

Cryptanalysis and Improvement of A New Electronic Traveler's Check Scheme Based on One-way Hash Function

*Jue-Sam Chou ¹, hsien-ching Chen ², Chun-Yun Chen ³

¹Department of Information Management, Nanhua University Chiayi 622 Taiwan, R.O.C

*: corresponding author

jschou@mail.nhu.edu.tw

Tel: 886+ (0)5+272-1001 ext.56536

²Department of Information Management, Nanhua University Chiayi 622 Taiwan, R.O.C

cwa0723@yahoo.com.tw

³Department of Information Management, Nanhua University Chiayi 622 Taiwan, R.O.C

c121350366max@gmail.com

Abstract

Recently, Liaw et al. proposed a hash based electronic traveler's check system. They claimed that their scheme is secure. However, after analyses, we found that their scheme is vulnerable to key compromise impersonation **and parallel session** attack. Further, we will improve their scheme to avoid such an attack.

Keywords: Electronic payment, electronic check, one-way hash function, on-line, off-line, KCI attack, long-term secure key

1. Introduction

Due to the rapid development of electronic commerce. Electronic payment schemes are studied intensively recently. In such schemes, the payer/payee can use bank payment instrument (credit card, debit card, or even current account) without revealing any confidential data during the payment [14, 15, 16]. Generally, an electronic payment system [10, 11, 12] can be divided into three types: on-line credit card payment, electronic cash (e-cash), and electronic check (e-check) which are three extensions of credit card, cash, and check in the real life correspondingly. In fact, cash and check are two frequently used tools and a traveler's check can be used as cash in the real world. Hence, a traveler's check should have the same characteristics as both cash and check do. Similarly, an electronic traveler's check [6, 7] must also include both the characteristics of electronic cash [8, 9] and electronic check [5, 8, 9] in the Internet. If an electronic traveler's check owner loses his check, the designed scheme should be responsible for the possible loss. Many such studies [1, 4, 6, 7, 9, 11, 12, 14] had been proposed. They all have the needed security requirements of an electronic traveler's check system, For example, each entity in the payment system trusts only his bank and the transactions always go under a trusted node. The other needed security requirements of an electronic traveler's check are listed as follows[1].

- (1) No forgery : The electronic traveler's check should prevent malicious users or merchants from forging it and ensure the fairness for users, merchants, and the bank in the transaction environment.
- (2) No double spending : If an electronic check has been used twice, the bank is able to find out who the malicious user or merchant is during the payment phase and deposit phase.
- (3) Specific user : An electronic traveler's check must be signed by both the bank and the user. It should include the identification information regarding both the bank and the user and only the specific owner of the electronic traveler's check should be able to use it.
- (4) Reissuing : When an electronic traveler's check is lost, the user should use his serial number and his payer's endorsement to report the loss. Then, the bank can easily reissue a new one for the user.
- (5) Anonymity : The owner of the electronic traveler's check should be anonymous; that is, the merchant must not know the real identity of the user throughout the whole transaction process.

In 2001, Hsien et al. [7] proposed an electronic traveler's check system based on discrete logarithm problem. Subsequently, some other studies in this aspect [1, 4, 6] that use exponential operations and one-way hash functions are proposed. In 2007, Liaw et al. proposed a new electronic traveler's check scheme based on one-way hash function [1]. They claimed that their scheme is secure against forgery attack since when given a hash value, it is computationally infeasible for an attacker to find an input having the same hash value under a secure one-way hash function. However, we found that their method can not resist the KCI and parallel session attack. KCI attack defined by Wilson and Menezes [2] means that if a user A's long-term secret key is compromised by an adversary, the adversary can pretend other entities to communicate with A. In parallel session attack [17], two or more runs of protocol are executed concurrently under attacker's orchestration. The concurrent runs make the answer to a difficult question in one run available to the attacker so that he can use the answer in another run. In Liaw et al.'s scheme, if a customer's long-term secret key has been leaked, the attacker can impersonate other entity to communicate with him. We think this is caused by the improper design in the registration phase. In this paper, we will improve their scheme to prevent this kind of attack.

The rest of this paper is organized as follows. In Section 2, we review Liaw et al.'s protocol [1]. In Section 3, we show our attack on their scheme and then show our improvement in Sections 4. Finally, a conclusion is given in Section 5.

2. Review of Liaw et al.'s scheme

Liaw et al.'s electronic traveler's check contains four roles: the consumer, the bank, the merchant, and the clearing-house. It uses $X \rightarrow Y:Z$ to denote that sender X sends message Z to receiver Y , and includes two subprotocols: an on-line subprotocol and an off-line subprotocol. In their scheme, the customer only needs one-time registration to purchase an electronic traveler's check and would get an anonymous identity. Further, an optional equipment, like the Smart Card, can be applied in the scheme. In the following, we review both of their on-line and off-line subprotocols in Section 2.1 and 2.2 which are also shown in figure 1 and 2, respectively. (the more details can be referred to [1]).

2.1 The on-line subprotocol

The on-line scheme requires the bank and the clearing-house to be involved in the payment phase. The clearing-house verifies the identity of the user and checks whether double spending occurs. We describe the on-line subprotocol as follows and also is show it in Figure 1.

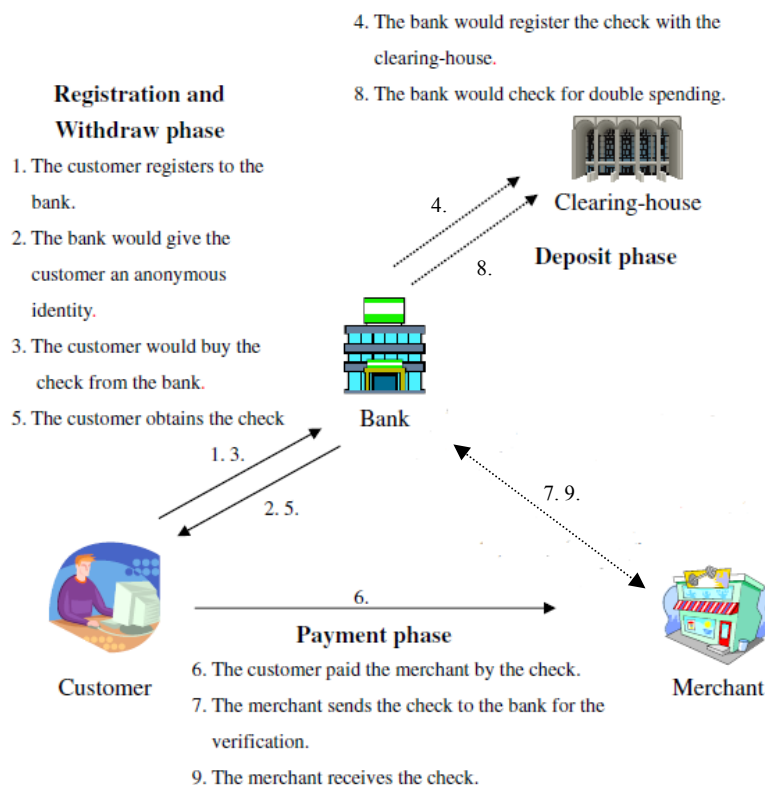


Figure 1: The steps of the on-line subprotocol (source: [1])

(a) The registration phase

In this phase, the customer submits his real identity to the bank for registration and obtains the right to buy an electronic traveler's check. He only needs one-time registration and can buy many electronic traveler's checks at anytime without

registering again. The registration phase includes following three steps:

Step 1. The customer chooses a random number K_{A1} to perform the exclusive OR operation together with his real identity ID_A . Then, he computes the partial anonymous identity $R_{ID_A} = h(ID_A \oplus K_{A1})$ by using a one-way hash function h , encrypts R_{ID_A} by the bank's public key Y_B , and sends the result $PE_{Y_B}(R_{ID_A})$ to the bank.

Step 2. The bank decrypts the received information, obtaining R_{ID_A} . It then chooses a random number K_{B1} and computes $B_{ID_A} = h(R_{ID_A} \oplus K_{B1})$. After that, it encrypts B_{ID_A} by using the customer's public key Y_A and sends the result $PE_{Y_A}(B_{ID_A})$ to him.

Step 3. The customer decrypts the received information by using his private key X_A to get the anonymous identity B_{ID_A} .

(b) The withdraw phase

In this phase, the customer takes the anonymous identity formed in the registration phase to buy an electronic traveler's check from the bank. The withdrawal phase includes following five steps

Step 1. The customer chooses a random number K_{A2} and computes a payer's endorsement $R_A = h(ID_A \oplus K_{A1})$. He then encrypts R_A by using his anonymous identity B_{ID_A} and sends the result $E_{B_{ID_A}}(R_A)$ to the bank.

Step 2. The bank decrypts the received information by computing $D_{B_{ID_A}}(E_{B_{ID_A}}(R_A))$. Then, it chooses another random number K_{B2} and uses its private key X_B to compute its identity $R_B = h(X_B \oplus K_{B2})$. The bank then generates the payee's endorsement R by computing $R = R_A \oplus R_B$, and encrypts R, R_B , and timestamp T_1 by using symmetric key B_{ID_A} . After this, it sends the result $E_{B_{ID_A}}(R, R_B, T_1)$ to the customer.

Step 3. The customer decrypts the received information $E_{B_{ID_A}}(R, R_B, T_1)$, obtaining R, R_B, T_1 . He checks to see whether timestamp T_1 is valid or

not. If it is valid, he then computes $R' = R_A \oplus R_B$ to verify whether it is equal to the payee's endorsement R . If they are equal, the customer computes the payment requirement $C_A = h(R \oplus M_i \oplus Q_{M_i})$, where M_i is the face value of the electronic traveler's check, and Q_{M_i} is the amount of the electronic traveler's check. Then, M_i, Q_{M_i}, C_A and a timestamp T_2 would be encrypted by the customer using symmetric key R_A , and the result $E_{R_A}(M_i, Q_{M_i}, C_A, T)$ would be sent to the bank.

Step 4. The bank decrypts $E_{R_A}(M_i, Q_{M_i}, C_A, T)$ to obtain M_i, Q_{M_i}, C_A, T_2 . It checks whether timestamp T_2 is within a reasonable range. If it is, the bank computes $C'_A = h(M_i \oplus Q_{M_i} \oplus R_A \oplus R_B)$ ($= h(R \oplus M_i \oplus Q_{M_i})$) to verify whether C'_A is equal to C_A . If so, the bank computes $TC_{M_i} = h(R \oplus M_i \parallel S_{Q_{M_i}})$, where $S_{Q_{M_i}}$ is a serial number, \parallel denotes a concatenation operation, and TC_{M_i} is an electronic traveler's check. After that, the bank computes and sends $E_{R_A}(TC_{M_i}, S_{Q_{M_i}})$ to the customer. Then, the bank sends TC_{M_i} to the clearing-house for recording and safekeeping and stores it in the smart card which was issued to the customer by the bank. If this is the second time the customer buys an electronic traveler's check, he does not need to register again and can begin with a new withdrawal phase.

Step 5. The customer decrypts $E_{R_A}(TC_{M_i}, S_{Q_{M_i}})$ to obtain TC_{M_i} and $S_{Q_{M_i}}$. He then computes $TC'_{M_i} = h(R_A \oplus R_B \oplus M_i \parallel S_{Q_{M_i}})$ ($= h(R \oplus M_i \parallel S_{Q_{M_i}})$) and compares the newly decrypted TC_{M_i} with TC'_{M_i} to verify whether the electronic traveler's check is legitimate. If the verification succeeds, the customer would store TC_{M_i} and the serial number $S_{Q_{M_i}}$ in his smart

card.

If the electronic traveler's check is lost, the customer should send $(TC_{M_i}, S_{Q_{M_i}})$ to report the loss. The bank can then reissue a new electronic traveler's check $TC_{M_i} (= h(R_A \oplus R_B \oplus M_i \parallel S_{Q_{M_i}}))$ to the customer.

(c) The payment phase

In this phase, a customer buys goods from a merchant with an electronic traveler's check. This phase includes following four steps:

Step 1. When the customer buy goods, he should encrypt the information

$TC_{M_i}, M_i, S_{Q_{M_i}}, T_3$ by using his anonymous identity B_ID_A as a symmetric

key to generate the check message $E_{B_ID_A}(TC_{M_i}, M_i, S_{Q_{M_i}}, T_3)$ and send it to

the merchant.

Step 2. After receiving $E_{B_ID_A}(TC_{M_i}, M_i, S_{Q_{M_i}}, T_3)$, the merchant should forward it

to the bank.

Step 3. The bank decrypts the check message to obtain TC_{M_i} , the face value M_i ,

$S_{Q_{M_i}}$ and timestamp T_3 . Then, it checks whether timestamp T_3 is within a

reasonable range and verifies whether $TC_{M_i} = h(R \oplus M_i \parallel S_{Q_{M_i}})$ holds or not.

If both hold, The bank sends TC_{M_i} to the clearing-house via a secure

channel and verifies whether it is a double spending by using the serial

number $S_{Q_{M_i}}$. If it is a double spending, the bank can find out the real

identity of the customer by using the value B_ID_A . If all the verifications

of $TC_{M_i}, M_i, S_{Q_{M_i}}$, and T_3 are successful, the bank deposits it and

computes $C_{M_i} = h(TC_{M_i} \oplus R_B)$. It then computes and sends $E_{R_B}(C_{M_i}, TC_{M_i})$

to the merchant.

Step 4. After the merchant has received the electronic traveler's check by the secret message from the bank, the transaction has been finished.

(d) The deposit phase

In this phase, the merchant sends the received electronic traveler's check to the bank, the bank will verify whether this check contains his own signature. If so, the bank will deposit the amount of money on the electronic traveler's check into the merchant's account. The deposit phase includes following two steps:

Step 1. The merchant sends $E_{R_B}(C_{M_i}, TC_{M_i})$ and a timestamp T_4 to the bank.

Step 2. The bank decrypts $E_{R_B}(C_{M_i}, TC_{M_i})$ to obtain C_{M_i} and TC_{M_i} , and checks whether timestamp T_4 is within a reasonable range. If so, the bank verifies whether $C_{M_i} = TC_{M_i} \oplus R_B$ holds. If it holds, TC_{M_i} would be sent to the clearing-house via a secure channel to verify its validity. If it is valid, the bank deposits the amount on the electronic traveler's check into the customer's account.

2.2 The off-line subprotocol

The difference between the on-line and off-line subprotocol is that the bank and the clearing-house are not involved in the off-line version. This off-line scheme also contains four phases. We describe the off-line subprotocol as follows and also show it in Figure 2.

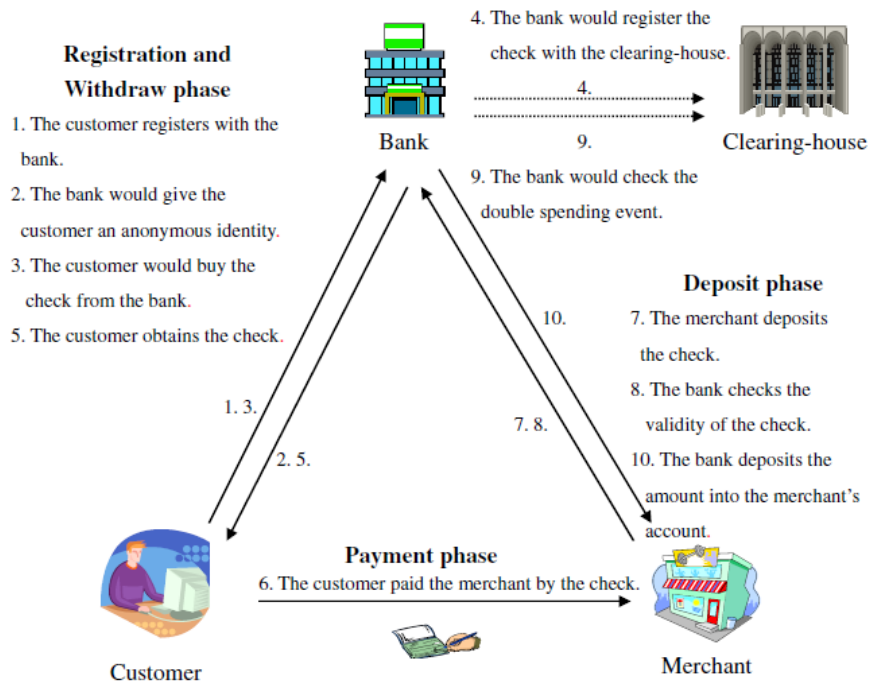


Figure 2: The steps of the off-line subprotocol (source: [1])

(a) The registration phase

This phase is the same as the on-line registration phase.

(b) The withdraw phase

This phase is the same as the on-line withdraw phase.

(c) The payment phase

After the customer has bought the merchandise, he uses the electronic traveler's check to pay the merchant. The payment phase includes the following two steps:

Step 1. After the customer has selected goods to buy, he encrypts the electronic traveler's check TC_{M_i} , serial number $S_{Q_{M_i}}$ and timestamp T_3 by using the merchant's public key Y_C to generate the check message $E_{Y_C}(TC_{M_i}, M_i, S_{Q_{M_i}}, T_3)$, and then sends it to the merchant.

Step 2. After receiving the check message $E_{Y_C}(TC_{M_i}, M_i, S_{Q_{M_i}}, T_3)$, the merchant decrypts it by using its private key X_C to obtain $TC_{M_i}, M_i, S_{Q_{M_i}}$, and T_3 , and checks whether timestamp T_3 is within a reasonable range. After the merchant has confirmed that both T_3 is in time and the amount M_i is correct, the merchant delivers the goods to the customer.

(d) The deposit phase

In this phase, the merchant sends the electronic traveler's check to the bank. The bank would verify whether it is valid. If so, the bank sends the check to the clearing-house for confirming whether double spending happens. If it does not occur, the bank deposits the amount of the electronic traveler's check to the merchant's account. The deposit phase includes the following two steps:

Step 1. The merchant encrypts $TC_{M_i}, M_i, S_{Q_{M_i}}$, and the timestamp T_4 by using bank's public key Y_B and sends the result $E_{Y_B}(TC_{M_i}, M_i, S_{Q_{M_i}}, T_4)$ to the bank.

Step 2. The bank decrypts the received $E_{Y_B}(TC_{M_i}, M_i, S_{Q_{M_i}}, T_4)$ by using his private key X_B , obtaining TC_{M_i} , the amount M_i , serial number $S_{Q_{M_i}}$, and T_4 . It would then check whether timestamp T_4 is within a reasonable range. If so, the bank computes $TC'_{M_i} = h(R \oplus M_i \parallel S_{Q_{M_i}})$ and

verifies whether it is equal to TC_{M_i} . If it is, the bank sends this check to the clearing-house to check whether double depositing or double spending occurs. If both do not exist, the bank deposits the amount of this check into the merchant's account.

3. KCI and parallel session attacks on Liaw et al.'s scheme

KCI attack means that if an attacker E knows the private key of A, he can masquerade as others to communicate with A [2]. Parallel session (PS) attack means two or more runs of a protocol are executed concurrently under the attacker's orchestration [17] as described in Introduction. In the registration phase, Liaw et al.'s scheme does not need a secure channel. We found that this makes their scheme suffer from KCI and parallel session attack. In section 3.1 and 3.2, we show the KCI and PS attacks on their on-line and off-line versions, respectively.

3.1 Attack on the on-line subprotocol

Assume that attacker E obtains the secret key X_A of customer A. He then can masquerade as bank B to communicate with A which is possible since there may exist a site spoofing attack. We show the attacks on the registration phase and the withdraw phase as follows which are also shown in Figure 3 and Figure 4, respectively.

(a) On the registration phase

We show our attack on the registration phase by using the following steps.

Step 1. When customer A registers to bank B. He chooses a random number K_{A1} and generates the customer's anonymous identity R_ID_A by computing $R_ID_A = h(ID_A \oplus K_{A1})$. He then encrypts it by using the public key Y_B of the bank and sends the encryption $PE_{Y_B}(R_ID_A)$ to E who is now pretending to be bank B as shown in window 1 of Figure 3.

Step 2. After receiving $PE_{Y_B}(R_ID_A)$, E opens window 2 and pretends as customer A to communicate with B. He retransmits the received encryption to B

Step 3. Bank B decrypts the received information by his private key, obtaining R_ID_A . Then, B chooses a random number K_{B1} and computes $B_ID_A = h(R_ID_A \oplus K_{B1})$. He encrypts B_ID_A by using the customer's public key Y_A and sends the result $PE_{Y_A}(B_ID_A)$ to E who now pretends as customer A.

Step 4. After receiving $PE_{Y_A}(B_ID_A)$, E decrypts it by using the private key X_A

to get the anonymous identity B_ID_A . He then encrypts B_ID_A by using A's public key and sends the result $PE_{Y_A}(B_ID_A)$ to A in window 1 of Figure 3.

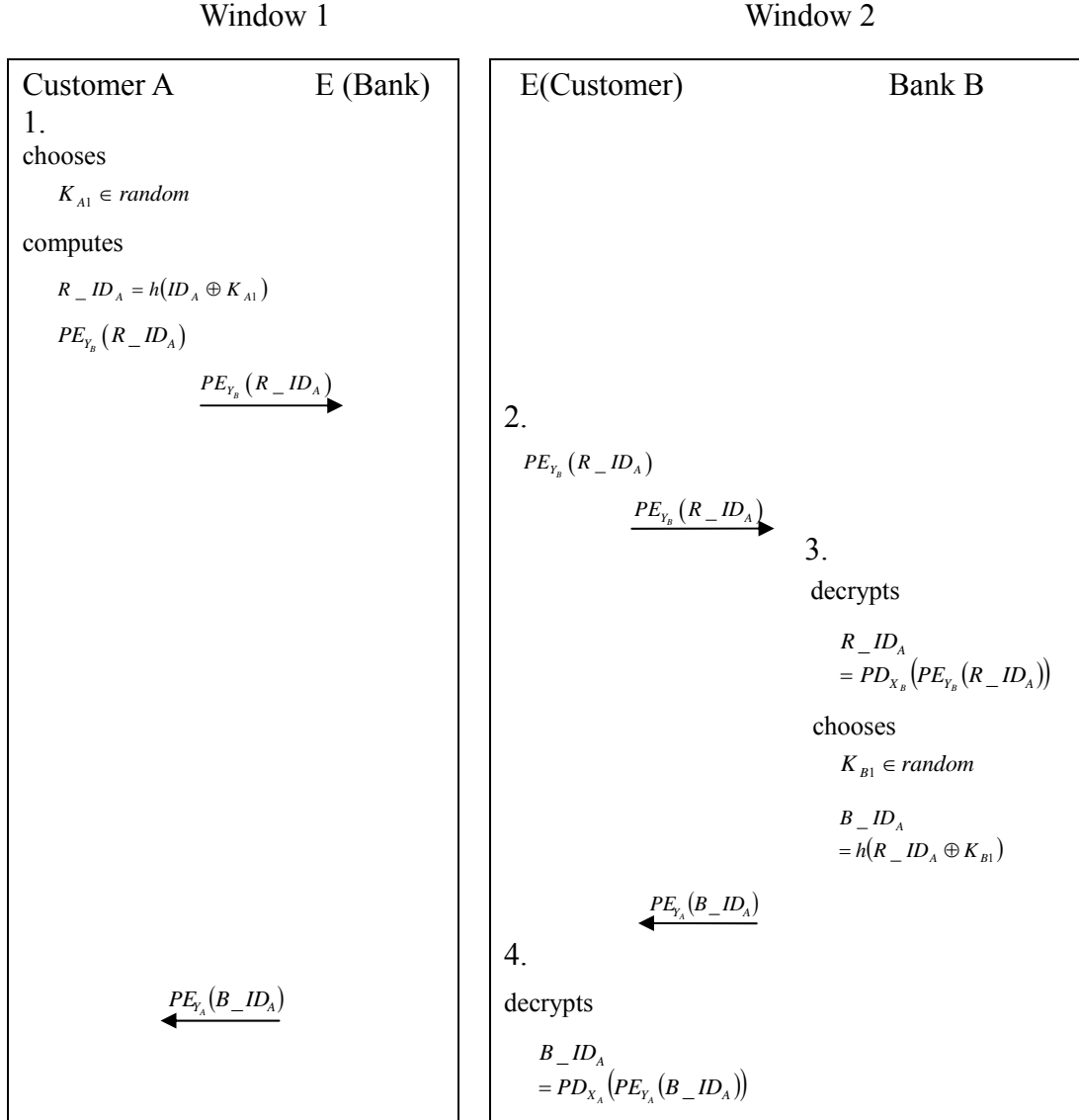


Fig 3: Our attack on registration phase

(b) On the withdraw phase

After having the anonymous identity B_ID_A from the registration phase, E can masquerade as customer A to buy an electronic traveler's check from bank B. We show our attack on the withdraw phase as follows and also depict it in Figure 4.

Step 1. E masquerades as customer A. He chooses a random number K_{A2} , computes $R_A = h(B_ID_A \oplus K_{A2})$, and encrypts R_A by using B_ID_A as the symmetric key. He then sends the result $E_{B_ID_A}(R_A)$ to bank B.

Step 2. Bank B decrypts the received message, obtaining R_A . Then, it chooses a random number K_{B2} , computes $R_B = h(X_B \oplus K_{B2})$ by using his private key X_B , and calculates $R = R_A \oplus R_B$. It then encrypts R , R_B , and timestamp T_1 by the symmetric key B_ID_A and sends the result $E_{B_ID_A}(R, R_B, T_1)$ to E.

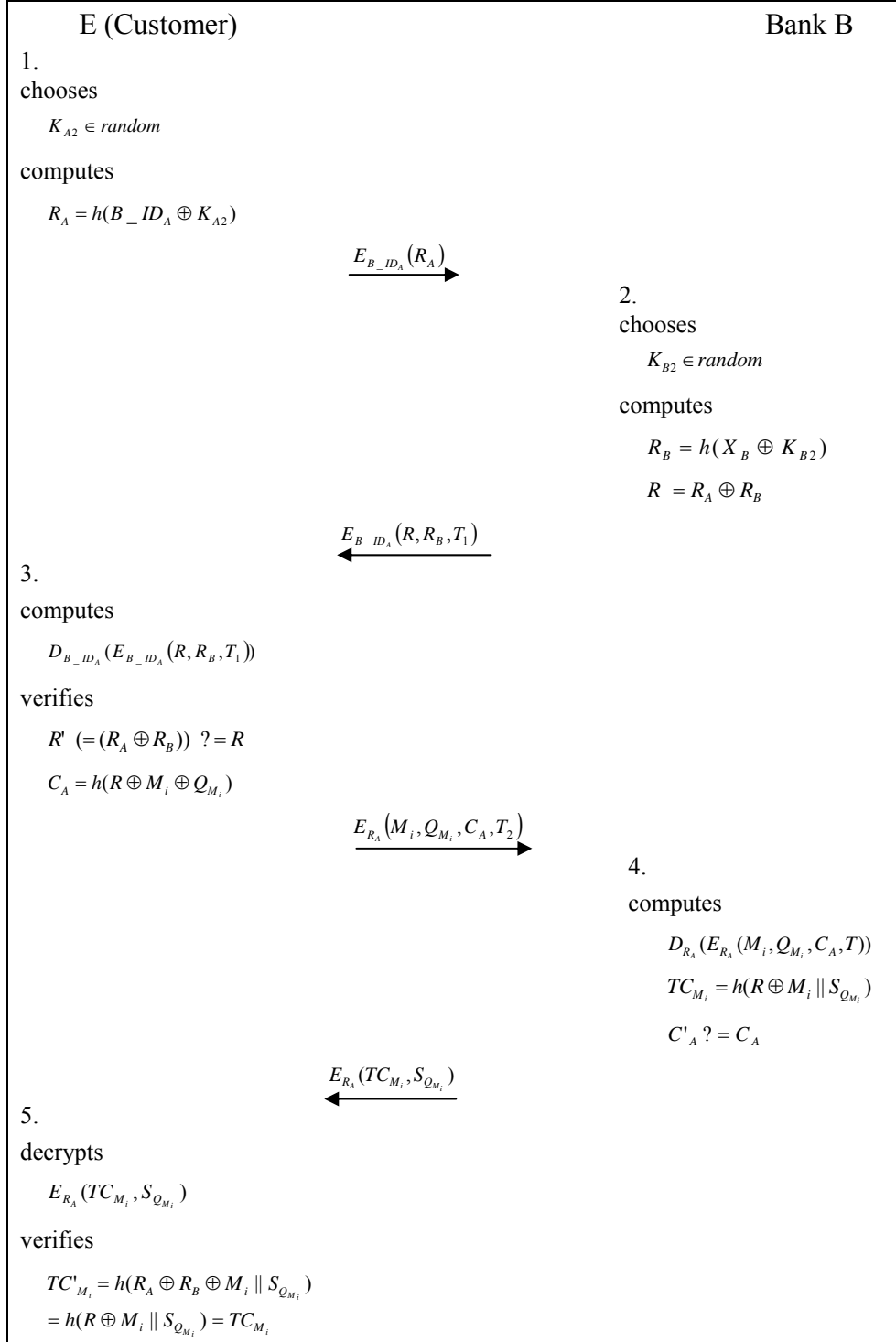


Figure 4: Our attack on withdraw phase

Step 3. E decrypts the received message by using B_ID_A and obtains R, R_B , and timestamp T_1 . E computes $R' = R_A \oplus R_B$ to verify whether this newly computed R' is equal to R , and checks whether the timestamp T_1 is valid. If both are correct, E computes $C_A = h(R \oplus M_i \oplus Q_{M_i})$, where R is the payee's endorsement, M_i is the face value of the electronic traveler's check, Q_{M_i} is the amount of the electronic traveler's check, and C_A is the payment requirement. E then chooses a timestamp T_2 and encrypts M_i, Q_{M_i}, C_A , and T_2 by using symmetric key R_A and sends the result $E_{R_A}(M_i, Q_{M_i}, C_A, T_2)$ to bank B

Step 4. Bank B decrypts $E_{R_A}(M_i, Q_{M_i}, C_A, T_2)$, obtaining M_i, Q_{M_i}, C_A and T_2 . Then, B checks whether T_2 is within a reasonable range or not. If so, bank B computes $C'_A = h(M_i \oplus Q_{M_i} \oplus R_A \oplus R_B)$ to verify whether C'_A is equal to C_A . If so, B generates the electronic traveler's check TC_{M_i} by computing $TC_{M_i} = h(R \oplus M_i \| S_{Q_{M_i}})$, where $S_{Q_{M_i}}$ is the serial number of the electronic traveler's check chosen by B. Then, B sends TC_{M_i} to the clearing-house for recording and safekeeping. After that, bank B encrypts and sends $E_{R_A}(TC_{M_i}, S_{Q_{M_i}})$ to E. If this is the second time E buys an electronic traveler's check, he does not need to register again. He can begin with a new withdraw phase.

Step 5. E decrypts $E_{R_A}(TC_{M_i}, S_{Q_{M_i}})$ by using R_A to obtain TC_{M_i} and $S_{Q_{M_i}}$.

He then computes $TC'_{M_i} = h(R_A \oplus R_B \oplus M_i \| S_{Q_{M_i}})$ to compare with TC_{M_i} to verify whether both are equal. If the verification succeeds, E stores TC_{M_i} and the serial number $S_{Q_{M_i}}$ in his smart card.

When the electronic traveler's check is lost, E should send $(TC_{M_i}, S_{Q_{M_i}})$ to report the loss. Then, bank B can reissue a new electronic traveler's check $TC_{M_i} = h(R_A \oplus R_B \oplus M_i \parallel S_{Q_{M_i}})$ to E.

(c) On the payment phase

E can buy goods with an electronic traveler's check withdrawn from a merchant. The payment phase includes following four steps and also is depicted in Figure 5.

Step 1. When buying goods, E should encrypt the information $TC_{M_i}, M_i, S_{Q_{M_i}}, T_3$ by using his anonymous identity B_ID_A as a symmetric key to form the message $E_{B_ID_A}(TC_{M_i}, M_i, S_{Q_{M_i}}, T_3)$ which will be then sent to the merchant.

Step 2. After receiving $E_{B_ID_A}(TC_{M_i}, M_i, S_{Q_{M_i}}, T_3)$, the merchant should forward it to bank B.

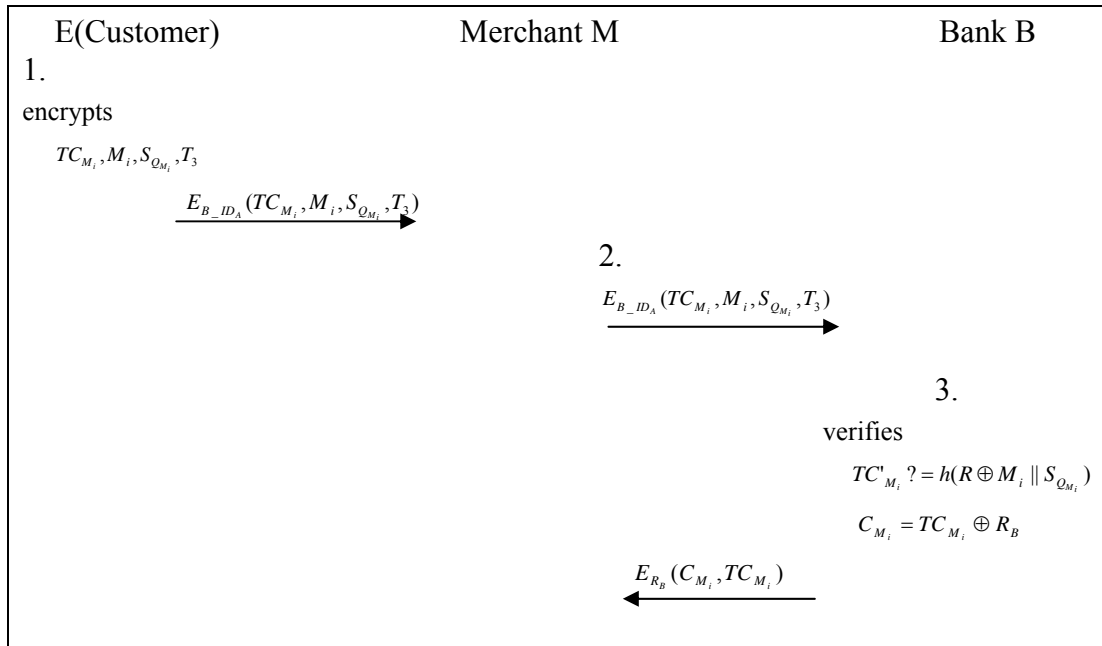


Figure 5. Payment phase

Step 3. B decrypts the message to obtain TC_{M_i} , the face value M_i , the serial number $S_{Q_{M_i}}$, and timestamp T_3 . Then, it checks to see whether timestamp T_3 is within a reasonable range. If so, B computes

$TC'_{M_i} = h(R \oplus M_i || S_{Q_{M_i}})$ and compares it with TC_{M_i} to verify whether TC'_{M_i} is valid. If it is valid, B should also send TC_{M_i} to the clearing-house via a secure channel for verifying whether TC_{M_i} has been double spending by using the serial number $S_{Q_{M_i}}$. If the electronic check has been double spent, bank B can find out the real identity of the customer by value B_ID_A . If the verifications of all the $TC_{M_i}, M_i, S_{Q_{M_i}}, T_3$, and the identity are all successful, the bank computes $C_{M_i} = TC_{M_i} \oplus R_B$ and encrypts both C_{M_i} and TC_{M_i} by using R_B as a symmetric key. The result $E_{R_B}(C_{M_i}, TC_{M_i})$ will be sent to the merchant. Only after the electronic traveler's check TC_{M_i} has passed the bank's verification, it can then be deposited in the bank.

Step 4. When the merchant receives the electronic traveler's check from the bank, the transaction has been finished.

(d) On the deposit phase

In this phase, merchant M sends the electronic traveler's check to bank B, then bank B will verify his own digital signature on the electronic traveler's check. If the signature is valid, B will deposit the amount on the electronic traveler's check into the merchant's account. The deposit phase includes following two steps and also is depicted it in Figure 6.

Step 1. The merchant sends both $E_{R_B}(C_{M_i}, TC_{M_i})$ and a timestamp T_4 to bank B.

Step 2. Bank B checks whether timestamp T_4 is within a reasonable range. If so, it decrypts $E_{R_B}(C_{M_i}, TC_{M_i})$, obtaining C_{M_i} and TC_{M_i} . Then, B computes and verifies whether $C'_{M_i} (= TC_{M_i} \oplus R_B) = C_{M_i}$ holds, If it holds, TC_{M_i} would be sent to the clearing-house via a secure channel to check

whether it is unredeemed or not. If the electronic traveler's check is unredeemed, bank B should deposit the amount on the electronic traveler's check into E's account.

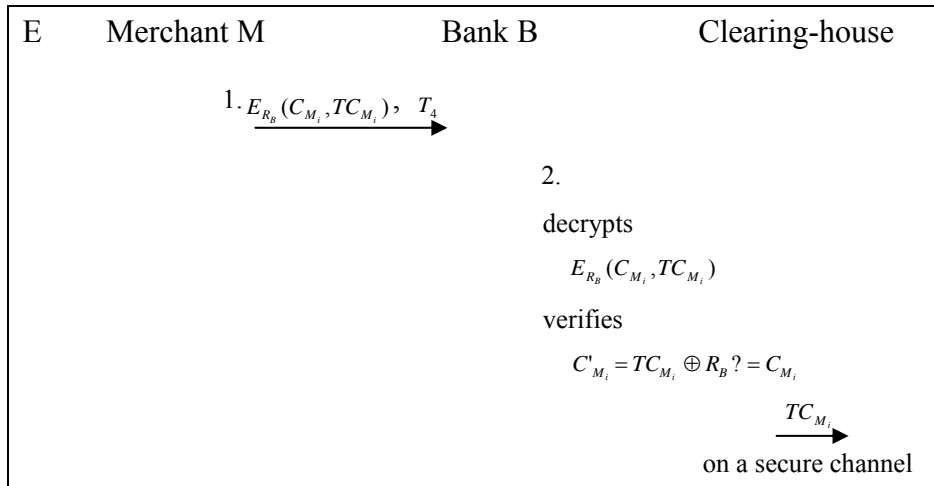


Figure 6: On the deposit phase

Finally, adversary E has successfully obtained the electronic traveler's check. In the other words, attacker E achieves the payment phase and the deposit phase successfully.

3.2 KCI and Parallel session attack on the off-line subprotocol

The difference between the on-line and off-line version is that bank B and clearing-house are not involved in the off-line subprotocol. Our attacks on the off-line registration phase and off-line withdraw phase are the same as the ones launched on the on-line registration phase and on the on-line withdraw phase, respectively. We omit them here.

4. Our improvement

To prevent the KCI and parallel session attack in Liaw et al.'s scheme, we improve both the registration phase and the withdraw phase in the on-line and off-line subprotocol respectively. The other phases will be the same as the ones in Liaw et al.'s scheme. The details of our improvement for the on-line subprotocol are described in Section 4.1 and improvement for the off-line subprotocol is similar to the one on the on-line version as described in Section 4.2.

4.1 Improvement on the on-line subprotocol

In this section, we show our improvements on the registration phase and withdraw phase respectively to get rid of the KCI and parallel session attack. The

other two phases are the same as the ones in the original version.

(a) On the registration phase

We show this phase by using the following steps and also depict it in Figure 7.

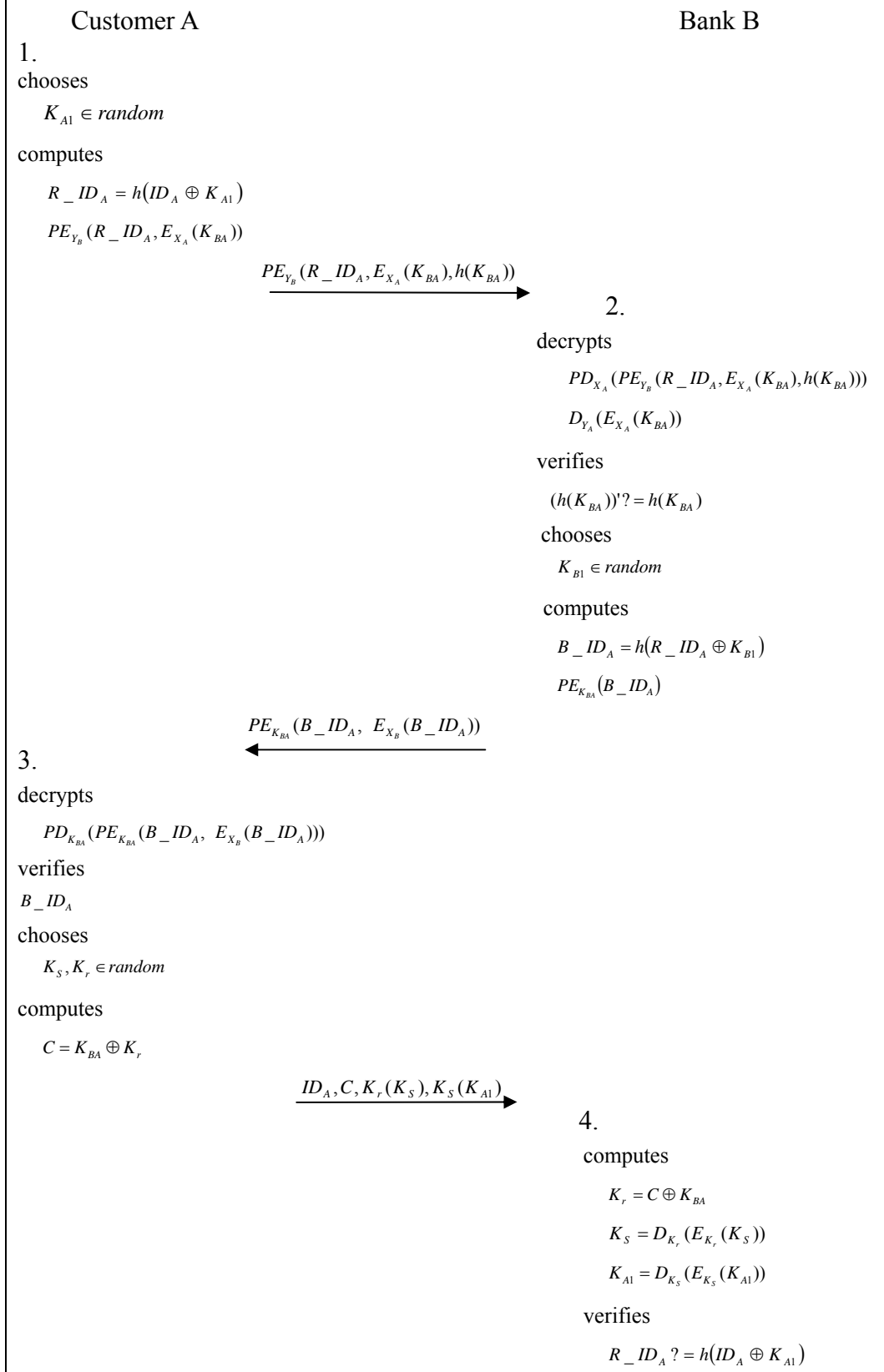


Figure 7 . Our improvement on the registration phase

Step 1. The customer chooses a random number K_{A1} . He also randomly chooses K_{BA} as the session key shared with bank B. He then uses his private key PE_{X_A} to encrypt K_{BA} and computes $h(K_{BA})$ and $R_ID_A = h(ID_A \oplus K_{A1})$. Then, he encrypts R_ID_A , $PE_{X_A}(K_{BA})$, and $h(K_{BA})$ by using the bank's public key Y_B and sends the result $PE_{Y_B}(R_ID_A, PE_{X_A}(K_{BA}), h(K_{BA}))$ to bank B.

Step 2. After receiving the message, B decrypts it using his private key, obtaining R_ID_A , $PE_{X_A}(K_{BA})$, and $h(K_{BA})$. He then uses A's public key to decrypt $PE_{X_A}(K_{BA})$, obtaining K_{BA} , and uses hash function $h(\cdot)$ to compute $h(K_{BA})$ and compares it to the one in the decryption result. If they are equal, B chooses a random number K_{B1} , computes $B_ID_A = h(R_ID_A \oplus K_{B1})$, and signs on B_ID_A by using his private key. Then, B encrypts both B_ID_A and the signature $E_{X_B}(B_ID_A)$ by using K_{BA} and sends the result to customer A.

Step 3. After receiving the message, A decrypts it, obtaining B_ID_A and B's signature. He verifies the signature. If it is valid, A chooses a random number K_S as the session key shared with B and K_r as the encryption key of K_S . He computes $C = K_{BA} \oplus K_r$ and sends $ID_A, C, K_r(K_S), K_S(K_{A1})$ to B.

Step 4. After receiving the message from A, B computes $K_r = C \oplus K_{BA}$, $K_S = D_{K_r}(E_{K_r}(K_S))$, $K_{A1} = D_{K_S}(E_{K_S}(K_{A1}))$, and verifies whether $R_ID_A = h(ID_A \oplus K_{A1})$ holds. If they are equal, B accepts A's registration.

(b) On the withdraw phase

In this phase, both communicating parties (A and B) use K_S as the session key

to communicate with each other. We show our improvement on the withdraw phase by using the following steps and also depict it in Figure 8

Step 1. Customer A chooses a random number K_{A2} and computes $R_A = h(B_ID_A \oplus K_{A2})$. Then, he encrypts B_ID_A and R_A by session key K_S and sends the result $E_{K_S}(B_ID_A, R_A)$ to the bank.

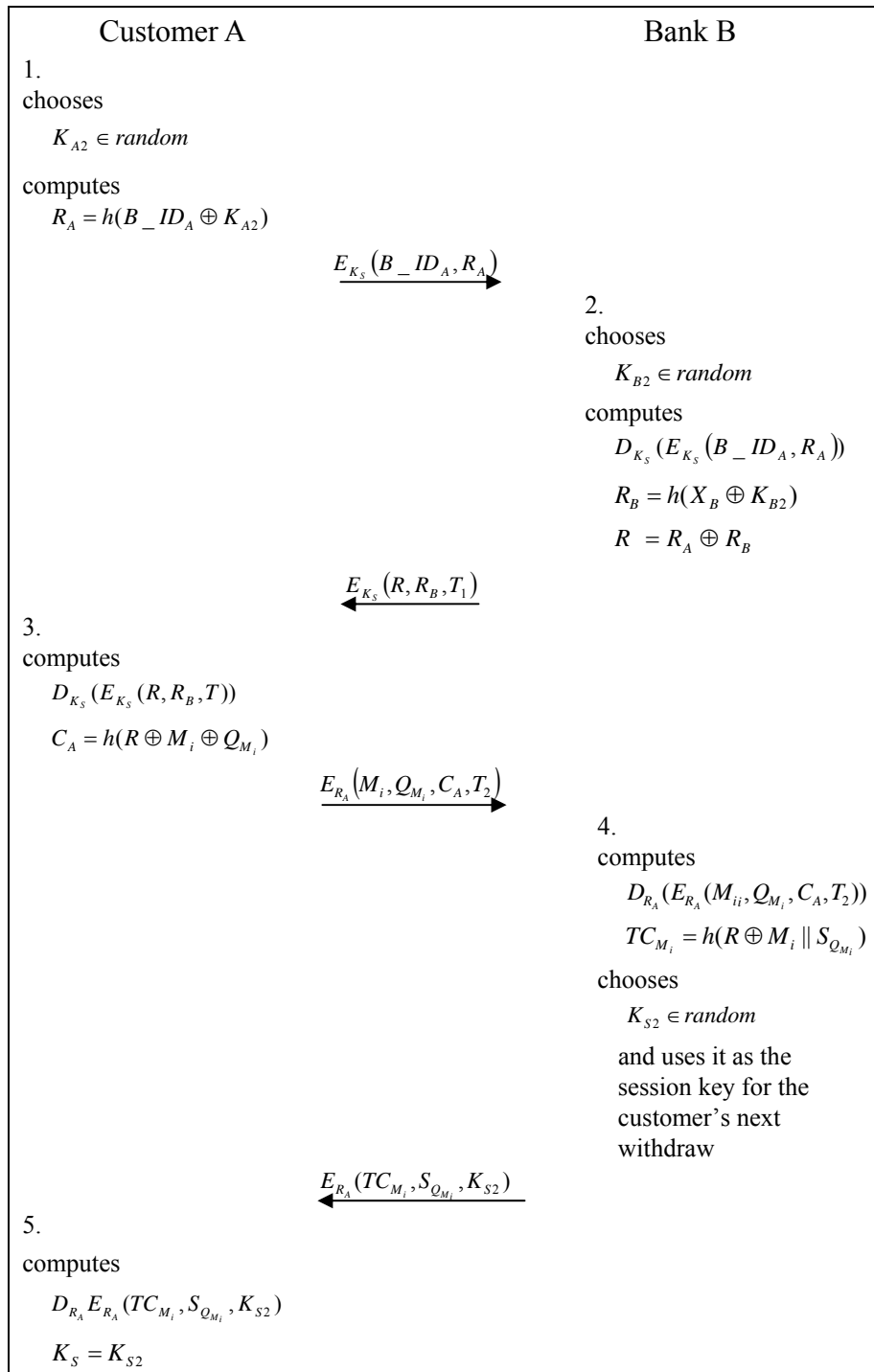


Figure 8 . Our improvement on the withdraw phase

Step 2. B chooses a random number K_{B2} and computes $D_{K_S}(E_{K_S}(B_ID_A, R_A))$,

$R_B = h(X_B \oplus K_{B2})$, and $R = R_A \oplus R_B$. He then encrypts R , R_B and

timestamp T_1 by K_S and sends the result $E_{K_S}(R, R_B, T_1)$ to A.

Step 3. A decrypts the received message, obtaining R , R_B and T_1 . He checks to

see whether T_1 is valid. If it is valid, A computes

$C_A = h(R \oplus M_i \oplus Q_{M_i})$ and encrypts M_i, Q_{M_i}, C_A and timestamp T_2

by R_A and sends the result $E_{R_A}(M_i, Q_{M_i}, C_A, T_2)$ to B.

Step 4. B decrypts the message from A, obtaining M_i, Q_{M_i}, C_A and T_2 . He

checks to see whether T_2 is valid. If it is valid, B computes

$TC_{M_i} = h(R \oplus M_i \parallel S_{Q_{M_i}})$, chooses a random number K_{S2} to be the next

session key for A's next withdraw, and encrypts $TC_{M_i}, S_{Q_{M_i}}, K_{S2}$ by R_A .

He then sends the result $E_{R_A}(TC_{M_i}, S_{Q_{M_i}}, K_{S2})$ to A.

Step 5. A decrypts the received message, obtaining $TC_{M_i}, S_{Q_{M_i}}$ and K_{S2} . He

replaces K_S as K_{S2} for the next protocol run.

In our improvement, even if an attacker knows the private key of customer A, he can not masquerade as B to communicate with A. That is, the KCI and PS attack fails.

4.2. Improvement on the off-line subprotocol

The difference between the on-line and off-line subprotocols is that bank B and the clearing-house are not involved in the off-line version but the on-line version involves both of them. Since our improvement makes no relationship to this point, our improvement on the off-line subprotocol is the same as the on-line version. We therefore omit it here.

5. Conclusion

In this paper, we propose a KCI and parallel session attack on both Liaw et al.'s an on-line and off-line electronic traveler's check subprotocols. We also propose an

improvement to resist against the attack we launch.

References

- [1] Horng-Twu Liaw, Jiann-Fu Lin, Wei-Chen Wu, "A new electronic traveler's check scheme based on one-way hash function," *Electronic Commerce Research and Applications* 6, 2007, pp. 499–508.
- [2] S. B. Wilson, A. Menezes, "Authenticated Diffie-Hellman key agreement protocols", *Proceedings of the 5th Annual Workshop on Selected Areas in Cryptography (SAC'98)*, Lecture Notes in Computer Science, 1999, pp. 339-361.
- [3] WANG et.al, "Improvement of McCullagh-Barreto key agreement with KCI-security," *The Journal of China Universities of Posts and Telecommunications*, April 2009, 16(2), pp. 68–71.
- [4] Yong Xu, Jindi Liu, "Electronic Check System Design Based on RFID," *Information Science and Engineering (ICISE)*, 2009 1st International Conference on IEEE Conferences , 2010, pp. 5345 – 5348.
- [5] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithm," *IEEE Trans. Inform. Theory* IT-31 (4) 1985, pp. 469–472.
- [6] C.J. Tsao, C.Y. Chen, C.Y. Ku, "An electronic bearer check system," *IEICE Trans. Commun.* E85-B (1), 2002, pp. 325–331.
- [7] Jong-E Hsien, Chih-Cheng Hsueh, Chien-Yuan Chen, "An electronic Traveler's check system," in: *2001 Conference on Theory and Practice for Electronic Commerce*, Taiwan, 2001, pp. 164–169.
- [8] Shengbao Wang, Zhenfu Cao, Maurizio Adriano Strangio and Lihua Wang, "Cryptanalysis and Improvement of an Elliptic Curve Diffie-Hellman Key Agreement Protocol," December 14, 2007.
- [9] Wei-Kuei Chen, "Efficient on-line electronic checks," *Applied Mathematics and Computation* 162, 2005, pp. 1259–1263.
- [10] Jianhong Zhang^{1,2}, Wei Zou¹, Dan Chen³ and Yumin Wang. "On the Security of a Digital Signature with Message Recovery Using Self-certified Public Key," *Informatica* 29, 2005, pp. 343–346.
- [11] WANG Jian-hui, LIU Jing-wei, LI Xiao-hui, KOU Wei-dong, "Fair e-payment protocol based on blind signature," *The Journal of China Universities of Posts and Telecommunications* October 2009, 16(5), pp. 114–118.
- [12] Jesus Tellez Isaac a , Jose Sierra Camara b, Sherali Zeadally c, Joaquin Torres Marquez b. "A secure vehicle-to-roadside communication payment protocol in vehicular ad hoc networks," *Computer Communications* 31, 2008, pp 2478–2484.

- [13] W. Diffie and M.E. Hellman, “New Directions in Cryptography,” IEEE Transactions on Information Theory, Vol. IT-22, No.6, November, 1976, pp. 644-654
- [14] Zhang, Q. Markantonakis, K. Mayes, K. “A Mutual Authentication Enabled Fair-Exchange and Anonymous E-Payment Protocol,” 26-29 June 2006, pp.20 – 20.
- [15] Zhao Huawei Liu Ruixia, “A Scheme to Improve Security of SSL,Circuits, Communications and Systems,” 2009. PACCS '09. Pacific-Asia Conference on IEEE Conferences, 2009, pp. 401 – 404.
- [16] Harb, H, Farahat, H, Ezz, M, “SecureSMSPay: Secure SMS Mobile Payment model,” Anti-counterfeiting, Security and Identification, 2008. ASID 2008. 2nd International Conference on IEEE Conferences, 2008 , pp. 11 – 17.
- [17] Wenbo Mao, “*Modern Cryptography-Theory & Practice*”, Practice Hall Inc, 2004.