A New Designated Verifier Proxy Signature in Standard Model

Li-Hong Guo, Hai-Tao Wu
Dept. of Communications Engineering, Nanjing Institute of Technology Nanjing, China
E-mail: guolihongnj@163.com, wuhaitao@njit.edu.cn

Qing Yang
Dept. of Computer Engineering, Nanjing Institute of Technology Nanjing, China
E-mail: yangq@njit.edu.cn

***Abstract*-Proxy signature schemes have been suggested for use in a number of applications, so the security of proxy signature scheme is more and more important. However, at present, almost all the proxy signature schemas were proven secure in the random oracle model, which has received a lot of criticism. Recently, Yu et al. proposed a designated verifier proxy signature scheme without random oracles by using Waters hashing technique. The formal models and a strictly logical process were provided in this schema. But Kang et al. show some attacks on Yu et al.s scheme and offer its insecurity proof. In this paper, in order to overcome the weaknesses in Yu et al.s scheme we make supplements on it and make it secure resist Kang et al.s attacks.**

Keywords-designated verifier proxy signature; security; attacks; standard model

I. INTRODUCTION

At present, many proxy signature schemes [1, 2, 3] have been proposed and suggested for use in many fields. Such as proxy blind signature [4], proxy ring signature [5], proxy multi-signature [6], and so on.

In 2003, Dai et al. [7] first introduce the concept of designated verifier proxy signature, which allows the proxy signer to convince the designated receiver that he has signed the specific message while protecting his signing privilege from knowing by other parties. In this scheme, it provides authentication of a message without providing a non-repudiation property of traditional digital signature. This kind of signature is very useful in electronic commerce applications. However, as far as a designated verifier proxy signature is concerned, provable security is very essential.

In 2004, G. Wang et al. [8] pointed out that Dai el al.s scheme is not secure by identifying a forgery attack. In this attack, the original signer alone can forge valid proxy signatures to frame the proxy signer. Later, several designated verifier proxy signature schemes were proposed [9, 10, 11]. However, most of them have received a lot of criticism since what they provide the security proofs in the random oracle model are not sound with respect to the standard model. Recently, Yu et al. [12] proposed a new designated verifier proxy signature scheme based on Waters hashing technique [13]. They claimed that the new construction is the first designated verifier proxy signature secure, whose security does not rely on the random oracles. Unfortunately, Kang et al.s scheme [14] shows some attacks on Yu et al.s scheme. So, their scheme is also not secure. In this paper, we propose a new designated verifier proxy signature scheme in standard model, which can face with the attacks proposed in Kang et al.s literature.

The rest of this paper is organized as follows. In the next section, some preliminary works are given. The formal models of designated verifier proxy signature is described in section 3. In section 4, the new designated verifier proxy signature scheme in the standard model is described. In section 5, we analyze the new scheme. Finally, we conclude the scheme.

II. PRELIMINARIES

In this section, we will provide some preliminaries knowledge used in this paper, which is described in the literature [12].

A. Standard Model

The standard model is defined in the design process without the aid of any hypothetical model, its security is only based on the difficult problem that has been recognized. Therefore the provably secure scheme in the standard model is more worthy of people's trust.

B. Random oracle Model

Before the digital signature scheme is worked in the message, the message usually uses the Hash operation, and by the hash to compression message length. If for any input and output hash value operation and output function of spatially homogeneous distribution in the calculation is not distinguishable, then you can think Hash is a random function Oracle, that is to say, in the random oracle model, the Hash function is formalized as an oracle, and the Oracle can have a completely random output for each query. The example is shown in Figure 1.

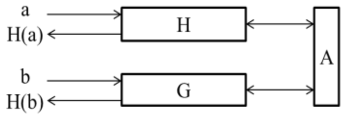

Figure 1. An example about random oracle model.

C. Bilinear Pairings

Let G_1 be a cyclic additive group and G_2 be a cyclic multiplicative group of the same large prime order q. We also assume that the discrete logarithm problems (DLP) in both G_1 and G_2 are hard to solve.

A map $\hat{e}: G_1 \times G_1 \to G_2$ is an admissible bilinear map if it satisfies the three following properties:

- Computable: $\forall P,Q \in G_1$, there exists an efficient algorithm to compute $\hat{e}(P,Q)$;
- Bilinear: $\forall P,Q \in G_1$ and $\forall a,b \in Z_q^*$, we have $\hat{e}(aP,bQ) = \hat{e}(P,Q)^{ab}$;
- Non-degenerate: $\exists P,Q \in G_1$, $\hat{e}(P,Q) \neq 1$.

D. Complexity Assumptions

Definition 1. Computational Diffie-Hellman (CDH) Problem in G_1.

Given $g, g^a, g^b \in G_1$ for some unknown $a,b \in Z_p$, compute $g^{ab} \in G_1$.

The success probability of a polynomial algorithm A in solving the CDH problem in G_1 is denoted: $Succ_{A,G_1}^{CDH} = \Pr[A(g,g^a,g^b) = g^{ab} : a,b \in {}_R Z_p]$.

Definition 2. Computational Diffie-Hellman (CDH) Assumption in G_1.

Given $g, g^a, g^b \in G_1$, for some unknown $a,b \in Z_p$, $Succ_{A,G_1}^{CDH}$ is negligible.

III. FORMAL MODELS OF DESIGNATED VERIFIER PROXY SIGNATURE

In this section, we will describe the outline and the security requirements of the new scheme.

A. Outline of Designated Verifier Proxy Signature

There exist three participants in a designated verifier proxy signature, namely Alice, Bob and Cindy, who act as the original signer, the proxy signer and the designated verifier respectively. A designated verifier proxy signature consists of the following algorithms.

1) *Setup*

Given a security parameter k, this algorithm outputs the systems parameters.

2) *Keygen*

It takes as input the security parameter k and outputs the secret-public key pair (sk_i, pk_i) for $i \in \{a,b,c\}$ denotes Alice, Bob and Cindy respectively.

3) *Delegationgen*

Given the system's parameter, the original signer's private key and the warrant W to be signed, this algorithm outputs the delegation σ_w.

4) *Proxysign*

This algorithm takes as input the proxy signer's private key sk_b, the delegation σ_w, the designated verifier's public key pk_c and a message m to generate a signature σ.

5) *Verify*

A deterministic algorithm that accepts a message m, the warrant W, a signature σ_m, the original signer and the proxy signer's public key (pk_a, pk_b), the designated verifier's private key sk_c and returns T if the signature is valid, otherwise returns $\perp$ indicating the signature is invalid.

6) *Transcript simulation*

An algorithm that accepts a message m, a warrant W and the verifier's private key sk_c is to produce an identically distributed transcript σ^* that is indistinguishable from the original designated verifier proxy signature σ.

B. Security Notions

There are four types adversaries involved in the system.

Type 1: adversary A1 only has the public keys of Alice and Bob. He aims to forge the original signers standard signature or to forge the proxy singers proxy signature.

Type 2: adversary A2 has the public keys of Alice and Bob, he additionally has the secret key of the original signer Alice. He aims to forge the proxy singers proxy signature.

Type 3: adversary A3 has the public keys of Alice and Bob, he additionally has the secret key of the proxy signer Bob. He wants to forge the original signers standard signature.

Type 4: adversary A4 only has the public key of Cindy. He wants to identify the validity of the proxy signature.

From the analysis of adversaries type, we can find that if a designated verifier proxy signature scheme is unforgeable against the adversary in Type 2 and Type 3, it is also unforgeable against the adversary in Type 1. In a warrant-based proxy signature scheme, the delegation is original signer's standard signature on the warrant, which contains proxy's public key, a period of validity. The restrictions on the messages that the signer can sign and so on. Therefore, this kind of proxy signature can prevent the misuse of the delegation.

IV. A NEW DESIGNATED VERIFIER PROXY SIGNATURE SCHEME

In the setup phase, ParamGen algorithm and KeyGen algorithm is computed as follows:

1) *ParamGen*

Let G_1, G_T be two cyclic groups of order q and q is a prime number. Parameter g and g1 are the generators of G_1. Parameter e denotes the bilinear pairing map $G_1 \times G_1 \to G_T$. The master public parameters are (g , G_1, G_T , e , q).

2) *KeyGen*

The original signer Alice randomly chooses $x_a, y_a \in Z_q$ to compute the corresponding public key $u_a = g^{x_a}$ and $v_a = g^{y_a}$. Similarly, for the proxy signer Bob, he also randomly selects $x_b, y_b \in Z_q$ to produce the corresponding public key $u_b = g^{x_b}$ and $v_b = g^{y_b}$. The designated verifier also randomly selects $x_D, y_D \in Z_q$ to produce the corresponding public key $u_D = g^{x_D}$ and $v_D = g^{y_D}$.

3) *Delegation*

Let W denote a delegated warrant, which includes proxy signers identity, deadline, and so on. To produce a delegation of warrant W, the original signer Alice computes as follows:

- Randomly choose r_1 , $r_a \in_R Z_q$ to compute $x_a + Wy_a + r_a$. If $x_a + Wy_a + r_a$ is not a quadratic residue modulo q , then try again with a different value r_a .
- Then compute $\sigma_{\omega 1} = g_1^{x_a + Wy_a + r_a}(u'\prod_{i \in W} u_i)^{r_1}$ $\sigma_{\omega 2} = g^{r_1}$ $\sigma_{\omega 3} = g^{r_a}$
- And send ($\sigma_{\omega 1}$, $\sigma_{\omega 2}$, $\sigma_{\omega 3}$) to proxy signer Bob.
- On receiving ($\sigma_{\omega 1}$, $\sigma_{\omega 2}$, $\sigma_{\omega 3}$), proxy signer verifies whether the following equation $e(\sigma_{\omega 1}, g) = e(u_a v_a^W \sigma_{\omega 3}, g_1) e(u'\prod_{i \in W} u_i), \sigma_{\omega 2})$ holds. If it holds, then ($\sigma_{\omega 1}$, $\sigma_{\omega 2}$, $\sigma_{\omega 3}$) is acted as the signing key of proxy signer.
- ProxySign: Let M be a 160-bit message in the admission range of warrant W. Otherwise, we can adopt a suitable collision resistant hash function to hash the message to 160bits. To generate a signature Sig on the message M with ($\sigma_{\omega 1}$, $\sigma_{\omega 2}$, $\sigma_{\omega 3}$) and secret key (x_b , y_b), the proxy signer computes as follows:
- Randomly chooses r_1' , r_2 , $r_b \in_R Z_q$ to compute $x_b + Wy_b + r_b$. If $x_b + Wy_b + r_b$ is not a quadratic residue modulo q , then we try again with another $r_b \in_R Z_q$.
- Then computes

  $$\sigma_1 = \sigma_{\omega 1}(u'\prod_{i \in W} u_i)^{r_1'} g_1^{x_b + My_b + r_b}(m'\prod_{j \in M} m_j)^{r_2}$$

  $$\sigma_2 = u_D v_D = g^{x_D + y_D}$$

  $$\sigma_3 = g^{r_a + r_b}$$

  $$\sigma_4 = g^{(r_1 + r_1')}$$

- $\sigma_5 = g^{r_2}$
- The resultant proxy signature on message M is (σ_1 , σ_2 , σ_3 , σ_4 , σ_5 W).

4) *Verify*

Given a proxy signature (σ_1 , σ_2 , σ_3 , σ_4 , σ_5 W) on message M, a verifier first checks whether M belongs to the admission ranger of W. If it is valid, then it verifies as follows:

$$e(\sigma_1, \sigma_2) = e(u_a v_a^W u_b v_b^M \sigma_3, g_1^{x_D + y_D}) * e((u'\prod_{i \in W} u_i), \sigma_4^{(x_D + y_D)}) \\ * e((m'\prod_{j \in M} m_j), \sigma_5^{(x_D + y_D)}) \quad (1)$$

If the above equation (1) holds, then the result returns true; otherwise, the result returns false.

V. ANALYSIS OF THE SCHEME

In this section, we will firstly show the correctness of our scheme. Then we prove that our scheme is secure against all types of adversaries.

A. *Correctness*

The correctness of the scheme can be directly verified by the following equation (2):

$$\begin{aligned} & e(\sigma_1, \sigma_2) \\ & = e(\sigma_{\omega 1}(u'\prod_{i \in W} u_i)^{r_1'} g_1^{x_b + My_b + r_b}(m'\prod_{j \in M} m_j)^{r_2}, g^{x_D + y_D}) \\ & = e(g_1^{x_a + Wy_a + r_a}(u'\prod_{i \in W} u_i)^{r_1 + r_1'} g_1^{x_b + My_b + r_b}(m'\prod_{j \in M} m_j)^{r_2}, g^{x_D + y_D}) \\ & = e(g_1^{x_a + Wy_a + r_a}, g^{x_D + y_D}) e((u'\prod_{i \in W} u_i)^{r_1 + r_1'}, g^{x_D + y_D}) e(g_1^{x_b + My_b + r_b}, g^{x_D + y_D}) e((m'\prod_{j \in M} m_j)^{r_2}, g^{x_D + y_D}) \\ & = e(u_a v_a^W g^{r_a}, g_1^{x_D + y_D}) e((u'\prod_{i \in W} u_i)^{x_D + y_D}, g^{r_1 + r_1'}) e(u_b v_b^M g^{r_b}, g_1^{x_D + y_D}) e((m'\prod_{j \in M} m_j), g^{r_2 (x_D + y_D)}) \\ & = e(u_a v_a^W g^{r_a} u_b v_b^M g^{r_b}, g_1^{x_D + y_D}) e((u'\prod_{i \in W} u_i), g^{(r_1 + r_1')(x_D + y_D)}) e((m'\prod_{j \in M} m_j), g^{r_2 (x_D + y_D)}) \\ & = e(u_a v_a^W u_b v_b^M g^{r_a + r_b}, g_1^{x_D + y_D}) e((u'\prod_{i \in W} u_i), g^{(r_1 + r_1')(x_D + y_D)}) e((m'\prod_{j \in M} m_j), g^{r_2 (x_D + y_D)}) \end{aligned} \quad (2)$$

B. *Resistance One Kangs Attacks*

From our scheme we can see that, if the proxy signature is verified successfully, A4 has got the value of xD+yD, which has been thought as impossible. We just omit it.

Our scheme is based on Sun et als scheme [15], which has been proved its security on the property of unforgeability. We just analyze the resistance on Kangs attacks.

Attack 1 On receiving the delegation ($\sigma_{\omega 1}$, $\sigma_{\omega 2}$, $\sigma_{\omega 3}$) and the warrant W, the attacker randomly selects $r_1^* \in Z_P$ and alters the delegation $\sigma_{\omega 1}$ as $\sigma_{\omega 1}^*$, it is equation (3).

$$\begin{aligned} \sigma_{\omega 1}^* & = \sigma_{\omega 1}(u'\prod_{i \in W} u_i)^{r_1^*} \\ & = g_1^{x_a + Wy_a + r_a}(u'\prod_{i \in W} u_i)^{r_1}(u'\prod_{i \in W} u_i)^{r_1^*} \\ & = g_1^{x_a + Wy_a + r_a}(u'\prod_{i \in W} u_i)^{r_1 + r_1^*} \end{aligned} \quad (3)$$

While receiving ($\sigma_{\omega 1}$, $\sigma_{\omega 2}$, $\sigma_{\omega 3}$), proxy signer verifies whether the following equation $e(\sigma_{\omega 1}, g) = e(u_a v_a^W \sigma_{\omega 3}, g_1) e(u'\prod_{i \in W} u_i), \sigma_{\omega 2})$ holds. Then, attacker have to make $\sigma_{\omega 2}^* = g^{r_1 + r_1^*}$, which has solved the CDH problem.

Attack 2 On receiving the proxy signature σ on one message M, attacker make a false signature σ_1^* on message M.

$$\begin{aligned} \sigma_1^* & = \sigma_1 \cdot (m'\prod_{j \in M} m_j)^{r_2^*} \\ & = \sigma_{\omega 1}(u'\prod_{i \in W} u_i)^{r_1'} g_1^{x_b + My_b + r_b}(m'\prod_{j \in M} m_j)^{r_2 + r_2^*} \end{aligned} \quad (4)$$

Let σ_1^* be equation (4), in order to verify whether the following equation (5):

$$e(\sigma_1,\sigma_2)=e(u_a v_a^W u_b v_b^M \sigma_3, g_1^{x_D+y_D})*e((u'\prod_{i\in W}u_i),\sigma_4^{(x_D+y_D)})$$
$$*e((m'\prod_{j\in M}m_j),\sigma_5^{(x_D+y_D)}) \quad (5)$$

Let equation (5) hold, attacker have to make $\sigma_5 = g^{r_2+r_2^*}$. If the attacker does it successfully, it solved the CDH problem.

Attack 3 and attack 4

From the scheme, we can see that if our scheme can resist attack 1 and attack 2, naturally it can resist attack 3 and attack 4.

VI. CONCLUSIONS

In this paper, we have presented a new designated verifier proxy scheme in the standard model, which is the supplement of Yu et al.s scheme. The formal models and proofs were provided to prove its correctness and unforgeability in this scheme, which can be proven secure in the standard model. More importantly, it can face with the attacks of Kang et al.s scheme.

ACKNOWLEDGMENT

This work was supported by the Scientific Research Funding of Nanjing Institute of Technology (No.ZKJ201511), the Innovation Funding of Nanjing Institute of Technology (No.CAJA201306), and the Innovation Funding of Nanjing Institute of Technology (No. CAJA201506).

REFERENCES

[1] M. Mambo, K. Usuda, E. Okamoto, Proxy signature: Delegation of the power to sign messages, *IEICE Transactions on Fundamentals*, vol. 79, pp. 1338-1353, 1996.

[2] G. Wang, F. Bao, J. Zhou, Robert H. Deng, Security analysis of some proxy signatures, *in: ICICS 2003, in: LNCS*, Springer-Verlag, Berlin, 2003, pp. 305-319.

[3] X. Huang, Y. Mu, W. Susilo, W. Wu, Proxy signature without random oracles, in: MSN 2006, in: LNCS, Springer-Verlag, Berlin, 2006, pp. 473-484.

[4] W. Lin, J. Jan, A secure personal learning tools using a proxy blind signature scheme, *in Proceedings of International Conference on Chinese Language Computing*, Illinois, USA, 2000, pp. 273-277.

[5] F. Zhang, R.S. Naini, C. Lin, Some new proxy signature schemes from bilinear pairings, *Progress on Cryptography: 25 Years of Cryptography in China*, in: Kluwer International Series in Engineering and Computer Science, 2004, pp. 59-66.

[6] L. Yi, G. Bai, G. Xiao, Proxy multi-signature scheme: A new type of proxy signature scheme, *Electronic Letters*, vol. 36, pp. 527-528, 2000.

[7] J. Z. Dai, X. H. Yang, and J. X. Dong, Designated-receiver proxy signature scheme for electronic commerce, *in Proceedings of IEEE International Conference on Systems*, Man and Cybernetics, 2003, pp. 384389.

[8] G. Wang, Designated-verier proxy signatures for e-commerce, *in Proceedings of IEEE International Conference on Multimedia and Expo*, 2004, pp. 1731-1734.

[9] X. Huang, Y. Mu, W. Susilo, and F. Zhang, Short designated verier proxy signature from pairings, *in Proceedings of the International Conference on Embedded and Ubiquitous Computing Workshops*, 2005, pp. 835844.

[10] R. X. Lu and Z. F. Cao, Designated verier proxy signature scheme with message recovery, *Applied Mathematics and Computation*, vol. 169, pp. 12371246, 2005.

[11] J. Zhang and J. Mao, A novel ID-based designated verier signature scheme, *Information Sciences*, vol. 178, pp. 766773, 2008.

[12] Y. Yu, C. Xu, X. Zhang, and Y. Liao, Designated verier proxy signature scheme without random oracles, *Computers and Mathematics with Applications*, vol. 57, pp. 13521364, 2009.

[13] B. Waters, Efficient identity-based encryption without random oracles, *in Proceedings of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques: Advances in Cryptology*, 2005, pp. 114127.

[14] Baoyuan Kang, Attacks on one designated verier proxy signature scheme, *Journal of Applied Mathematics*, vol. 2012, pp. 1-6, 2012.

[15] Ying Sun, Chunxiang Xu, Yong Yu, Yi Mu, Strong unforgeable proxy signature scheme secure in the standard model, *The Journal of Systems and Software*, vol. 84, pp. 1471 1479, 2011.