Front. Electr. Electron. Eng. China 2008, 3(2): 167–170
DOI 10.1007/s11460-008-0035-1

RESEARCH ARTICLE

Bo YANG, Zibi XIAO, Yixian YANG, Zhengming HU, Xinxin NIU

A strong multi-designated verifiers signature scheme

Abstract Based on Chameleon Hash and D. Boneh's one round multi-party key agreement protocol, this paper proposes a multi-designated verifiers signature scheme. In this scheme only the verifiers designated by the signer can independently verify the signature. And no one else other than the designated person can be convinced by this signature even if one of the designated verifiers reveals the secret value. The analysis of the proposed scheme shows that it satisfies non-transferability, unforgeability and privacy of the signer's identity and has to low computational cost.

Keywords multi-designated verifiers signature, multilinear map, privacy of signer's identity

1 Introduction

Designated verifier signature (DVS), first proposed by Jakobsson, Sako and Impagliazzo at Eurocrypt'96 [1] is a special type of digital signature that provides authentication of a message without non-repudiation, which is the main property of traditional signatures, thus protecting the anonymity of the signer. Such signature scheme has numerous applications in various situations, such as electronic voting, electronic auction, call for tenders and software licensing. The scheme proposed in Ref. [1] has the property of signer ambiguous in the sense that one cannot verify whether the signer Alice or the designated verifier Bob issues the signature, since Bob can always create a signature that is indistinguishable from the original one. However, the signature remains universally verifiable, that is, anyone can make sure that there are only two potential signers. Hence, if the signature sent by Alice is intercepted and captured by a third party on the line before Bob receives it, the third party can identify that the signer is Alice, since it is now sure with high probability that the designated verifier did not forge the signature. To overcome this problem, Saeednia et al. proposed strong designated verifier signature (SDVS) scheme [2]. In their scheme, the designated verifier is required to use his secret key to verify the validity or invalidity of a signature. Thus, only the designated verifier can verify the signature. Laguillaumie and Vergnaud formalized the property of privacy of signer's identity, which captures the strong designated verifier property [3].

In Ref. [4], Desmedt raised the problem of generalizing the concept of DVS to multi-designated verifiers signatures (MDVS), which allows several designated verifiers to verify the signature. MDVS may be useful in a multi-users setting. For instance, it seems promising for the design of fair distributed contract signing. In Ref. [5], Laguillaumie and Vergnaud provided a formal definition of MDVS and proposed a construction based on ring signatures. In their scheme, an additional encryption layer was needed in order to achieve the property of privacy of signer's identity, which made the protocol loose its spontaneity and become less efficient. In fact, by taking advantage of Joux's tripartite secret exchange protocol, they only proposed an efficient bi-designated verifiers protocol based on bilinear maps. But in their scheme, if one of the designated verifiers reveals the secret value shared by the three parties to an adversary, the adversary can verify the signature, and will believe that the signature is indeed created by the original signer.

In this paper, we proposed a new MDVS scheme based on Chameleon Hash and D. Boneh's one round multi-party key agreement protocol [6], which has the following properties: The signature can be verified independently by each of the n designated verifiers designated by the signer. And no-one else other than the designated person can be convinced by this signature, even if one of the designated verifiers reveals the secret value, because the n designated verifiers can associate with each other to forge the signature of any other message. The session key shared by the signer and the designated verifiers was needed when verifying the signature, so the scheme achieves robustness.

Translated from *Journal of Beijing University of Posts and Telecommunications*, 2007, 30(5): 1–4 [译自: 北京邮电大学学报]

Bo YANG (✉), Yixian YANG, Zhengming HU, Xinxin NIU
Information Security Center, Beijing University of Posts and Telecommunications, Beijing 100876, China
E-mail: cookie_yb73@126.com

Zibi XIAO
School of Science, Wuhan University of Science and Technology, Wuhan 430081, China

2 Preliminaries

2.1 Chameleon Hash function

The idea of Chameleon Hash functions was introduced and formalized by Krawczyk and Rabin in the construction of their chameleon signature schemes, which were based on the hash-and-sign paradigm [7]. To authenticate a message m, a signer computes its digest value h using a Chameleon hash function, and then signs h using an arbitrary signing algorithm out of the signer's choice.

The name 'chameleon' refers to the ability of the owner of the trapdoor information to change the input of the function to any value without changing the output. A Chameleon Hash function is a trapdoor collision-resistant Hash function associated with a user C, who has published a public (hashing) key, denoted as P_C, and holds the corresponding secret key (the trapdoor of finding collisions), denoted as S_C. The public key P_C defines a Chameleon Hash function $\mathrm{CHH}_C(\cdot,\cdot)$. Anyone who knows the public key P_C can efficiently compute the Hash value for each input. However, there exists no efficient algorithm for anyone except the holder of the secret key S_C to find collisions for every given input.

After inputting a message m and a random string r, this function generates a hash value $\mathrm{CHH}_C(m,r)$ which satisfies the following properties:

1) Collision resistance. Without knowledge of trapdoor information S_C, no efficient algorithm is available to find pairs m_1,r_1 and m_2,r_2, where $m_1 \neq m_2$ to satisfy $\mathrm{CHH}_C(m_1, r_1) = \mathrm{CHH}_C(m_2,r_2)$ by inputting the public key P_C, except with negligible probability.

2) Trapdoor collisions. With the knowledge of trapdoor information S_C, collision can be easily found. There is an efficient algorithm, wherein by inputting the secret key S_C, any pair m_1,r_1, and any additional message m_2, a value r_2 can be found that satisfies $\mathrm{CHH}_C(m_1,r_1) = \mathrm{CHH}_C(m_2,r_2)$.

3) Uniformity. All messages m induce the same probability distribution on $\mathrm{CHH}_C(m,r)$ for r is chosen uniformly at random (in particular, from seeing $\mathrm{CHH}_C(m,r)$ for randomly chosen r, nothing is learned about the message m). This condition can be relaxed to require that the above distributions are not necessarily identical for all messages but computationally indistinguishable.

2.2 n-mutilinear map

Definition 1 We say that a map $e : G_1^n \to G_2$ is an n-multilinear map if it satisfies the following properties:

1) G_1 and G_2 are groups of the same prime order.

2) If $a_1,a_2,\ldots,a_n \in Z$ and $x_1,x_2,\ldots,x_n \in G_1$, then $e\left(x_1^{a_1},x_2^{a_2},\ldots,x_n^{a_n}\right) = e(x_1,x_2,\ldots,x_n)^{a_1a_2\ldots a_n}$.

3) The map e is non-degenerate in the following sense: if g is a generator of G_1, then $e(g,\ldots,g)$ is a generator of G_2.

Definition 2 Cryptographic n-multilinear map is an n-multilinear map when it satisfies the requirements as follows:

1) The group operation in G_1 and G_2 is efficiently computable.

2) The map e is efficiently computable.

3) There is no efficient algorithm to compute discrete log in G_1.

Throughout the paper, we represent group elements of G_1 and G_2 as binary strings, namely $G_1,G_2 \subset \{0,1\}^*$.

2.3 Multilinear Diffie-Hellman assumption

The multilinear Diffie-Hellman assumption describes that for given g, g^{a_1}, …, $g^{a_{n+1}}$ in G_1, it is hard to compute $e(g,\ldots,g)^{a_1\cdots a_{n+1}}$ in G_2. That is, there does not exist polynomial time algorithm to compute $e(g,\ldots,g)^{a_1\cdots a_{n+1}}$ from g, g^{a_1}, …, $g^{a_{n+1}}$.

3 Multi-designated verifiers signature

3.1 Definition

MDVS is a group of five algorithms as follow:

1) Common parameter generation (Setup). It is a probabilistic algorithm which inputs a security parameter K and outputs public parameters.

2) Signer key generation (Skeygen). It is a probabilistic algorithm which inputs public parameters and a signer A, and outputs a pair of keys $(\mathrm{PK}_A,\mathrm{SK}_A)$.

3) Verifier key generation (Vkeygen). It is a probabilistic algorithm which inputs public parameters and verifiers B_i, and outputs a pair of keys $(\mathrm{PK}_{B_i},\mathrm{SK}_{B_i})$.

4) n-designated signing (MDVSign). It is an algorithm which inputs a message m, a signing secret key SK_A, the n verifying public keys of the n verifiers B_i, PK_{B_i} ($i = 1,2,\ldots, n$) and public parameters, and outputs a $(B_1,B_2,\ldots,B_n)$-designated verifier signature σ of m. This algorithm can be either probabilistic or deterministic.

5) n-designated verifying (MDVVerify). It is a deterministic algorithm which inputs a bit string σ, a message m, a signing public key PK_A, a verifying secret key SK_{B_i} ($i = 1,2,\ldots,n$), and public parameters, and tests whether σ is a valid $(B_1,B_2,\ldots,B_n)$-designated verifiers signature of m with respect to the keys PK_A, $\mathrm{PK}_{B_1},\ldots,\mathrm{PK}_{B_n}$.

3.2 Security requirements

MDVS scheme must satisfy the following properties:

1) Correctness. A properly formed $(B_1,B_2,\ldots,B_n)$-designated verifiers signature must be accepted by the verifying algorithm.

2) Unforgeability. Given a signer A, it is computationally infeasible, without the knowledge of the secret key of either signer A or those of all verifiers B_i ($i = 1,2,\ldots,n$), to

produce a $(B_1, B_2,\ldots,B_n)$-designated verifiers signature that is accepted by the verifying algorithm.

3) Non-transferability. Even if the designated verifiers reveal their secret values to a third-party, the third-party cannot distinguish whether a MDVS was produced by the designator or forged by the designated verifiers.

4) Privacy of signer's identity (strength). Given a message m and a $(B_1,B_2,\ldots,B_n)$-designated verifiers signature σ of m, it is computationally infeasible without the knowledge of the secret key of one B_i for some $i\in[1,n]$ or that of the signer, to determine which pair of signing keys to be used to generate σ.

4 Description of scheme

In this section, we present a new MDVS scheme based on Chameleon Hash function and multilinear map. The construction is as follows:

Setup: given a security parameter K, the algorithm chooses the system parameters that include two groups G_1,G_2 of prime order P, a cryptographic n-multilinear map e between G_1 and G_2, a generator g of G_1. It also chooses a cryptographic Hash function $h_1 : \{0,1\}^* \to z_p^*$.

Skeygen: signer A picks a random $x_A \in z_p^*$, and computes $y_A = g^{x_A} \in G_1$. A's public key is y_A and secret key is x_A.

Vkeygen: designated verifier B_i picks a random $x_i \in z_p^*$, and computes $y_i = g^{x_i} \in G_1$. y_i is the public key and x_i is the secret key of n designated verifiers B_i $(i = 1,2,\ldots,n)$.

MDVSign: given a message $m \in z_p^*$, signer A picks a random $r \in z_p^*$, computes $Y = \prod_{i=1}^{n} y_i$, $M = H_B(m,r) = Y^m g^r$ (B denotes n designated verifiers $B_1,\ldots,B_n$). Then signer A randomly chooses $k \in z_p^*$ such that $\gcd(k,p) = 1$. Computes $S = e(y_1,\ldots,y_n)^{x_A}$, $t = h_1(S)$, $\gamma = g^{kt}$, $\delta = k^{-1}\{h_1(M) - x_A h_1(\gamma)\}$. The signature on message m is (m,r,γ,δ). Finally, Signer A sends (m,r,γ,δ) and $y_1,y_2,\ldots,y_n$ to every designated verifier B_i.

MDVVerify: every designated verifier B_i computes independently $S = e(y_1,y_2,\ldots,y_{i-1},y_A,y_{i+1},\ldots,y_n)^{x_i}$, $t = h_1(S)$, $Y = \prod_{i=1}^{n} y_i$, and Chameleon Hash value $M = H_B(m,r) = Y^m g^r$. Then test whether $\gamma^\delta y_A^{h_1(\gamma)t} = g^{h_1(M)t}$ holds with equality.

5 Analysis on proposed schemes

5.1 Security

1) Correctness. Firstly, because

$$\begin{aligned} &e(y_1,y_2,\ldots,y_{i-1},y_A,y_{i+1},\ldots,y_n)^{x_i} \\ &= e(g^{x_1},g^{x_2},\ldots,g^{x_{i-1}},g^{x_A},g^{x_{i+1}},\ldots,g^{x_n})^{x_i} \\ &= e(g,\ldots,g)^{x_1\cdots x_n x_A} = e(g^{x_1},\ldots,g^{x_n})^{x_A}, \end{aligned}$$

n designated verifiers and the signer will obtain the same session key S. Secondly, the correctness of the MDVS scheme is justified as follows:

$$\gamma^\delta y_A^{h_1(\gamma)t} = g^{kt\delta} g^{x_A h_1(\gamma)t} = g^{[k\delta + x_A h_1(\gamma)]t} = g^{h_1(M)t}.$$

2) Non-transferability. The non-transferability is achieved because n designated verifiers can collude to generate a signature of other message m', which is indistinguishable from that of the message m by the original signer. Every designated verifiers B_i computes respectively $x_i(m - m')$, then $r' = r + \sum_{i=1}^{n} x_i(m - m')$. Because

$$\begin{aligned} H_B(m',r') &= Y^{m'} g^{r'} = \left(\prod_{i=1}^{n} y_i\right)^{m'} g^r g^{(m-m')\sum_{i=1}^{n} x_i} \\ &= g^{m'\sum_{i=1}^{n} x_i} g^r g^{(m-m')\sum_{i=1}^{n} x_i} = g^{m\sum_{i=1}^{n} x_i} g^r \\ &= \left(\prod_{i=1}^{n} y_i\right)^{m} g^r = Y^m g^r = H_B(m,r), \end{aligned}$$

(m',r',γ,δ) is a valid signature of message m'. Therefore, no-one else other than the designated person can be convinced by the authenticity of the signature even if one of the designated verifiers reveals the secret value to others. However, a user in the designated verifiers group will be convinced because if he has not colluded, he is ensured that the signature is authentic.

3) Unforgeability. Firstly, Chameleon Hash function $H_B(m,r) = Y^m g^r$ is collision resistant. Without the knowledge of trapdoor information, computing trapdoor collision, that is, for any given m, m' and r all in z_p^*, finding a value r', such that $H_B(m,r) = H_B(m',r')$, i.e., $Y^m g^r = Y^{m'} g^{r'}$, $g^{r'} = g^r Y^{(m-m')}$, is identical to the hardness of resolving discrete logarithms problem. Secondly, our signature algorithm is virtually a generalized ElGamal signature [8] on a Chameleon hash value of the message m.

4) Privacy of signer's identity. The verification of validity or invalidity of signatures in our scheme may only be performed by designated verifiers, since the session key S shared by the signer and n designated verifiers is involved in the verification algorithm. Furthermore, the session key S comes from the one-round multi-party key exchange protocol presented by D. Boneh [6], and its security is based on the multilinear Diffie-Hellman assumption. Therefore, even if an eavesdropper can capture the transcripts sent to B_i by signer A, all that he may observe is a set of transcripts that are actually indistinguishable from random strings of the same length and distribution. This means the eavesdropper can neither verify such a signature, nor distinguish the transcripts from a random string of the same length and distribution. Thus, our scheme is inherently strong.

5.2 Efficiency

We compare our scheme (when n equals to 2) with the bi-designated verifiers signature scheme in Ref. [5] in terms of computation overhead. Both signing algorithms require four

exponentiations, one reversion and one pairing operation, so the efficiency is equivalent. However, the verifying algorithm of the scheme in Ref. [5] needs two exponentiations and four pairing operations while our scheme requires six exponentiations and only one pairing operation. The pairing computation is the operation which insofar takes the most running time, in this sense, our scheme is more efficient.

6 Conclusions

MDVS is very important in a multi-user setting. In this paper, we presented a new strong MDVS scheme. Using this scheme, *n*-designated verifiers can verify the signature independently and no-one else other than the designated person can be convinced by this signature. The security analysis of the proposed scheme shows that it satisfies non-transferability, unforgeability, and privacy of signer's identity and has low computational cost.

Acknowledgements This work was supported by the National Basic Research Program of China (No. 2007CB311203), the National Natural Science Foundation of China (Grant No. 90604022).

References

1. Jakobsson M, Sako K, Impagliazzo R. Designated verifier proofs and their applications. In: Proceedings of Advances in Cryptology - Eurocrypt'96. Berlin: Springer-Verlag, 1996, 1070: 143–154
2. Saeednia S, Kremer S, Markowitch O. An efficient strong designated verifier signature scheme. In: Proceedings of the 6th International Conference on Information Security and Cryptology - ICISC 2003. Berlin: Springer-Verlag, 2004, 2971: 40–54
3. Laguillaumie F, Vergnaud D. Designated verifier signature: anonymity and efficient construction from any bilinear map. In: Proceedings of the 4th International Conference on Security in Communication Networks'04 (SCN04). Berlin: Springer-Verlag, 2005, 3352: 105–119
4. Desmedt Y. Verifier-Designated Signatures. Rump Session, Crypto'03, 2003
5. Laguillaumie F, Vergnaud D. Multi-designated verifiers signatures. In: Proceedings of the 6th International Conference on Information and Communications Security - ICICS 2004. Berlin: Springer-Verlag, 2004, 3269: 495–507
6. Boneh D, Silverberg A. Applications of multilinear forms to cryptography. Contemporary Mathematics, 2003, 324: 71–90
7. Krawczyk H, Rabin T. Chameleon signatures. In: Proceeding of Network and Distributed System Security Symposium, California. 2000, 143–154
8. Menezes A J, van Oorschot P C, Vanstone S A. Handbook of Applied Cryptography. 5th ed. Boca Raton: CRC Press, 2001, 451–462