TELKOMNIKA, Vol.14, No.1, March 2016, pp. 335~341
ISSN: 1693-6930, accredited **A** by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v14i1.2919

Certificateless Signature Scheme Based on Rabin Algorithm and Discrete Logarithm

Xiangjun Xin*[1], Chaoyang Li[2]
[1,2] School of Mathematics and Information Science, Zhengzhou University of Light Industry,
Zhengzhou 450002, China
*Corresponding author, e-mail: xin_xiang_jun@126.com[1], lichaoyang2013@163.com[2]

Abstract

Certificateless signature can effectively immue the key escrow problem in the identity-based signature scheme. But the security of the most certificateless signatures usually depends on only one mathematical hard problem, which makes the signature vulnerable when the underlying hard problem has been broken. In order to strengthen the security, in this paper, a certificateless signature whose security depends on two mathematical hard problems, discrete logarithm and factoring problems, is proposed. Then, the proposed certificateless signature can be proved secure in the random oracle, and only both of the two mathematical hard problems are solved, can the proposed signature be broken. As a consequence, the proposed certificateless signature is more secure than the previous signatures. On the other hand, with the pre-computation of the exponential modular computation, it will save more time in the signature signing phase. And compared with the other schemes of this kind, the proposed scheme is more efficient.

Keywords*: certificateless signature, rabin algorithm, discrete logarithm problem*

1. Introduction

In 1984, Shamir first introduced the concept of identity-based public key cryptosystem (ID-PKC) [1], which was used to solve the certificate management problem in traditional public key cryptosystem. In the ID-PKC, the Key Generator Centre (KGC) generates a private key for the accepted user who registers in the KGC with his/her identity, while the user's identity is taken as the corresponding public key. For example, the user's name or email can be used as his/her public key, so the system doesn't need any public key certificate. Therefore, it can efficiently solve the certificate management problem suffered by the traditional public key cryptosystem. But, in the ID-PKC, all the users must trust the KGC, since he/she masters all the users' secret keys. In fact, in the internet, none of the participants can be fully trusted. This means that the malicious KGC can forge any user's signature, since he/she knows all the private keys of the registed users.

In order to solve the key escrow problem existing in ID-PKC, in 2003, the certificateless public key cryptosystem (CL-PKC) was presented [2]. In the CL-PKC, a user's private key consists of two parts, secret value and partial private key. The secret value is master by the user, while the partial private key is shared by both the user and KGC. Then, ID-PKC can avoid KGC having access to a user's private key, so the key escrow problem existing in the ID-PKC can be ingeniously solved. This means that certificateless signature can effectively immue the key escrow problem existing in the ID-based signature. Then, based on the idea in [2], many certificateless signature schemes (CLS) were proposed. Some of the early CLSschemes were not secure. For example, all the schemes proposed in [2-5] were insecure against type I adversary [6]. Because bilinear pairings have good cryptographic properties, some CLS schemes based bilinear pairings were proposed [7-12], also. These pairing-based schemes have been proved to be secure in the formal security model. However, in these pairing-based schemes, heavy pairing operations were required. According to the result in [13], one pairing operation was about 11110 multiplications in finite field F_3^{163}. In 2011, He et al. presented a CLS scheme without bilinear pairings [14]. Generally, it is believed that the efficiency of CLS without pairings will be more efficient than those paring-based schemes. To improve the efficiency of CLS, a number of CLS schemes without pairings were proposed [15-17]. Unfortunately, these

Received October 29, 2015; Revised January 19, 2016; Accepted February 4, 2016

schemes were vulnerable to type II adversary [18, 19]. Recently, Yeh et al. [20] presented a CLS scheme without pairings, and claimed that their scheme was efficient and practical for mobile communication. Nevertheless, in their security proof, the discrete logarithm value x of the given discrete logarithm instance was not derived. Hence, the security proof of his scheme is not sufficient. How to construct a secure and efficient CLS without pairings is still an open problem. On the other hand, different from the constructions of all the CLS schemes discussed above, the CLS proposed in [21] is a classical one based on RSA [22].

It should be noted that the security of all the CLS schemes discussed above is based on only one security problem: discrete logarithm problem, or computing Diffie-Hellman problem, or factoring problem. In this paper, we will present a CLS scheme based on discrete logarithm and factoring problems. That is, only both the two mathematical hard problems are solved can the proposed CLS scheme be broken. We will prove the security of the proposed scheme under formal security model. Compared with the scheme proposed in [21] whose security depends on only one hard problem, the proposed scheme has a better security. Compared with the scheme proposed in [23], the proposed scheme is a certificateless one, which doesn't need any certificate, and it doesn't need to send any public key certificate to the verifier before verifying a signature. On the other hand, with the pre-computation of exponential modular computation, it will save more time in the signature signing phase. Meanwhile, in the signature verification phase, it only needs two exponential operations. In concequence, the proposed CLS scheme is more efficient than similar kinds of schemes.

The rest of this paper is organized as follows. In section 2, we propose our new certificateless scheme. In section 3, the security analysis of the proposed scheme is discussed. In section 4, the efficiency and security comparison between the proposed scheme and the other schemes of this kind is analyzed. At last, in section 5, we conclude.

2. The Certificateless Signature Scheme

2.1 Complexity assumptions

Definition 1(Discrete Logarithm Problem (DLP). Let $N=pq$ be a RSA modular number satisfying $p=2p'+1$, $q=2q'+1$. $g\in Z_N^*$ is a generator of G by order $p'q'$. Given the parameters g, y, N, the discrete logarithm problem is to compute the exponent x such that $y=g^x \bmod N$. Throughout this paper, we assume it is hard to solve DLP.

Definition 2 (Factoring Problem (FP)). Let $N=pq$, where p and q are two large primes with security parameter λ. It is hard to factor N into p and q.

2.2 Our new certificateless signature scheme

Setup: Let p and q be two large primes, where $p=2p'+1$, $q=2q'+1$, and p' and q' are both large primes, too. $N=p\cdot q$ is the modular number. Assume it is hard to factor N. On the other hand, H:$\{0, 1\}^*\rightarrow Z_Q$ and H_1:$\{0, 1\}^*\times\{0, 1\}^*\rightarrow Z_N$ are two secure hash functions, where Z_Q is the set of quadratic residue under modular N.

Partial private key extract: Given the user's identity ID, KGC calculates $h=H(ID) \bmod N$ and $S_{ID}=\sqrt{h} \bmod N$. Note that the square root S_{ID} of h can be computed by Rabin algorithm [24]. Next, KGC returns the partial private key S_{ID} to the user.

Set private key: The user randomly chooses his/her secret value $x\in Z_N$, and adopts the pair (S_{ID}, x) as his/her private key.

Set public key: Given identity ID and the secret value x, the user computes $y=h^x \bmod N$ as his/her own public key, where $h=H(ID)$.

Sign: Given a message m,the user chooses a random number $k\in Z_N$ and compute

$$r=(S_{ID})^k \bmod N,\ l=H_1(m, r) \text{ and } t=k+xl. \quad (1)$$

The signature on the message m is $\sigma=(t, r, y)$.

Verify: Given the message m and the signature $\sigma=(t, r, y)$, the verifier computes

$h=H(ID)$ and $l=H_1(m, r)$. Then, he/she verifies whether the equation

$$h^t=r^2y^l \bmod N \quad (2)$$

holds. If it holds, the verifier accepts the signature, or he/she refuses.

The correctness of the proposed scheme is given below.

$$\begin{aligned} h^t &= h^{k+xl} \bmod N \\ &= h^k \cdot h^{xl} \bmod N \quad (S_{ID} = \sqrt{h} \bmod N) \\ &= (S_{ID}^2)^k \cdot (h^x)^l \bmod N \quad (r = (S_{ID})^k \bmod N, y = h^x \bmod N) \\ &= r^2 y^l \bmod N \end{aligned}$$

3. Security Analysis

In general, for a certificateless signature, two kinds of adversaries with different capabilities need to be considered [2]. Type I adversary acts as a dishonest user, who can replace any entity's public key with unique value, but he/she does not know the master secret key of KGC. Type II adversary acts a malicious KGC, who has an opposite capability that he/she knows the master secret key of KGC but cannot replace any entity's public key. In addition, in [3], Type I and Type II adversaries are also classified into three categories: normal, strong and super levels. These three categories of adversaries have different capabilities.

Normal: The normal-level Type I (or II) adversary has access to the valid signatures.

Strong: The strong-level Type I (or II) adversary can replace the public key of legal users, then forge a valid signature when the adversary possesses a corresponding secret value.

Super: The super-level Type I (or II) adversary has the ability to learn valid signatures for a replaced public key without any submission. In this paper, we take the super-level adversary into consideration in our security analysis.

By the discrete logarithm and factoring assumptions, we can prove that our signature is secure against the super-level Type I adversary and Type II adversary. Before presenting the security poof of the proposed scheme, we first review the forking lemma [25].

Lemma 1. [Forking Lemma][25] In the random oracle mode, for a generic signature scheme, let F be a Turing machine whose input only consists of public data. Assume that F can produce a valid signature (m, σ_1, h, σ_2) within a time bound T by un-negligible probability $\varepsilon \geq 10(n_s+1)(n_h+n_s)/q$, where n_h and n_s are the numbers of queries that F can ask to the random oracle and the signing oracle respectively. If the triple (σ_1, h, σ_2) can be simulated without knowing the secret key, with an indistinguishable distribution probability, then there is another machine which has control over the machine obtained from F replacing the signing oracle by simulation and produces two valid signatures (m, σ_1, h, σ_2) and (m, σ_1, h', σ'_2) such that $h \neq h'$ in the expected time less than $120686 \cdot n_h \cdot T/\varepsilon$.

Theorem 1. In the random oracle mode, for our CLS scheme, let Type I adversary have a polynomial-time algorithm α that can produce a valid signature (t, r, y) such that $h^t = r^2 y^l \bmod N$ within a time bound T by un-negligible probability $\varepsilon \geq 10(n_s+1)(n_h+n_s)/q$, where n_h is the number of queries that α can ask to the random oracle H_1, and n_s is the number of queries that α can ask to the signing oracle. If the signature (t, r, y) can be simulated without knowing the private key, with an indistinguishable distribution probability, then there is another machine which can solve the Discrete Logarithm Problem and Factoring Problem in the expected time less than $120686 \cdot n_h \cdot T/\varepsilon$.

Proof. Suppose Type I Adversary has a polynomial-time algorithm α that can break the proposed scheme with non-negligible advantage ε, and H, H_1 are two random oracles. Let β be the challenger.

In the initialization phase, the system setups the public parameters, the modular number N, the public key y, and the hash function H and H_1, all of which are the same as those described in section 2. Now, β's goal is to solve both hard problems: discrete logarithm problem and factoring problem. He/she tries to compute out discrete logarithm of y to the base h and to factor the RSA modular N into prime numbers p and q.

In the Query phase, the following oracle queries are adaptively issued by α, and each query is unique.

***H* Query:** Upon receiving an H query for the identity ID_i from α, β checks the list $list_H$ maintained by himself/herself, and returns h_i to α. The detailed steps are as follows.

If (S_{ID_i} , h_i, ID_i) already exists in the list $list_H$, β directly returns h_i to α. Otherwise, β randomly chooses $S_{ID_i} \in Z_N$, computes h_i= $S_{ID_i}^2 \bmod N$ and sets $H(ID_i)$:= h_i. Then, he/she returns h_i to α. After that, the triple (S_{ID_i} , h_i,ID_i) is added into the list $list_H$.

***H_1* Query:** Upon receiving a H_1 query for the pair (m, r_i) from α, β checks the list list_{H_1} maintained by himself/herself. Then he/she returns l_i to α. The detailed steps are as follows.

If (l_i, m, r_i) already exists in the list list_{H_1}, β directly returns l_i to α. Otherwise, β randomly chooses a number $l_i \in Z_N$. Then, he/she sets $H_1(m, r_i)$:= l_i and returns l_i to α. After that, β adds(l_i, m, r_i) into the list list_{H_1}.

Partial Private Key Extract Query: Upon receiving a query for the partial private key of the user with identity ID_i, β executes the H Query, and returns S_{ID_i} to α. In this phase, Type I adversary cannot query the partial private key of the target user with identity ID.

Private Key Query: Upon receiving a query for the private key of the user with identity ID_i, β queries the *Partial Private Key Extract Oracle* and gets the partial private key S_{ID_i} . Then, he/she randomly chooses a number $x_i \in Z_N$ and returns the pair (S_{ID_i} , x_i) to α.

Public Key Query: Upon receiving a query for the public key of the user with identity ID_i, β queries the random oracle H and Private Key Oracle. So, β can get (S_{ID_i} , h_i, ID_i,x_i). Then, β computes y_i= $h_i^{x_i} \bmod N$ and returns y_i to α.

Public Key Replacement: Upon receiving a query for the public key of the user with identity ID_i and public key y_i', β simulates the Public Key Query for the identity ID_i and replace y_i' with y_i. This replacement will be recorded by β.

Signing Query: Upon receiving a signing query for a user′s identityID_i on some message m, β executes the Private Key Query to obtain the pair (S_{ID_i} , x_i) and h_i. Then, he/she randomly chooses a number $k \in Z_N$andcomputes r_i=(S_{ID_i})^{k}mod$N$. Next, β executes the algorithm H_1 Query and obtains the triple (l_i, m, r_i) from the list_{H_1}. So, he/she computes t_i=k+$x_i l_i$. At last, β returns the signature σ_i=(t_i, r_i, y_i) to α, where y_i= $h_i^{x_i} \bmod N$. It is easy to verify that σ_i can pass the signature verification.

From the simulation, the challenger can successfully answer all the queries without being detected. Then, the algorithm α believes that he/she has successfully attacked the proposed scheme. Note that in forking lemma, l is the hash value of (m, r). And l only depends on m and r. Then, β can simulates another machine by using the forking lemma, and produces two valid signatures (t, r, y) and (t*, r, y) for the target user with identity ID and public key y. From equation (2), we have

$$h^t = r^2 y^l \bmod N \qquad (3)$$

$$h^{t*} = r^2 y^{l*} \bmod N , \qquad (4)$$

where$h$ can be seemed as the identity of the garget user with identity ID and public key y. From the equation (1), (3) and (4), we can get

$$t=k+xl, \qquad (5)$$

$$t^*=k+xl^*, \qquad (6)$$

where$l$≠l*. Therefore, from equations (5) and (6), we can derive

$$x = \frac{t - t*}{l - l*}.$$

Then, the discrete logarithm of *y* to the base *h* can be computed. On the other hand, from equations (3) and (4), we have

$$r^2 = h^t y^{-l} \bmod N, \tag{7}$$

$$r^2 = h^{t^*} y^{-l^*} \bmod N, \tag{8}$$

Then, if all the numbers t^*, t, l^*, l are even numbers, we can also get

$$r^2 = \left[h^{\frac{t}{2}} y^{\frac{-l}{2}} \right]^2 = \left[h^{\frac{t^*}{2}} y^{\frac{-l^*}{2}} \right]^2 \bmod N \tag{9}$$

from equations (7) and (8). Therefore, we can factor the RSA modular *N* by computing

$$\gcd(h^{\frac{t}{2}} y^{-\frac{l}{2}} + h^{\frac{t^*}{2}} y^{-\frac{l^*}{2}}, N) \text{ or } \gcd(h^{\frac{t}{2}} y^{-\frac{l}{2}} - h^{\frac{t^*}{2}} y^{-\frac{l^*}{2}}, N)$$

From Theorem 1, it is easy to get Theorem 2 as follow.

Theorem 2. The proposed certificateless signature scheme can achieve existential unforgeability against a super-level Type I adversary in the random oracle model. Only both of the discrete logarithm and factoring problems are solved can the proposed signature scheme be broken.

Similarly, we have Theorem 3 as follow.

Theorem 3. The proposed certificateless signature scheme can achieve existential unforgeability against a super-level Type II adversary in the random oracle model. Only both of the discrete logarithm and factoring problems are solved can the proposed signature scheme be broken.

Proof. By using the proof similar to that of Theorem 2, we can prove that our CLS scheme can achieve existential unforgeability against a super-level Type II adversary. Note that Type II adversary acts as a malicious KGC. Hence, a super-level Type II adversary can query the partial private key of any user, including the target user. Then, the difference is that in theorem 3, the super-level Type II adversary cannot mount a public key replacement attack to the target user, since this kind of adversary can query the partial private key of the target user.

4. Efficiency and Security Comparison

In this part, we compare the proposed scheme with the similar schemes of this kind [21, 23]. First, we make a security comparison among the similar schemes. The security of scheme proposed in [21] depends on only one hard problem. From the security proof in [21], we find the signature in [21] will be broken in case that either RSA problem or discrete logarithm problem is broken, while only both the two mathematical hard problems, discrete logarithm and factoring problems, are broken can the proposed signature can be forged. Then, compared with the scheme in [21], the proposed scheme has a better security. The security of the scheme proposed by Verma and Sharma [23] is also based on discrete logarithm and factoring problems. However, the security of their scheme cannot be proved in the formal security model. What is more, their scheme is a certificate-based one, in which the security of key management has to be considered. On the other hand, before verifying the signature in [23], the public key certificate should be transmitted to the verifier.

Table 1.Security Comparison of the Similar Schemes

Schemes	CLS	Security level and underlying problem	Formal security proof
Scheme [21]	Yes	I level: RSA Problem or DLP	Yes
Scheme [23]	No	II level: DLP and FP	No
The proposed scheme	Yes	II level: DLP and FP	Yes

I level: The security of the signature depends on only one hard problem.

II level: The security of the signature depends on two hard problems. Only both of the underlying hard problems are solved can the signature be broken.

DLP: Discrete Logarithm Problem.

FP: Factoring Problem.

Now, we compare the efficiency of the similar schemes. In a signature scheme, compared with the other operations under modular, the exponential operation is more time-consuming. So, for a signature scheme, the fewer exponential operations should be used.Then, we mainly compare the numbers of exponential operation and hash operation among the similar schemes. From the comparison in Table 2, it is found that the proposed scheme has the fewest exponential operations. Then, compared with the similar kinds of schemes, the proposed CLS scheme is more efficient. In fact, in the proposed scheme, the parameter *r* in the signing phase can be pre-computed and stored before signing a signature. Hence, the exponential operation in the signing phase of the proposed scheme can be ignored. This means that it will save much time to sign a signature, since it can reduce one or more exponential operations when the pre-computation is ignored.

Table 2.Efficiency Comparison of the Similar Schemes

Schemes	Signing phase		Verification phase	
	exponential operation	hash operation	exponential operation	hash operation
scheme[21]	3	1	4	2
scheme [23]	2	1	4	1
The proposed scheme	1	1	2	2

5. Conclusion

Due to the virtues of CLS, manycertificateless signatures have been proposed. However, the security of most CLS schemes depends on only mathematical hard problem. In this paper, we present a new CLS scheme, whose security depends on two complexity assumptions: DLP and FP. That is, only both DLP and FP are broken can our signature be forged. On the other hand, we show the proposed scheme has the fewest exponential operations during both signing phase and verifying phase. Therefore, compared with the others similar schemes, the proposed scheme is more secure and efficient.

(1) As shown in the Table 1, the proposed scheme is a CLS one, and it has II security level. And we present the formal security proof for the proposed scheme in the security model. Then, compared with the schemes in [22, 23], the proposed scheme is more secure.

(2) As shown in the Table 2, the proposed scheme has the fewest exponential operations during both the signing phase and verifying phase. Especially, with the pre-computation of exponential modular computation, it will save much time in signature signing phase. Then, for the proposed scheme, it will be more efficient to sign and verify a signature.

ACKNOWLEDGEMENTS

This work is supported by the Natural Science Foundation of China (Grant No. 61272525), the Foundation for Doctors of Zhengzhou University of Light Industry (NO. 20080014) and the Fundamental and Advanced Technology Research Project of Henan province (Grant No. 152300410129).

REFERENCES

[1] Shamir A. *Identity-based cryptosystem and signature schemes*. Proceedings of CRYPTO'84, G.R. Blakley and D. Chaum (Eds.), Paris France. 1984; LNCS196: 47-53.

[2] Al-Riyami S, Paterson KG. *Certificateless public key cryptography*. Proceedings of ASIA CRYPT 2003, Chi-Sung Laih (Ed.), Taipei, Taiwan. 2003; LNCS 2894: 452-473.

[3] Yum DH, Lee PJ. *Generic construction of certificateless signature*. Proceeding of 9th Australasian Conference on Information Security and Privacy, Wang H., et al.(Eds.), Sydney, Australia. 2004; LNCS 3108: 200-211.

[4] Huang X, Mu Y, Susilo W, Wong DS, Wu W. *Certificateless signature revisited*. Proceedings of ACISP 2007, Josef Pieprzyk, HosseinGhodosi, Ed Dawson (Eds.), Townsville, Australia. 2007; LNCS 4586: 308-322.

[5] Huang WS, Susilo W, Mu Y, Zhang F. *On the security of certificateless signature schemes from Asiacrypt 2003*. InCANS2005, Desmett Y. G., Wang H., Mu Y., Li Y.(Eds.), Shenzhen, China.2005; LNCS 3810: 13-25.

[6] Hu BC, Wong DS, Zhang Z, Deng X. *Key replacement attack against a generic construction of certificateless signature*. Proceedings of ACISP, Lecture Notes in Computer Science. 2006; 4058: 235-246.

[7] Li X, Chen K, Sun L. Certificateless signature and proxy signature schemes from bilinear pairings. *Lithuanian Mathematical Journal*. 2005; 45(1): 76-83.

[8] Cheng L, Wen Q, Jin Z. Cryptanalysis and improvement of a certificateless aggregate signature scheme. *Information Sciences*. 2015; 295: 337-346.

[9] Yuan Y, Wang C. Certificateless signature scheme with security enhanced in the standard model. *Information Processing Letters*. 2014; 114 (9): 492-499.

[10] Du H, Wen Q. Certificateless proxy multi-signature. *Information Sciences*. 2014; 276: 21-30.

[11] Tso R, Yi X, Huang X. Efficient and short certificateless signatures secure against realistic adversaries. *Journal of Supercomputing*. 2011; 55: 173–191.

[12] Horng S, Tzeng SF, Huang PH. An efficient certificateless aggregate signature with conditional privacy-preserving for vehicular sensor networks. *Information Sciences*. 2015; 317: 48-66.

[13] Zhang F, Safavi-Naini R, Susilo W. *An efficient signature scheme from bilinear pairings and its applications*. In Public Key Cryptography - PKC 2004, Singapore. 2004;1: 277-290.

[14] He D, Chen J, Zhang R. An efficient and provably-secure certificateless signature scheme without bilinear pairings. *International Journal of Communication Systems*. 2012, 25(11): 1432-1442.

[15] He D, Chen J, Hu J. A pairing-free certificateless authenticated key agreement protocol. *Int J Commun Syst*. 2012; 25: 221-230.

[16] Gong, Li P. Further improvement of a certificateless signature scheme without pairing. *Int J Commun Syst*. 2012; 10: 2450-2457.

[17] Swati V, Birendra KS, A new proxy blind signature scheme based on DLP, *International Journal of Information & Network Security*. 2012; 1(2): 60-66.

[18] Tian M, Huang L. Cryptanalysis of a certificateless signature scheme without pairings. *International Journal of Communication Systems*. 2013; 26(11): 1375-1381.

[19] Tsai JL, Lo NW, Wu TC. Weaknesses and improvements of an efficient certificateless signature scheme without using bilinear pairings. *International Journal of Communication Systems*. 2014; 27(7): 1083-1090.

[20] Yeh KH, Tsai KY, Fan YC. An efficient certificateless signature scheme without bilinear pairings. *Multimedia Tools and Applications*. 2014; 10: 1007-1015.

[21] Zhang JH, Mao J. An efficient RSA-based certificateless signature scheme. *The Journal of Systems and Software*. 2012; 85: 638-642.

[22] Rivest TL, Shamir A, Adleman L. A method for obtaining digital signatures and public key cryptosystems. *Communications of the ACM*. 1978; 3: 120-126.

[23] Verma S, Sharma BK. A new signature scheme based on factoring and discrete logarithm problems, *International Journal of Information & Network Security*. 2012;1(3): 158-162.

[24] Rabin MO. Digitalized signatures and public key function as intractable as factoring. *MIT/LCS/TR*. 1979; 5: 212-213.

[25] Pointcheval D, Stern J. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*. 2000; 13(3): 361−369.