Improvement of a Certificate less Signcryption Scheme without pairing

Xiao Zheng [a], Xiaohuan Yang

School of Computer and Software Engineering, Xihua University, Chengdu 610039, China

[a]xiao_zheng0910@163.com

Abstract. Recently, signcryption is widely attention since it could provide signature and encryption simultaneously, some certificate less signcryption (CLSC) schemes using bilinear pairing have been proposed, Compare to other operations, the bilinear pairing is time-consuming, So CLSC scheme without bilinear pairing is more practical, Recently, Shi et al. proposed a CLSC scheme without bilinear pairing and proved their schemes is secure against two types of adversaries. This paper puts forward a new certificate less signcryption based on Schnorr signature and computational Daffier-Hellman problem, which is more efficient than Shi et al. schemes, and demonstrates it is secure in the random oracle model.

Keywords: Certificate less, Signcryption scheme, bilinear pairing, and Random oracle model.

1. Introduction

Signcryption, first addressed by Zheng [1], is a cryptographic primitive that realizes both the function of digital signature and public key encryption simultaneously. Previous signcryption schemes are based on the traditional public key infrastructure (PKI) or cryptography mechanism based on identity, efficiency is low or the trusted third party is strongly dependent on. In 2003, certificate less of public key cryptography system (CLPKC) is proposed for the first time by Al-Riyami [2], user's keys consists of two parts: key generation center (KGC) generated partial private key and the secret value which is chosen by the user. It overcomes key escrow problem based on the identity cryptography, also eliminates the complex management problems based on public key certificate in traditional public key infrastructure, greatly improving the efficiency. In 2008, Barbosa and Farshim [11] first proposed certificate less signcryption (CLSC) scheme. Since then, some certificate less signcryption scheme have been proposed, most of these schemes is based on the bilinear pairings computation, and the bilinear pairings computation is known as one of the most complex cryptography operations [3]. The time needed for running a bilinear pairings about 10 times of index algorithms on finite field. Therefore, recently, certificate less signcryption without pairing is becoming research focus. In 2010, Xie et. Al. proposed a without bilinear pairings signcryption scheme [4], the computing efficiency is still not ideal. In the same year, Zhu Hui etc. proposed a certificate less signcryption scheme based on the discrete logarithm problem (DLP) [5], and pointed out that the scheme satisfies public verifiability and forward security; in 2011, Liu et al. put forward a high efficiency certificate less signcryption scheme [6]. Although the two solutions in terms of efficiency higher than Xie's scheme, but because of that user's partial private key and the secret value in the process of signcryption binding used as a private key, such that the type I of attacker can forge effective signcryption via public key replacement attack [7]. Public key replacement attack often is effective for certificate less signcryption scheme [8], so their schemes exist adaptively forgery attack and confidentiality issues. To solve these problems, some solutions were proposed including based on discrete logarithm [12] and prevent malicious-but-passive KGC attack [13].

In this paper, we put forward a certificate less signcryption scheme based on Schnorr signature [9-10] and Computational Diffie-Hellman problem (CDH), motivated by certificate less encryption scheme proposed in[14,15,16],our scheme is provably secure in the random oracle model. A new scheme to signcryption and unsigncryption process using 8 times mode multiplication operations and 4 hash operations, Efficiency is higher than the literature [4,13,16],and overcame in literature [7] cannot withstand the public key replacement attack.

The organization of the paper is sketched as follows: Section 2, we describe some preliminaries, including some complexity assumptions as well as the formal define of CLSC scheme ,In Section 3 we given the security model of CLSC, In Section 4,We present our scheme, give the security analysis in Section 5, Finally, we draw some conclusions in Section 6.

2. Preliminaries

2.1 Computational problem.

Definition 1 Discrete Logarithm problem (DLP)

Let p and q be primes such that $q|(p-1)$. let g be a generator of Z_p^* with orderq, given $(g, g^x)$for unknownx $\in Z_p^*$, the DLP problem in Z_p^* is to find x.

Definition 2 Computational Diffie-Hellman problem (CDH)

Given $(g, g^a, g^b)$for unknown $a, b \in Z_p^*$,the CDH problem in Z_p^* is to computer g^{ab}.

2.2 CLSC Scheme.

A CLSC scheme contains the following seven algorithms:

Setup: Taking the security parameter 1^k as input and outputs the master key msk and system parameterparams, all is done by KGC.

Partial Private Key Extract: Taking the master key mkand a user's identity $ID_i \in \{0,1\}^*$ as input. This algorithm is run by the KGC to output the partial private key d_i of U_i.

SetSecret Value: Taking the security parameter k and the system parameter paramsas input, this algorithm is run by user to output his private key sk_i.

SetPrivate Key: Taking a user's secret value sk_i and the partial private key d_i as input, this algorithm is played by the user to return his private key SK_i.

SetPublic Key: This algorithm takes the master key mk and a user's secret valuesk_i, and the partial key $d_i$as input and outputs his public key PK_i.

Signcryption: It takes the sender's private keySK_S, the receiver's identity $ID_R$and public keyPK_R, and a message m as input, returns a ciphertext σ.

Unsigncryption: Taking the sender's identity ID_S and public key PK_S , the receiver's private $Sk_R$and the corresponding σ as input, the algorithm is run by the receiver to outputm.

3. Security Model for CLSC

There are two types of adversaries in the CLSC[2], i.e. the Type I adversary A_1 and the Type Ⅱ adversary A_2. The adversary A_1isn't allowed to access the master key but itcan replace arbitrary user's public key. The adversary A_2 can access to the master key but it cannot replace public key of any of the user. Note if a scheme is satisfy security for adversary A_1, then it is also suit to adversary A_2.

3.1 Confidentiality.

The security model to prove the confidentiality (indistinguishability of encryptions under adaptively chosen ciphertext attacks (IND-CCA2)) for CLSC scheme is attained by the following two games against Type-I and Type-Ⅱ adversaries.

Game 1 The first scheme is run between a challenger C and a Type-I adversary $A_1$for a CLSC scheme.

Setup: C runs this algorithm to generate system parameters params, and then gives it to the adversary A_1 while keeping msk secret.

Hash Queries: adversary A_1can ask the hash value of arbitrary input.

Partial Key Extraction: A_1 can request the partial private key d_i and partial public key Pk_i for anyID_i,C computes ID_i's the partial private key d_i and partial public keyPK_i, and then returns them toA_1.

Public Key Extraction: upon receiving any identity ID_i's public key extraction, C computes the corresponding public key PK_{ID} and sends it toA_1.

Private Key Extraction C computes the identity ID_i's private key SK_i and sends it to A_1.Here, A_1 can't access to request this oracle on any identity for which the corresponding public key has been replaced. This is because that the challenger can't provide a user's full private key for which it doesn't know the secret value.

Public Key Replacement: A user U_A's public key PK_A can be replaced with any $PK_{A'}$ supported by A, upon receiving $PK_{A'}$ fromA, C replaces the public key PK_A of ID_A with $PK_{A'}$

Signcrypt: Getting a messagem, a sender's identityID_S, private key SK_S and the public key PK_S and a receiver's identity ID_R and public key PK_R , C returns cipher text $\sigma = \mathrm{Signcrypt}(SK_{S,}PK_S, PK_R, ID_S, ID_R, m)$.

Unsigncrypt: A_1 products the sender's identity ID_S and public key PK_S and the receiver's private key SK_R, C sends unsigncrypt $(ID_S, PK_S, SK_R, \sigma)$ toA_1.

Challenge: A_1 generates two equal length messagem_0, m_1, sender identity ID_{S^*} and receiver identityID_{R^*}, C picks randomly a bit $\beta \in \{0,1\}$ and signcrypts M_β with private key of$SK_{S^*}$and the public key of PK_{R^*} to generate the challenge ciphertext $\sigma^* = \mathrm{Signcryp}(SK_{S^*}, ID_{R^*}, PK_{R^*,}m_\beta)$ and returns it toA_1.

Guess:A_1 adaptively queries the oracles except that A_1 should not ask the partial key of ID_R and alsoshouldn'task the unsigncryption on σ^* with ID_{S^*} and ID_{R^*}, Eventually, A_1 outputs a bit β', adversary A_1 wins the games if $\beta' = \beta$.

Now define the A_1's advantage as$Adv_{A_1}^{IND-CLSC-CCA2} = |2\Pr[\beta' = \beta] - 1|$.

Game 2 The second is played between challenge Cand a Type Ⅱ adversaryA_2.

Setup: Same to game 1described above.

Queries phase: Similar to game 1 IND-CLSC-CCA 2-I except that A_2 can't replace any public keys. ButA_2 can compute the partial private key of any identity with themsk by itself and get the corresponding public key.

Challenge: Same as Type-I except that A_2 wouldn't ask the receiver ID_R's private key and can't replaceID_R's public key and also cannot make an unsigncryptionquery on the challenge cipher textσ^* under ID_{S^*} andID_{R^*}.

Guess: Same as Type- I confidentiality game IND-CLSC-CMA A_2 's advantage is define as:$Adv_{A_2}^{IND-CLSC-CCA2} = |2\Pr[\beta' = \beta] - 1|$.

3.2 Unforgeability.

The security model to prove the unforgeability (existential unforgeabilityagainst choose message attacks (EUF-CMA))for CLSC scheme is acquired by the following two games against Type-I forger $\mathcal{F}_1$ and Type-Ⅱ forger $\mathcal{F}_2$.

Game 3 The third game is run between the challenger C and the forger $\mathcal{F}_1$.

Setup: Same to CLSC's IND-CCA2 game described in Section 3.1.

Queries stage:$\mathcal{F}_1$ is allowed to access all the six oracle above.$\mathcal{F}_1$adaptively requires the oracles consistent with the constraints for Type-I forger (forger $\mathcal{F}_1$ is allowed to replace arbitrary user's public key but doesn't access to the master private keymsk).

Forgery: $\mathcal{F}_1$ outputs a signcryption $(ID_{S^*}, ID_{R^*}, \sigma^*)$ where the partial key and the private key of ID_{S^*}isn't been asked and wins the game if the $\mathrm{Unsigncrypt}(ID_{S^*}, PK_{S^*}, SK_{R^*}, \sigma^*)$ is valid.

Game 4 The fourth game is run between a challengeCand a Type $-$Ⅱ adversary $\mathcal{F}_2$: A forger $\mathcal{F}_2$ is given access to all the six oracles. The only constraints is that forger $\mathcal{F}_2$ owns the master private keymsk but is not allowed to replace any user's public keys.

Note that this allows the adversary have access to the secret key of the receiver of the forgery, which ensures the insider security.

4. Our CLSC Scheme

Our scheme involve in three participants: Key Generate Center KGC, Signcryptioner A, Unsigncryptioner B,the detail of these algorithm is described as follows:

Setup: The algorithm takes as input a security parameter k to generate two large primes p,q such that $q|(p-1)$,picks a generator g with order q, choosesx $\in Z_q^*$ randomly and computesy $= g^x$mod p, choosesthreehash functions : $H_1: \{0,1\}^* \times Z_p^* \to Z_q^*$, $H_2: \{0,1\}^l \times Z_p^* \times Z_p^* \to Z_q^*$, $H_3: Z_p^* \times Z_p^* \to \{0,1\}^l$, where ℓ is the length of message to be signcrypted. The system parameters are params$=(p,q,g,y,H_1,H_2,H_3)$,x is keptsecret.

Partial Key Extract: Given params,master-key and user's identity $ID_i \in \{ID_A, ID_B\}$,KGC picks $s_i \in Z_q^*$ randomly computes $w_i = g^{s_i}$, $t_i = s_i + xH_1(ID_A, w_i)$. Returns the partial private key t_i and the partial public key$PK_i = w_i$.

Set Secret Value: User picks $z_i \in Z_q^*$ randomly, computer $u_i = g^{Z_i}$ mod p, then user outputs the secret value z_i,

Set Private Key: Taking params, t_i and z_i as input ,this algorithm returns the user's private key $SK_{ID} = (t_i, z_i)$.

Set Public Key: Taking params,PK_i as input ,then returns the user's public key $PK_{ID} = (u_i, w_i,)$.

Signcrypt: To send a message $m \in \{0,1\}^n$ to receiver with identity ID_R and public key PK_R, sender with private keySK_S works as follows:

-Check whether $g^{t_i} y^{-H_1(ID_i, w_i)} = w_i$ mod p, if not ,output ⊥.

-Randomly pick $t \in Z_q^*$ and compute$T = g^t$ and compute $h = H_2(m, T, y)$.

-Compute $S = t - hz_A - ht_A$ mod p and $h_1 = H_1(ID_B, w_B)$.

-Choose $r \in Z_q^*$ randomly and compute$R = g^r$modp, $U = \left(u_B^h w_B y^{h_B}\right)^r$modp, $c = m \oplus H_3(R, U)$, send ciphertext $\sigma = (h, S, R, c)$ to receiver.

Unsigncrypt: To unsigncrypt a ciphertext$\sigma = (h, S, R, c)$ from sender with identity ID_S and public key PK_S, receiver with private key SK_Racts as follows:

Check whether $g^{t_A} = w_A y^{H_1(ID_A, w_A)}$ if not, output ⊥ and abort.

Compute $h_1' = H_1(ID_A, w_A)$, $m' = c \oplus H_3\left(R, R^{t_B + z_B h} \bmod p\right)$.

Compute $T' = g^S w_A^h y^{H_1(ID_A, w_A)^h} u_A^h$ mod p , if and only if $h = (m', T', y)$ hold accept m, otherwise return ⊥.

5. Security Analysis of the Proposed Scheme

In this section, we will provide our scheme is provably secure in the random oracle which treats H_1, H_2, H_3 as three random oracles.

Theorem 1. Under the CDH Assumption, our CLSC scheme is IND-CCA2 secure in the random oracle model.

This theorem follows from Lemma 1.

Lemma 1. Let us assume that there exists an IND-CCA2-I adversary A_1 has non-negligible advantage ε against our scheme whenasking. q_i queries to random oracles $H_i(i = 1,2,3)$, q_s signcryption queries and q_{pkr} public key replacement queries, q_u unsigncryption queries, q_{pak}partial key queries, Assume that the Schnorr signature [9] is $(\varepsilon', q_1, q_{pak})$-secure ,Then there is an algorithmC to solve the CDH with probability$\varepsilon' \geq \alpha\left(1 - \alpha - \frac{1}{q_{rp}}\right)^{q_{sk}} \frac{\varepsilon}{q_1{}^2 q_u}$.

Proof: Suppose that there exists an adversary A_1 can attack our scheme , We want to build an algorithm C that runs A_1 as a subroutine to solve CDH problem , Assume that C is given (p, q, g, g^a, g^b) as an instance of the CDH problem , its goal is to computeg^{ab} by interact with adversary A_1.

Setup:C sets $y = g^x$mod p, param$= (p, q, g, y, H_1, H_2, H_3)$,keepsmsk secret ,ThenCsends param to A_1,meanwhile maintains a list of $L_i(i = 1,2,3)$, L_D,L_{sk},L_{pk},L_s,L_u respectively used to track A_1 asking to random oracles$H_i(i = 1,2,3)$,q_{pak},q_{sk}, q_{pk} ,q_s, q_u,At the beginning these lists are empty.

H_1 -query: For each query (ID_i, w_i) ,if L_1 List contains (ID_i, w_i, h_1) , then C returns h_1 to A_1,Otherwise C chooses $h_1 \in z_q^*$randomly, returns h_1 to A_1, and adds (ID_i, w_i, h_1) to L_1 list.

H_2-query: For each query (m, T, y), if L_2 List contains (m, T, y, h_2), then C returns h_2 to A_1, Otherwise C picks $h_2 \in Z_q^*$ randomly, returns h_2 to A_1, and adds (m, T, y, h_2) to L_2 list.

H_3-query: For each query (R, U), if L_3 List contains (R, U, v), then C returns v to A_1, Otherwise C picks $v \in (0,1)^{\ell}$ randomly,, returns v to A_1, and adds (R, U, v) to L_3 list.

Partial Private Key query: When A_1 makes this query on ID_i, C runs as follows:

If (ID, w, t) exists in List L_D ,then returns (w, t) to A_1 ,otherwise picks $t, h_1 \in Z_q^*$ randomly, computes $w = g^t y^{-h_1}$,adds (ID, w, h_1) to L_1 List, adds (ID, w, t) to L_D List, then returns (w, t) to A_1.

Public Key query: For each queryID, C runs as follows:

If (ID, u, w, δ) exists in List L_{pk}, then returns (u, w) to A_1, Otherwise picks $\delta \in \{0,1\}$ randomly, which $Pr[\delta = 1] = \alpha$.

If $\delta = 0$,thenC runs as follows: if (ID, w, t) exists in List L_D, picks $z \in Z_q^*$ randomly, computes $u = g^z \text{mod } p$, adds (ID, t, z) to List L_{sk}, and adds $(ID, u, w, 0)$ to ListL_{pk} Then returns (u, w) to A_1 .Otherwise, picks $t, h_1 \in Z_q^*$ randomly, computes $w = g^t y^{-h_1}$,adds (ID, w, h_1) to L_1 List, adds (ID, w, t) to L_D List.Then picks$z \in Z_q^*$ randomly, computes $u = g^z$ mod p, adds (ID, t, z) to L_{sk}List , adds $(ID, u, w, 0)$ to L_{pk} List , then returns (u, w) to A_1.

If$\delta = 1$:picks $z, s \in Z_q^*$ randomly, computes$u = g^z \text{mod } p, w = g^s \text{mod} p$,adds $(ID, ?, z, s)$ to L_{sk} List , adds $(ID, u, w, 1)$ to L_{pk} List, returns (u, w) to A_1.

Private Key query: For each query ID, C proceeds as follow:

C returns the previously assigned value if (ID, u, w, δ) in L_{pk} List, if $\delta = 0$, C finds (ID, t, z) in L_{sk} List, returns (t, z) to A_1, Otherwise outputs $\perp$.

Public Key Replacement query: For each identity ID, A_1 can pick a new public key replacement previously public key.

Signcrypt query: For each query (ID_A, ID_B, m), C finds (ID_A, w_A, h_{1A}) in L_1 List, Searches A′s public key and secret value in L_{pk} List and L_{sk} List respectively, then picks $S, t, h, s \in Z_q^*$ randomly, computes $T = g^S\left(w_A y^{H_1(ID_A, w_A)} u_A\right)^h$,adds $(m, T, y, h)$to L_2List, picks $r \in Z_q^*$randomly, computes $R = g^r \text{mod } p v \in \{0,1\}^{\ell}$, $c = m \oplus v$, then adds $\left(R, \left(u_B^h w_B y^{H_1(ID_B, w_B)}\right)^r \text{mod } p , v\right)$ to L_3 List, (h, S, c, R) as message of m's signcryption, then sends it to A_1.

Unsigncrypt query: For each query (h, S, c, R, ID_A, ID_B), C finds $(ID_A, u_A, w_A, \delta_A)$ in L_{pk} List, if $\delta_B = 0$ and B's public key hasn't been replaced, searches (ID_B, t_B, z_B) in L_{sk} List, finds (ID_A, w_A, h_{1A}) in L_1 List, finds $\left(R, R^{t_B + z_B h} \text{mod } p, v\right)$ in L_3 List. Then computes $T' = g^S w_A^h y^{H_1(ID_A, w_A)^h} u_A^h$ mod p, $m' = c \oplus v$, if $h = H_2(m', T', y)$ holds, then returns m', otherwise abort simulation.

If $\delta_B = 0$ and B's public key has been replaced, or $\delta_B = 1$, finds (ID_A, w_A, h_{1A}) in L_1 List, if the record of the first input for R, i.e$(R, U, v) \in L_3$,then computes $m' = c \oplus v$, if the first record of m'i.e $(m', T, y, h) \in L_2$ and $h = H_2(m', T', y)$,then returns m' to A_1, Otherwise abort.

After the above queries: A_1 sends two equal length messages m_0, m_1 and (ID_A^*, ID_B^*) which is excepted to accept challenge identity to C, the corresponding private key and partial private key for ID_B^* should not asked. if $\delta_B^* = 0$, output false, Otherwise, C finds the public key corresponding to ID_A^* and ID_B^* in theL_{pk}list, picks $b^* \in \{0,1\}, S^*, t^*, s^*, h^* \in Z_q^*$ randomly, find$(ID_A^*, w_A^*, h_{1A}^*)$ in L_1List. computes $T^* = g^S\left(w_A y^{H_1(ID_A, w_A)} u_A\right)^h \text{mod } p$,adds (m_{b^*}, T^*, y, h^*) to L_2 List, picks $v^* \in \{0,1\}^{\ell}$,sets$R^* = g^b$, $c^* = m_{b^*} \oplus v^*$, sends challenge $\sigma^* = (h^*, S^*, c^*, R^*)$ to A_1,though C unknown $\left({u_B^*}^{h^*} w_B^* y^{h_{1B}^*}\right)^b$ mod p,But v^*simulates value of H_3.A_1can continue to run polynomial bounded queries to random oracles $H_i (i = 1,2,3)$, partial private query q_{pak}, private key query q_{sk}, public key query q_{pk}, signcrypt query q_s, unsigncrypt query q_u, C answers the above queries, But A_1 can't ask the corresponding private key and partial private key of ID_B^*, also can't unsigncrypt query for (h^*, S^*, c^*, R^*).

At the last, A_1 outputs b' as the guess for b^*, if $b' = b^*$, then C finds the secret value t_B^* in L_{sk} List, By the known $\delta_B^* = 1$, then C can find the corresponding s_B^* in L_{sk} List, asks $(ID_B^*, w_B^*, h_{1B}^*)$ in L_1 List, finds (m_b^*, T^*, h^*, y) in L_2 List, finds the first data for R^* query (R, U) in L_3 List, the final output $\left(\frac{w^*}{(g^b)^{z_B^* h^*}}\right)^{\frac{1}{h_{1B}^*}} = g^x$ as the response to CDH problem., In the private key query phase. The probability of not stop is $\left(1 - \alpha - \frac{1}{q_{rp}}\right)^{q_{sk}}$. In the unsigncrypt phase, only ID_B's private key is unknown or ID_B's public key has been replaced or $\delta_B = 1$, and generate effective signcryption not to ask H_2, H_3, in the process of game is terminated. So the probability for at least $\frac{1}{{q_1}^2 q_u}$ games smoothly run. In the generate challenge ciphertext stage, the probability of termination for α, if A_1 can win by the advantage ε, then C can solve CDH problem by the probability of $\varepsilon' \geq \alpha\left(1 - \alpha - \frac{1}{q_{rp}}\right)^{q_{sk}} \frac{\varepsilon}{{q_1}^2 q_u}$,.

Theorem 2. Let us assume that there exists an IND-CCA2-Ⅱ adversary A_2 has non-negligible advantage ε against our scheme when asking at most q_i queries to random oracles $H_i (i = 1,2,3)$, q_s signcryption queries and q_u unsigncryption queries, Then there is an algorithm C to solve CDH problem with probability $\varepsilon' \geq \frac{\alpha\varepsilon}{{q_1}^2 q_u}$.

Proof: The proof of theorem similar to that of theorem 1, except that A_2 can't ask public key replacement q_{pkr},C should sendparams andmsk to A_2, Set the corresponding public key for challenge ID_B^* to be $u^* = g^a$, the secret value $z (= a)$ is unknown, and set partial challenge ciphertext $R^* = g^b$, the corresponding $r (= b)$ is unknown ,if A_2 output $b' = b^*$, then C output $\left(\frac{w^*}{(g^b)^{t_B^*}}\right)^{\frac{1}{h^*}} = g^{z_B r}$ as the answer to CDH problem.The rest similar to above description in theorem 1.

Theorem 3. If the Schnorr signature is unforgeable ,then in this paper our scheme also is unforgeable.

Proof: As for adversary A_1,the scheme is unforgeable.

If A_1 can't replace the public key w_A, then the process of generating (σ_2, h_2) is for the private key t_A generated message m Schnorr signature process, By Schnorr signature unforgeable known, this forge is can't success.

If A_1 can replace public key w_A, then the process of generating the corresponding new private key t_i also is for master key x generated message (user's identity) Schnorr signature process, this can't also success.

As for adversary A_2 ,this scheme is also unforgeable.

Although A_2 knows the master key mskand partial private key t_A, But A_2 don't know the secret value z_A, and also can't replace public key of any user, So the process of generating (σ_2, h_2) is for the private key z_A generated message Schnorr signature process, Obviously,this can't success.

6. Conclusion

In this paper, user's partial private key and secret value have been used separately, it can resist the public key replacement attack in the literate [7], But the existing scheme as well as this paper proposed scheme are proved security in the random oracle model, The security of our scheme is based on the hardness assumption of CDH problem and DLP problem. How to construct safe and efficient of certificate less signcryption scheme under the standard model is currently a worthy of studying problem.

References

[1] Y. Zheng. Digital signcryption or how to achieve cost (signature & encryption)≪cost (signature) cost(encryption). In Advances in Cryptology-CRYPTO'97, LNCS 1294, p. 165–179, Springer-Verlag, 1997.

[2] S.S. Al-Riyami and K.G. Paterson. Certificateless public key cryptography. In Advances inCryptology-ASIACRYPT 2003, volume 2894 of LNCS , pages 452–473.Springer-Verlag, 2003.
[3] Fan C J. Sun W, Huang V. provably secure randomized blind signature scheme based on bilinear pairing [J]. Computers and Mathematics with Applications,2010,60(2): 285-293.
[4] Xie W J Zhang Z Certificateless Signcryption without Pairing [EB/OL]. [2010-06-20]. http://eprint.ia-cr.org/2010/187.
[5] Zhu H, Li H. Certificateless Signcryption Scheme without pairing.Journal of Computer Research and Development. 2010,47(9): 1587-1594.
[6] Liu WH, Xu CX. Certificateless signcryption scheme without bilinear pairing. Journal of Software, 2011, 22(8): 1918-1926.
[7] Jian Y.Cryptanalysis for two certificateless signcryption schemes. Manufacturing Automation 2013,35(1): 83-85.
[8] Zhou CX Cryptanalysis and improvement of some certificateless signcryption schemes.Computer Engineering and Science.2013,35(8):69-76.
[9] C.P. Schnorr. Efficient identification and signatures for smart cards. In Advancesin Cryptology-CRYPTO’89 Proceedings, volume 435 of LNCS, pages 239–252.Springer-Verlag, 1990.
[10] C.P. Schnorr. Efficient signature generation by smart cards. Journal of Cryptology,4(3):161–174, 1991.
[11] Barbosa M, Farshim P. Certificateless signcryption. In: Proceedings of the 2008 ACM Symposium on Information, Computer and communications security. 2008,661-664.
[12] Gao JX, Wu XP. Secure certificateless signcryption scheme without bilinear pairing. Application Research of Computers.2014,31(4):1194-1198.
[13] Shi W B Neeraj KUMAR, Cryptanalysis and improvement of a certificateless signcryption scheme without bilinear pairing. Higher Education Press and Springer-Verlag Berlin Heidelberg, [2014-02-10].
[14] J.Baek, R. afavi-Nainni, and W. Susilo. Certificateless public key encryption without pairing. In Information Security, volume 3650 of LNCS, pages 134-148. Springer-Verlag, 2005.
[15] Y. Sun, F. Zhang, and J. Baek. Strongly secure certificateless public key encryption without pairing. In Cryptology and Network Security, volume 4856 of LNCS, pages 194-208. Springer-Verlag, 2007.
[16] Li XH. Secure certificateless signcryption scheme without bilinear pairing. Journal of Normal University, 2014 (11) 41-50.