Comments on Micropayment Schemes for Multi-Merchants with Anonymity and Untraceability

Shin–Jia Hwang
Department of Computer Science and Information Engineering,
Tamkang University, Tamsui, New Taipei City, 251, Taiwan, R.O.C.
E-mail: sjhwang@mail.tku.edu.tw

Hung-Zhi Tsai
Department of Computer Science and Information Engineering,
Tamkang University, Tamsui, New Taipei City, 251, Taiwan, R.O.C.
E-mail: 600410962@s00.tku.edu.tw

Abstract

Hwang and Sung proposed the micropayment scheme for multiple merchants with anonymity and untraceability. Then Bayyapu and Das proposed their scheme by improving the performance. However, those schemes still suffers the traceability by untrustworthy bankers and double spending caused by the cooperation attack performed by malicious merchants. Moreover, the performance of Bayyapu and Das's scheme is not better than the one of Hwang and Sung's scheme.

Keywords*: Micropayment, blind signatures, double spending, anonymity, traceability*

1. Introduction

Micro-payment is an electronic payment system major for the transaction with very small amount money. For the consideration of efficiency and cost, the one-way payword chains for small amount payment are created by using one-way hash functions [7].

In [7], the payword chains should be merchant-specific to prevent double spending problem. If some customer wants to transact with different merchants, he/she should generate multiple payword chains for different merchants, respectively. This way is inconvenient for the customers. Therefore, some improved micro-payment schemes are proposed such that one payword chain can be used to transact with multiple merchants [2, 5, 6].

To protect the transaction privacy, customers need to be anonymous and untraceable during the transaction with the merchants. To provide anonymity and untraceability, the blind signature scheme [3] may be adopted. By adopting the blind signature scheme over finite elliptic curve fields, Hwang and Sung's micropayment scheme is proposed to provide multiple merchants, anonymity, and untraceability properties. Then, by adopting the RSA blind signature scheme, Bayyapu and Das proposed their scheme to improve the performance of Hwang and Sung's scheme. Because oonly the blind signature schemes are different, those two schemes are almost the same.

Though those two schemes adopt the blind signature scheme, they cannot provide anonymity and untraceability to protect customers' privacy. Moreover, the double spending is still a serious flaw for those two schemes. In the next section, Hwang and Sung's scheme is reviewed first and our comments are also given. In Section 3, Bayyapu and Das's scheme is reviewed. Our comment showing that Bayyapu and Das's scheme is not more efficient than Hwang and Sung's scheme is given in the same section. The last section is our conclusion.

2. Our Comments on Hwang and Sung's Scheme

2.1 Hwang and Sung Scheme

Hwang and Sung [5] proposed their multiple micro-payments by using one-way hash functions and blind signature scheme based on the elliptic curve cryptosystem (ECC for short). To review their scheme, some notations should be defined in the following table.

Table 1: Notations in Hwang and Sung's Scheme

B	Bank
C	Customer
M	Merchant
ID_x	ID of x ,where $x \in \{B,C,M\}$
I	Customer individual information
k_x	Private key of x,where $x \in \{B,C,M\}$
P_x	Public key of x,where $x \in \{B,C,M\}$
$\{M\}_K$	The ciphertext generated by encrypting on the message M with the secret key K
OI	Order information

Hwang and Sung scheme consists of four phases. Those phases are described, respectively.

Registration Phase

First of all, the bank B generates the public elliptic curves EC and the element P on the curves. Then B selects a secret key k. A public key hash function h is also published. Both C and M have to register with the bank B, in this phase. Then C and M share the secret keys K_{CB} and K_{MB} with B, respectively. C also selects a pseudonymous ID_C, which is unique to every customer.

Blinding Phase

In this phase, the customer C obtains an authorized one-hash chain from the bank B, by performing the following protocol.

Step 1: C sends $\{ID_C, I\}$ to B. Then B validates ID_c, and computes and sends $R'=kP$ to C.

Step 2: C selects a random number W_n and creates a hash chain W_n, W_{n-1}, …, W_1, W_0 after receiving R', where $W_i= h(W_{i+1})$, for i= n-1, n-2, …, 1, 0. Here the number n is the limited amount that B allows C could spend each time. Then C computes $R= uR'+vP$, $m= h(R||w_0)$, and $m' = m/u$, where u and v are the random numbers chosen by C. C finally sends $\{m', n\}_{K_{CB}}$ to B.

Step 3: After obtaining {m', n} by decrypting $\{m', n\}_{K_{CB}}$, B first checks where or not n is smaller than the limited C could spend and then computes $S' = k_B m' + k$. Finally B sends S' to C.

Step 4: C computes $S = S'u + v$ and checks whether or not $SP = mP_B + R$. If the equation holds, C gets the valid signature (R, S) on message m.

Step 5: B also creates two factors: $T_c= h(ID_c, r_B)$ and $S_c= \{s_i|s_i = h(s_{i+1}, T_c), i = N-1,…, 0\}$, where r_B is the random number chosen by B and N is the maximal number of transaction that C can do on the business.

Transaction Phase

After obtain anonymously authorized hash chain and the signature (R, S), the customer C can transact with some merchants. Suppose that C wants to transact with some merchant M, C, M, and B perform the following procedure to complete the kth transaction.

Step 1: C sends the transaction request $\{A_{M,} ID_C, ID_B\}_{K_{CB}}$ to B, where A_M denotes the Internet address of the merchant.

Step 2: B uses each shared secret key to decrypt $\{A_{M,} ID_C, ID_B\}_{K_{CB}}$ until the decrypted ID_C matches the identity of actual owner's the decryption key K_{CB}. If ID_C is authenticated and valid, the bank B randomly chooses a one-time session key K_{CM}, and sends $\{K_{CM}\}_{K_{CB}}$ to C.

Step 3: C first obtains the K_{CM} by decrypting $\{K_{CM}\}_{K_{CB}}$. C computes $R_{CM}=$

$h(w_i \oplus (s_k||K_{CM}))$ and sends $\{R_{CM}$, (R, S, m), w_0, (w_j, t), s_k, OI, Exp$\}_{K_{CM}}$ to M, where t= j-i+1.

Step 4: M verifies the blind signature (R, S) on the message m by using SP = mP_B+ R. If (R, S) is valid, M checks whether or not R_{CM}= $h(w_{j-t+1} \oplus (s_k||K_{CM}))$ and w_0= $h^i(w_i)$, where w_{j-t+1}= $h^{t-1}(w_j)$. If R_{CM}= $h(w_{j-t+1} \oplus (s_k ||K_{CM}))$ and w_0= $h^i(w_i)$ hold, M starts to selling items to C.

Redemption Phase

Suppose that the merchant M wants to redeem the received payment $\{R_{CM}$, (R, S, m), w_0, (w_j , t), s_k, OI, Exp$\}$, M and B perform the following procedure to finish the redemption.

Step 1: M sends the redemption $\{R_{CM}$, (R, S, m), w_0, (w_j , t), s_k, OI, Exp$\}_{K_{MB}}$ to B.

Step 2: B checks the redemption's validity date, verifies the blind signature (R, S) on m, and validates the payword (w_j, t). This step is the same as Step 4 of the procedure in Transaction phase. Finally, B extracts the money from C′s account and transfers it to M′s account.

2.2 Our Comments

Two security flaws of Hung and Sung's scheme are stated. The first flaw is that Hwang and Sung's scheme does not satisfy untraceability property. On Step 1 in the transaction phase, the customer C sends the merchant's address to the bank to distribute the session key K_{CM} between C and M. By using the session key K_{CM}, Bank know the transaction detail between the customer C and merchant M. In other word, B can trace the transaction behavior of any customer, so Hwang and Sung's scheme does not satisfy untraceability property. To remove this traceability flaw, a trusted key distribution center maybe involved to distribute the session key K_{CM}.

The second flaw is the double spending problem, caused by merchants' conspiracy. For example the customer transacted with two different merchants M_1 and M_2. The customer pays M_1 the paywords w_1, w_2, w_3,…, w_6, so the payment P_1= $(w_6, 6)$ is sends to M_1. Then Customer pays M_2 the paywords w_7, w_8, w_9, so the payment P_2= $(w_9, 3)$ is sends to M_2. M_2 knows w_7, w_8, w_9, so M_2 can give w_7 to M_1. After obtaining w_7, M_1 replaces the old payment P_1 with the new payment P_1'= $(w_7, 7)$ and submitted the payment P_1' in the redemption phase. The bank detects the double spending paywords of the customer. However, the bank cannot distinguish this double spending is caused by the customers' double use of the payword or merchants' conspiracy. This dispute of the double spending flaw cannot be solved. Therefore, Hwang and Sung's scheme suffers from the traceability and double spending flaws.

3. Our Comments on Bayyapu and Das's Scheme

3.1 Bayyapu and Das's Scheme

Bayyapu and Das [6] improve the computational efficiency of Hung and Sung scheme by replacing the ECC-based blind signature scheme with the RSA-based blind signature scheme [3, 4]. In the following, only the blinding phase of Bayyapu and Das's scheme is described since the other phases are the same as the corresponding phases in Hung and Sung's scheme.

Blinding Phase

The customer C sends a withdrawal request to the bank B and gets the signature on the withdrawal message. The bank and the customers perform the following protocol.

Step 1: The bank B chooses two large primes p and q, computes λ= pq and $\phi(\lambda)$= (p-1)(q-1), and selects the public key P_B such that 1< P_B <$\phi(\lambda)$ and gcd(P_B, $\phi(\lambda)$)=

1. Finally, find its private key k_B such that $P_B k_B \equiv 1 \bmod \phi(\lambda)$.

Step 2: C first constructs the hash chain w_n, w_{n-1}, …, w_1, w_0, where $w_i = h(w_{i+1})$, for i= n-1, n-2, …, 1, 0. Then C chooses two random numbers u and v, computes $\alpha = u^{P_B} h(w_0)(v^2+1) \bmod \lambda$, and sends (A, α) to B, where A denotes the message specifying the customer's cash expiry date, the total value of each hash word, and the upper limit of the customer account on B.

Step 3: B chooses a random factor $x < \lambda$ and sends x to C.

Step 4: After receiving x, C selects a random number u′ and computes $b = uu' \bmod \lambda$, $\beta = b^{P_B}(v-x) \bmod \lambda$. Send β to B.

Step 5: After receiving β, B computes $\beta^{-1} \bmod \lambda$ and $t = h(A)^{k_B}(\alpha(x^2+1)\beta^{-2})^{2\,k_B} \bmod \lambda$, and sends (β^{-1}, t) to C.

Step 6: After receiving (β^{-1}, t), C computes $c = (vx+1)\beta^{-1}b^{P_B} \bmod \lambda = (vx+1)(v-x)^{-1} \bmod \lambda$, $s = tu^2 \times (u')^4 \bmod \lambda$, where (A, c, s) is the signature issued by B on the blinded messages A and w_0.

The blind signature (c, s) on A and w_0 can be verified by checking the equation:

$$s^{P_B} \equiv h(A)h(w_0)^2(c^2+1)^2 \bmod \lambda.$$

Therefore, on Step 4 in Transaction phase and on Step 2 in Redemption phase, the verification of the blind signatures adopts this equaiton $s^{P_B} \equiv h(A)h(w_0)^2(c^2+1)^2 \bmod \lambda$.

3.2 Our Comments

Bayyapu and Das's scheme is also the same as Hwang and Sung's scheme, except the different underlying blind signature schemes. Therefore, those two schemes suffer from the traceability and double spending flaws.

Moreover, Bayyapu and Das do not improve the computation performance of Hwang and Sung's scheme. The performance comparison [6] between Hwang and Sung's and Bayyapu and Das scheme is illustrated in Table 2. Some notations defined in [6] are defined below.

t_h : The computation time for hash operation

t_{ECCa}: The computation time for one point addition over the elliptic curve.

t_{ECCe}: The computation time for one scalar multiplication of a point over elliptic curve.

t_a: The computation time for one modular multiplication.

t_e: The computation time for one modular exponentiation.

t_s: The computation time for symmetric key encryption.

Table 2: Blind Phase Performance Comparison between Hwang and Sung's and Bayyapu and Das Schemes

	Blinding Phase
Hwang and Sung scheme	$4t_h + 7t_{ECCe} + 3t_{ECCa} + t_s$
Bayyapu and Das scheme	$2t_h + 6t_e$

However, the computational cost of one t_{ECCe} is much cheaper than the cost of one t_e and the computational cost of one t_{ECCa} is much cheaper than the cost of one t_a, even though both the computational complexities of t_{ECCe} in ECC and t_e in RSA are all cubic in the bit length of the module used [1]. According to Table 3, for the same security level, the bit length of the module in RSA signature scheme is much longer than the bit length of the module in the ECC signature scheme. For example, the same security level in 80 bit, RSA needs 1024 bits and ECC needs 160 bits. So the bit length of modules in RSA is approximately six times than the bit length of the module in ECC. Therefore, the cost of $7t_{ECCe} + 3t_{ECCa}$ is much cheaper than the cost of $6t_e$. Consequently, Bayyapu and Das scheme is not more efficient than Hwang and Sung scheme.

Table 3:

Algorithm Family	Cryptosystems	Security Level (bit)			
		80	128	192	256
Integer factorization	RSA	1024 bits	3072 bits	7680 bits	15360 bits
Elliptic curves	ECDH, ECDSA	160 bits	256 bits	384 bits	512 bits

4. Conclusions

Two security flaws for Hwang and Sung's scheme are pointed out. One is the traceability flaw and one is the double spending problem, caused by merchants' conspiracy. Because Bayyapu and Das's scheme is also the same as Hwang and Sung's scheme, their scheme also suffers those two security flaws. Moreover, due to our performance analysis, Bayyapu and Das's scheme does not improve the computational performance of Hwang and Sung's scheme.

References

[1] Christof Paar and Jan Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners*, 1st Ed., 2nd Printing, New York: Springer, 2010, pp. 149-171.

[2] C. Wang, C. Chang and C. Lin, "A New Micro-payment System Using General Payword Chain," *Electronic Commerce Research Journal*, vol. 2, no. 1-2, pp. 159-168, 2002.

[3] D. Chaum, "Blind Signatures for Untraceable Payments," *Advances in Cryptology: Proceedings of Crypto 1982, New York: Springer*, 1983 pp. 199-203.

[4] H. Y. Chien, J. K. Jan and Y. M Tseng, "RSA-based Partially Blind Signature with Low Computation," *Proceedings of the Eighth International Conference in Parallel and Distributed Systems*, Kyongju City, South Korea, Jun. 29, 2001, pp. 385-389.

[5] M. S. Hwang and P. C. Sung, "A Study of Micro-payment Based on One-Way Hash Chain," *International Journal of Network Security*, vol. 2 no.2, pp. 81-90, 2006

[6] P. R. Bayyapu and M. L. Das, "An Improved and Efficient Micro-Payment Scheme," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 4, no. 1, pp. 91-100, 2009.

[7] R. Rivest and A. Shamir, "PayWord and MicroMint: Two Simple Micropayment Schemes," *Proceedings 1996 International Workshop on Security Protocols*, LNCS 1189, New York: Springer, 1996, pp. 69-87.