:

$$SP = eQ_s + R.$$

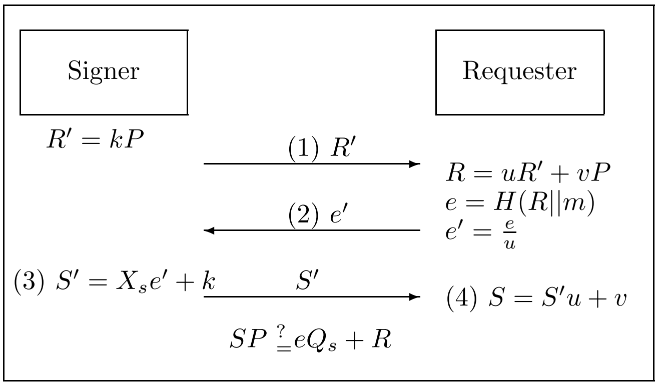

Figure 3: The blind signature which is based on ECC

If this verification is successful, then the requester gets a valid signature.

We can apply the blind signature technique when the user (U) withdraws a coin from the bank (B). The procedures is as follows:

1) In order to ask a withdrawal of coin c to B, U sends the requests to B.

2) Later, B selects a random number k, computes R', and sends R' to U. After receiving R', U computes R and e, using secret random value u and v. Then, U calculates the blinded value e and sends to B.

$$\begin{aligned} R' &= kP \\ R &= uR' + vP \\ e &= H(R||c) \\ e' &= \frac{e}{u} \end{aligned}$$

3) B uses his private key to generate blind signature S' for e' and sends it to U.

$$S' = X_s e' + k$$

4) U unblind B's signature S' by using u and v, and verifies S by checking the equation: $\mathrm{SP} = \mathrm{eQ}_s + \mathrm{R}$.

$$\mathrm{S} = \mathrm{S'u} + \mathrm{v}$$

5) If the equation is even, then it is accepted.

After obtaining S, U can pay the withdrawn coin to the merchant.

3 Proposed Scheme

After reviewing literatures about PayWord based micropayment, we try to propose a new micro-payment scheme. There are two drawbacks in general PayWord based micro-payment scheme. The first drawback is overspending prevention problem. The second drawback is multiple payments problem.

In PayWord system, customer C has to spend paywords to a specific vendor. In order to solve this problem, the broker creates new hash chain values that enable a user to make payments with multiple vendors in Kim et al.'s scheme [14]. The new chain is generated by hashing w_i and s_i where s_i is based on a shared user-broker secret. Unlike a normal chain, the user signs a commitment to the chain root and releases each following w_i as the payment. Since the final hash w_n is never fixed, the chain can be extended indefinitely by continuing to generate further w_i values. However, because of the s_i is secret, the vendor is unable to verify any of the paywords offline. The vendor must trust the user to send valid paywords. Indeed, even if the user cheats, the vendor cannot later prove this later.

In the case of solving this problem, we adopt a payment root which is created by the customer. The merchant can verify all paywords offline which cannot be denied by the user later.

In addition, we propose a new blind signature which is based on ECC and is introduced in Section 2. We use blind signature which is based on ECC when the customer asks for a withdrawal request. It will be a satisfactory anonymous requirement.

Our proposed scheme is debit-based. We can divide our proposed scheme into four phases: registration phase, blinding phase, transaction phase, and redemption phase. The participants within the proposed scheme consist of customer (C), merchant (M) and broker (B).

In the registration phase, C and M establish relations with B. B creates secret key to C and M. C and M must use these keys to carry out the business transactions. In the Blinding phase, C sends a withdrawal request to B. C creates a hash chain and B signs it using B's secret key. Then B sends the blinded signature back to C. In the transaction phase, C decides to ask for service from M and sends a transaction request to M. In the redemption phase, M makes a redemption request to B. Finally; B transfers the amount of the payment to M's account.

The notations used in the proposed scheme are listed in Table 1:

Table 1: Notations

B	The bank
C	The customer
M	The merchant
ID_B	The ID of the bank
ID_C	The pseudonymous identity of the customer in the transaction
ID_M	The ID of the merchant
P	A generator point in ECC
X_B	The private key of the bank
Q_s	The public key of the bank
k, r_B	The randomly chosen number by the bank
u, v, r_C	The randomly chosen number by the customer
A_M	The Internet host address of the merchant
K_{CB}	A secret key shared by customer and bank
K_{MB}	A secret key shared by merchant and bank
K_{CM}	A one-time session key shared by customer and merchant. It is created by B
I_C	The individual information of C
OI	Order information. Such as category, amount and total value which C want to ask for
$\{M\}_{Kx}y$	Message is encrypted by the secret key
$H(\cdot)$	A hash function such as MD5 or SHA
$H^r(W_n)$	A hash function H is applied r times to an argument W_n iteratively

3.1 Registration Phase

Both C and M have accounts with B and B is trusted by the other entities. Each C and M shares a secret key K_{CB} and K_{MB} with B respectively. C selects a pseudonymous identity ID_C which is unique to every customer. B and C share a secret key K_{CB}. Equally, M has an account with B and they share a secret key K_{CB}. ID_M is the real identification information of M.

$$(ID_C, ID_B, Expiry)_{K_{CB}}$$

Where *Expire* denotes the date on which the hash chain is invalid. It can limit the length of time both M and B need to store information about the state of a hash chain.

3.2 Blinding Phase

Before C asks for service from M, C sends a withdrawal request to B as follows:

Step 1: C sends $\{ID_C, I_C\}$ to B. After checking the identity of C, B sends R' to C. That is, $R' = kP$.

Step 2: After receiving R', C selects a random number W_N and creates a hash chain $W_N, W_{N-1}, \ldots, W_1$, W_0. W_0 is the root of the hash chain and each element of the hash chain satisfies $W_i = H(W_{i+1})$, where $i = N-1, N-2, \ldots 1, 0$. N is the limited amount that B allows C to spend. Then, C calculates $R = uR' + vP$, $e = H(R||W_0)$, $e' = \frac{e}{u}$, and sends $\{e', N\}_{K_CB}$ to B.

Step 3: If N is smaller than the limited amount that B allows C to spend, B calculates $S' = X_s e' + k$ and sends it to C. Otherwise, B rejects C's request.

Step 4: Upon receiving S', C calculates $S = S'u + v$ and checks the following equation:

$$SP = eQ_s + R$$

If this verification is successful, then C gets a valid signature. The pair (R, S) is the signature issued by B in the blinded message e.

Step 5: B creates two special and significant factors: T_C and S_C. We define T_C and S_C as the following.

$$\begin{aligned} T_C &= h(C, r_B) \\ S_C &= \{s_i | s_i = h(s_i + 1, T_C), i = N-1,,0\} \end{aligned}$$

T_C is used to make clear that the new hash values generated by a broker is issued to whoever since no one except the broker can create it. r_B is a random number which is chosen by B. S_C is the new hash chain values that enable a customer to make payments with multiple merchants and should be generated with T_C.

3.3 Transaction Phase

After browsing the M's web site, C decides to ask for service from M.

Step 1: C sends transaction request $\{A_M, ID_C, ID_B\ \}_{K_{CB}}$ to B.

Step 2: B maintains a table of each C's ID_C and knows the secret key $K_C B$. Therefore B can decrypt the request and check C's authenticity. If C passes the verification, B creates a one-time session key K_{CM} for C and M. Then B sends $\{K_{CM}\}_{K_{CB}}$ to C.

Step 3: C asks for service from M. After receiving K_{CM}, C calculates $R_{CM} = H(W_j \oplus (s_k \parallel K_{CM}))$, and sends $\{R_{CM}, (R, e, S), W_0, (W_j, k), s_k, OI, Expire\}_{K_{CM}}$ to M. Where $k = j - i + 1$. For example, assume that C pays the paywords $W_1, W_2, \ldots, W_6$ to the first merchant M_1, i.e., the payment $P_1 = (W_6, 6)$ is sent to the merchant M_1. Then, C will pay the paywords W_7, W_8, W_9 to the second merchant M_2, i.e., the payment $p_2 = (W_9, 3)$ is sent to M_2.

Step 4: M verifies the blind signature and R_{CM} as in the following:

$$\begin{array}{rcl} SP & = & eQ_s + R \\ W_{n-1} & = & H(W_n), wheren = j-1, j-2, K, 1 \\ H^{k-1}(W_j) & = & H^{k-2}(W_{j-1}) = \ldots = H(W_{j-k+2}) \\ & = & W_{j-k+1} \\ R'_{CM} & = & H(W_{j-k+1} \oplus (s_k \parallel K_{CM})) \end{array}$$

Step 5: If the above equations hold, M can start selling electronic items or services to C.

3.4 Redemption Phase

M should carry out redemption process with B after a period of time.

Step 1: M sends request for redemption to B as follows:

$$\{R_{CM}, (R, e, S), W_0, (W_j, k), s_k, OI, Expire\}_{K_{MB}}$$

The redemption messages contain the blind signature, order information, payment root, the expiry date of the hash chain and the payment (W_j, k) given by C in the transaction phase.

Step 2: B checks the date of validity, and verifies the blind signature. Then verifies each paywords (W_j, k). This process is the same as in step 4 of the transaction phase. Finally, B extracts the payment from C's account and transfers the amount to M's account.

4 Analysis

4.1 Security

4.1.1 Blindness

The signer signs a message without knowing its contents. Blindness is the first important property in a blind signature. In our scheme, the requester calculates $R = uR' + vP$, and generates e' which is a concatenation of R and m with a hash function $H(\cdot)$. Then, he/she sends them to the signer. Hence, the signer cannot know message m.

4.1.2 Unforgeability

No one can forge (m, R, S) because the elliptic curve discrete logarithm problem is difficult to solve. We assume three situations as follows.

Situation 1: If someone tried to fake R_1, m_1, he/she cannot obtain S_1. Because $S_1 P = e_1 Q_s + R_1$ and S_1 is unknown. It is an elliptic curve discrete logarithm problem and difficult to solve.

Situation 2: If someone gets S_1, m_1, he/she cannot obtain R_1. Because $S_1 P = e_1 Q_s + R_1$, R_1 is unknown, and $e_1 = H(R_1 \parallel m_1)$. It is also an elliptic curve discrete logarithm problem and difficult to solve.

Situation 3: If someone tries to fake R_1 and S_1, he/she cannot obtain m_1. Because $S_1 P = e_1 Q_s + R_1$, he/she cannot get e_1 without m_1. It is an elliptic curve discrete logarithm problem and is a problem to solve.

4.1.3 Untraceability

If anyone obtains the valid signature, he/she cannot link this signature to the message. In our scheme, if the signer keep a record set (k_i, R'_i, e'_i, S'_i), where $i = 1, 2, \ldots, n$, he/she cannot trace the blind signature. We expand this as follows.

When the requester reveals n records (m_i, R_i, S_i) to the public, the signer will compute the values e_i and u', and obtain S_i and R_i, where $e_i = H(Ri \parallel mi)$, and $u' = \frac{e_i}{e'_i}$. However, the signer cannot trace the blind signature by detecting whether each R_i and $R_i + 1$ have the same relation. Therefore, the signer cannot trace the blind signature.

4.2 Double Spending Detection

Before doing business with M, C sends $\{R_{CM}, (R, e, S), W_0, (W_j, k), s_k, OI, Expire\}_{K_{CM}}$ to M. The payment root R_{CM} is equal to $H(w_j \oplus (s_k \parallel K_{CM}))$. Every time when C makes a purchase, the s_k, K_{CM} are not the same. Hence, B can detect double spent paywords if C expends double the paywords that they have already.

4.3 Forgery Prevention

Before a transaction, C sends a request to B to make a blind signature on W_0 using B's private key. No one can achieve it only a proper C who can create paywords. Besides, in order to process a correct redemption, M must have knowledge of the payment information. It is impossible for someone to find out even one of them, because only if he could know the secret key K_{CM}, K_{MB} to forge it.

4.4 Overspending Prevention

In blinding phase, C sends $\{e', N\}_{K_{CM}}$ to B. Where N is the limited amount that B allows C to spend. If N is

Table 2: The summary of the computation and communication cost for micro-payment systems

	PayWord	Wang et al.	Kim et al.	Proposed Scheme
Public Key Signature				
C	1	1	1	1
M	2	2	2	0
B	1	1	1	1
Symmetric Key Encryption				
C	0	0	0	4
M	0	0	0	1
B	0	0	0	5
Hash Function				
C	n	N+1	N+1	N+2
M	n	n+1	0	n+1
B	n	n	n+1	n+1
Network Connection				
C	0	0	1	1
M	1	1	1	1
B	0	0	0	0

smaller than the limited amount that B allows C to spend, B calculates $S' = X_s e' + k$ and sends it to C. Otherwise, B rejects C's request. For this reason, it is impossible for C to create over an amount which B allows C to spend.

4.5 Multiple Payment

In the transaction phase, C sends a request to B to gets K_{CM} and creates the payment root $R_{CM} = H(W_j \oplus (s_k \parallel K_{CM}))$ where W_i is the first unused payword in the payword sequence, s_k, K_{CM} are created by B. Hence, every time when C makes a purchase, the R_{CM} is not the same. C enables to make payments with multiple merchants.

We compare our scheme with other micro-payment schemes in this literature. Table 2 shows the performance of the computation and communication of each transaction. Table 3 shows comparisons of the public key signature, symmetric key encryption, hashing function and network connection, that can e performed number per second on a typical workstation [10]. The symmetric key encryption cryptography and hashing function are more efficient. They are suitable for micro-payments. PayWord, Wang et al. and Kim et al.'s schemes use a public key signature to generate and verify a certificate, which is not efficient. Our scheme uses a public key in blinding phase to blind the message (money) that anyone can verify the blinding signature but cannot link the message-signature pair. That is to say, people use digital cash to purchase goods or a service on the Internet which is the same as in the real life.

5 Conclusions and Future Work

5.1 Conclusions

In this study, we describe the requirements of micro-payment and reviewe the related works and literatures about micro-payment supporting multiple payments based on Payword. Furthermore, we have proposed a new micro-payment scheme in Section 3. The advantages of this scheme are described as follows:

1) Our proposed scheme is based on a one-way hash chain. The fundamental goals of a micro-payment system design are efficient and low cost. The one-way hash function is a simple and efficient technique that is suitable for a micro-payment.

2) On the user's side, system security and speed are very important points. In our proposed scheme, we use blind signatures to ensure that the paywords are untraceable and the user's private information is concealed. The ECC is more efficient than RSA and DSA [27].

3) Only two secret keys K_{CB} and K_{CM} are needed between C, B and M in this scheme. No certificate is required.

Due to these features, the proposed scheme is suitable for micro-payments for information goods on the Internet in real life.

5.2 Future Works

A micro-payment system is used to buy information goods or service over the computer network. The important factors in such a payment are small amounts of payment

Table 3: The comparisons for the computation and network connection speed [10]

Operation	Number per second
Public Key Signature (1024 bits RSA)	2
Symmetric Key Encryption (DES)	2,000
One-way Hashing Function (MD5/SHA)	20,000
Network Connection (TCP/Internet)	1,000

Table 4: The summary of the comparison for micro-payment systems

	PayWord	PayTree	Wang et al.	Kim et al.	Proposed Scheme
Anonymity	×	×	×	×	○
Double spending detection	○	×	○	○	○
Forgery prevention	○	○	○	○	○
Non-repudiation	○	○	○	○	○
Overspending prevention	×	○	×	○	○
Multiple payments	×	○	○	○	○

value and high frequency of transactions on the electronic commerce network.

Mobile telephony is a growing market all over the world. The Internet revolution is advancing rapidly and commercial interests abound today. Electronic commerce is a new business circumstance on the open network especially in a wireless network environment. People nowadays like to carry their mobile phones or PDAs with them. It is very convenient for them to connect to the Internet anytime and anywhere. As users buy mobile Internet service from multiple merchants, they hope to continue their original communication without interruption. However, the home mobile Internet service provider to which mobile user might not be the current provider from which the mobile user can purchase services. Mobile users can buy Mobile Internet Services from multiple Mobile Internet Services provider more security, efficiency and seamlessly is very important nowadays. It is a new challenge at present.

References

[1] R. Anderson, C. Manifavas and C. Sutherland, "NetCard-A practical electronic cash system," in *Proc. of Security Protocols Workshop*, Lecture Notes in Computer Science, LNCS 1189, Springer Verlag, pp.49-57, 1997.

[2] J. Camenisch, J. Piveteau, and M. Stadler, "Blind signatures based on discrete logarithm problem," in *Advances in Cryptology, EUROCRYPT'94*, Lecture Notes in Computer Science, LNCS 950, Springer Verlag, pp. 428-432, 1994.

[3] D. Chaum, "Blind signatures for untraceable payments," in *Advances in Cryptology, CRYPTO'82*, pp. 199-203, 1982.

[4] N. Daswani and D. Boneh, "Experimenting with electronic commerce on the PalmPilot," in *Proc. of 3th Financial Cryptography Conference, FC'99*, Lecture Notes in Computer Science, LNCS 1648, Springer Verlag, February 1999.

[5] W. Diffie and M. Hellman, "New directions in cryptography", *IEEE Transactions in Information Theory*, vol. IT-22, pp. 644-654, November 1976.

[6] C. I. Fan and C. L. Lei, "Efficient blind signature scheme based on quadratic rsidues," *IEE Electronic Letters*, pp. 811-813, 1996.

[7] S. Glassmann, M. Manasse, M. Abadi, P. Gauthier, and P. Sobalvarro, "The Millicent protocol for inexpensive electronic commerce," in *Proc. of 4th International World Wide Web Conference*, Boston, MA, pp. 603-618, Dec. 1995.

[8] N. M. Haller, "The S/KEY one-time password system," in *Proc. of the ISOC Symposium on Network and Distributed System Security*, San Diego, CA, Feb. 1994.

[9] R. Hauser, M. Steiner, and M. Waidner, "Micropayments based on iKP," in *Proc. of SECURICOM'96, 14th Worldwide Congress on Computer and Communications Security and Protection*, pp.67-82, 1996.

[10] M. S. Hwang, I. C. Lin, and L. H. LI, "A simple Micro-payment scheme", *The Journal of Systems and Software* vol. 55, pp. 221-229, 2001.

[11] M. S. Hwang, C. C. Lee, and Y. C. Lai, "An untraceable blind signature scheme", *IEICE Transactions on Foundations*, vol. E86-A, no. 7, pp. 1902-1906, 2003.

[12] M. S. Hwang, S. F. Tzeng, and C. S. Tsai, "Generalization of proxy signature based on elliptic curves", *Computer Standards & Interface*s, vol. 26, no. 2, pp. 73-84, 2004.

[13] C. S Jutla and M. Yung, "PayTree: Amortized-signature for flexible micro-payments," in *Proc. of*

Second USENIX Association Workshop on Electronic Commerce, pp.213-221, November 1996.

[14] S. Kim and W. Lee, "A PayWrod-based Micropayment protocol supporting multiple payments," in Computer Communications and Networks, 2003. ICCCN 2003. Proceedings. The 12th International Conference on, pp.609-612, 20-22 Oct. 2003.

[15] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, pp. 203-209, 1987.

[16] N. Koblitz, A. Menezes, and S Vanstone, "The state of elliptic curve cryptography," *Designs, Codes and Cryptography*, vol. 19, pp. 173-193, 2000.

[17] L. Lamport, "Password authentication with insecure communication," *Commun. of ACM*, vol. 24, no. 11, pp. 770-772, 1981.

[18] I. C. Lin, M. S. Hwang, and C. C. Chang, "The general Pay-Word: A Micro-payment scheme based on one-way hash functions," *Designs, Codes and Cryptography*, 2002.

[19] *Micro Payment Transfer Protocol (MPTP) Version 0.1*, W3C Working Draft 22-Nov-95, http://www.w3.org/pub/WWW/TR/WD-mptp

[20] V. Miller, "Use of elliptic curves in cryptography," in *Advances in Cryptology, CRYPTO'85*, pp. 417-426, 1986.

[21] T. Pedersen, "Electronic payments of small amounts," in *Proc. of Security Protocols Workshop*, Lecture Notes in Computer Science, LNCS 1189, Springer Verlag, pp.59-68, 1997.

[22] R. L. Rivest and A. Shamir, "PayWord and MicroMint: Two simple micro-payment schemes," in *Proc. of Security Protocols Workshop*, Lecture Notes in Computer Science, LNCS 1189, Springer Verlag, pp.69-87, 1997. Also in CryptoBytes, Pressed by RSA Laboratories, vol.2, no.1, pp.7-11, 1996.

[23] R. L. Rivest, "Electronic lottery tickets as micropayments," in *Proc. of Financial Cryptography Conference, FC'97*, Lecture Notes in Computer Science, LNCS 1318, Springer Verlag, pp.307-314, 1998.

[24] M. Saeki, *Elliptic curve cryptosystems*, School of Computer Science McGill University, Montreal, Feburary 1997, http://www.cs.mcgill.ca/ crepeau/PDF/memoire-mugino.pdf.

[25] J. Stern, S. Vaudenay, "SVP: A flexible micropayment scheme," in *Proc. Financial Cryptography Workshop*, LNCS, Springer Verlag, 1997.

[26] S. F. Tzeng and M. S. Hwang, "Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem," *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 61-71, 2004.

[27] S. A. Vanstone, "Elliptic curve cryptosystem- the answer to strong, fast public-key cryptography or securing constrained environments," *Information Security Technical Report*, vol. 2, no. 2, pp. 78-87, 1997.

[28] S. M. Yen, L. T. Ho, and C. Y. Huang, "Internet micro-payment based on unbalanced one-way binary tree," in *Proc. Of International Workshop on Cryptographic Techniques and E-Commerce, CrypTEC'99*, Hong Kong, pp. 155-162.

Min-Shiang Hwang was born on August 27, 1960 in Tainan, Taiwan, Republic of China (ROC.). He received the B.S. in Electronic Engineering from the National Taipei Institute of Technology, Taipei, Taiwan, ROC, in 1980; the M.S. in Industrial Engineering from the National Tsing Hua University, Taiwan, in 1988; and a Ph.D. in Computer and Information Science from the National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathematics at the National Cheng Kung University, Taiwan, from 1984-1986. Dr. Hwang passed the National Higher Examination in field "Electronic Engineer" in 1988. He also passed the National Telecommunication Special Examination in field "Information Engineering", qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also the chairman of the Department of Information Management, Chaoyang University of Technology (CYUT), Taiwan, during 1999-2002. He was a professor and chairman of the Graduate Institute of Networking and Communications, CYUT, during 2002-2003. He obtained the 1997, 1998, 1999, 2000, and 2001 Outstanding Research Award of National Science Council of the Republic of China. He is currently a professor and chairman of the department of Management Information Systems, National Chung Hsing University, Taiwan, ROC. He is a member of IEEE, ACM, and Chinese Information Security Association. His current research interests include electronic commerce, database and data security, cryptography, image compression, and mobile computing. Dr. Hwang has published over 100 articles on the above research fields in international journals.

Pei-Chen Sung was born on June 26, 1976 in Taichung , Taiwan, Republic of Chain (ROC.). She received the B.C. degree in Finance and M.S. degree in Information Management from Chaoyang University of Technology, Taichung, Taiwan, ROC, in 2002 and 2005. She is currently an assistant of the college of Social Science and Physical Education, National Changhua University of Education, Taiwan, ROC. She is a part-time lecturer concurrently of WuFen Institute of Technology, Taiwan, ROC. Her current research interests include electronic commerce and computer networks.