複数指定検証者署名の安全性について

伊豆 哲也 † 武仲 正彦 † 牛田 芽生恵 ‡ 太田 和夫 ‡

† 株式会社富士通研究所
〒 211-8588 川崎市中原区上小田中 4-1-1

‡ 電気通信大学
〒 180-8585 調布市調布ヶ丘 1-5-1

{izu,takenaka}@labs.fujitsu.com

あらまし 指定検証者署名の拡張として, Laguillamie-Vergnaud 複数指定検証者署名を導入し, MDVS1, MDVS2 という 2 つの具体的な方式を提案した. 本稿は MDVS1, MDVS2 に対する署名偽造攻撃を提案する.

Multiple Designated Verifiers Signatures Reconsidered

Tetsuya Izu† Masahiko Takenaka† Mebae Ushida‡ Kazuo Ohta‡

†FUJITSU Lab.,
4-1-1, Kamikodanaka, Nakahara-ku,
Kawasaki, 211-8588, Japan

‡The University of Electro-Communications,
1-5-1, Chofugaoka, Chofu, 182-8585, Japan

{izu,takenaka}@labs.fujitsu.com

Abstract A *multiple designated verifiers signature* (MDVS) was introduced by Laguillaumie-Vergnaud, as a natural extension of the *designated verifiers signature* (DVS) introduced by Jakobsson-Sako-Impagliazzo and Chaum independently, in which specific verifiers chosen by the signer (*designated verifiers*) are the only entities who can verify the signature. Laguillaumie-Vergnaud also constructed two concrete MDVS schemes MDVS1 and MDVS2 from bilinear maps which are proved to be secure in the random oracle model. This paper proposes a new forgery attack against MDVS1 and MDVS2, which allows an adversary, from a valid signature σ on a document, to forge a signature $\overline{\sigma}$ on the same document. Because of the definition of the unforgeability of MDVS schemes, when all designated verifiers are colluded, then can forge a signature σ' on an arbitrary document (and thus the same document). However, the signer cannot distinguish who forged a signature (whether the adversary or the colluded designated verifiers) when the forged signature is given. Thus, the signer cannot convince the designated verifiers and this is critical for MDVS because the scheme is based on the trusty relationship between the signer and the designated verifiers.

1 Introduction

Jakobsson-Sako-Impagliazzo and Chaum independently introduced the notion of the *designated verifier signature* (DVS) in 1996 [JSI96, Cha96] in which a specific verifier chosen by the signer (*designated verifier*) is the only entity who can verify the signature. Though DVS does not have non-repudiation property unlike standard digital signature schemes, DVS has many applications such as e-voting and e-contract. In 2003, Desmedt pointed out some

possible generalizations of DVS [Des03], especially, a question whether DVS can be extended to allow several designated verifiers.

In response to this question, Laguillaumie-Vergnaud introduced the *multiple designated verifiers signature* (MDVS) in which several designated verifiers chosen by the signer the only entities who can verify the signature. Not only a formal definition and required security, Laguillaumie-Vergnaud constructed two concrete schemes MDVS1 and MDVS2 from bilinear maps: MDVS1 is based on Joux's tripartite key exchange [Jou00] (thus it only accepts two designated verifiers) [LV04], while MDVS2 is an extension of MDVS1 to n designated verifiers [LV07] [1]. Compared to Jakobsson-Sako-Impagliazzo's idea for allowing multiple designated verifiers [JSI96], Laguillaumie-Vergnaud's schemes are efficient in the sense of the unnecessity of encryption layer.

Contribution of this Paper

In this paper, we propose a forgery attack against MDVS1 and MDVS2. The proposed attack is very similar to the strong existential forgery attack against standard signature schemes: on input a valid multiple designated verifiers signature σ on a document m, the proposed attack enables an adversary to construct a forged signature $\overline{\sigma}$ on the same document. Since MDVS is called unforgeable when the forgery is infeasible without the secret key of the signer or the secret keys of all designated verifiers [LV04], the signer knows in advance the possibility of the forgery σ' on an arbitrary document (and thus the same document) by colluded all designated verifiers. The most remarkable result of our forgery is that the signer cannot distinguish who forged the signature (whether the adversary or the colluded designated verifiers). Thus, the signer cannot convince designated verifiers when the forged signature is given. This is critical for MDVS because the scheme is based on the trusty relationship between the signer and the designated verifiers.

[1] However, the first scheme is not obtained from the second scheme even if the number of designated verifiers are limited to two.

2 Multiple Designated Verifiers Signature (MDVS)

This section briefly explains the *multiple designated verifiers signature* (MDVS) scheme introduced by Laguillamie-Vergnaud [LV04, LV07].

2.1 Definition

A multiple designated verifiers signature consists of five algorithms Setup, SKeyGen, VKeyGen, Sign and Verify.

Setup is a probabilistic algorithm which takes a security parameter k as input and outputs the public parameters.

SKeyGen is a probabilistic algorithm which takes the public parameter as input and outputs a pair of secret and public key $(\mathsf{sk}_S, \mathsf{pk}_S)$ for a signer.

VKeyGen is a probabilistic algorithm which takes the public parameter as input and outputs a pair of secret and public key $(\mathsf{sk}_V, \mathsf{pk}_V)$ for a designated verifier.

Sign is a probabilistic or deterministic algorithm which takes a message m, the signer's secret key sk_S, the designated verifiers' public keys $\mathsf{pk}_1, \ldots, \mathsf{pk}_n$ (where n denotes the number of the designated verifiers) and the public parameters as input, and output a $(V_1, \ldots, V_n)$-designated verifiers signature σ on m.

Verify is a deterministic algorithm which takes a document m, a $(V_1, \ldots, V_n)$-designated verifiers signature σ on m, the signer's public key pk_S, the public key pk_i for some $i \in \{1, \ldots, n\}$

and the public parameters as input, and outputs the validity of the signature.

A properly generated $(V_1, \ldots, V_n)$-designated verifiers signature must be accepted by the verifying algorithm Verify using one (namely each) designated verifier's secret key.

As a secure MDVS scheme, it should be *unforgeable* in the sense that forging a $(V_1, \ldots, V_n)$-designated verifiers signature which is accepted by the verifying algorithm is infeasible without the signer's secret key or all designated verifiers' secret keys. A MDVS scheme is called (weakly) secure if it satisfies the unforgeability and the *source hiding*, and strongly secure if satisfies the unforgeability, source hiding, and the *privacy of signer's identity*. However, we do not discuss the other security properties but the unforgeability in this paper. We refer to the original paper [LV04] for details.

2.2 MDVS from Bilinear Maps

This section describes two MDVS schemes from bilinear maps proposed by Laguillamie-Vergnaud [LV04, LV07].

In the followings, $\mathbb{G}$ ($\mathbb{G}_T$) is an additive (multiplicative) cyclic groups with order p (prime), respectively, and P is a generator of $\mathbb{G}$ (namely, $\mathbb{G} = \langle P \rangle$). We assume that the Computational Diffie-Hellman (CDH) problem in these groups are infeasible. Let e be a bilinear map from $\mathbb{G} \times \mathbb{G}$ to $\mathbb{G}_T$ such that $e(P, P) \neq 1$ (non-degeneracy) and $e(aP, bP) = e(P, P)^{ab}$ for all $u,\ v \in \mathbb{G},\ a, b \in \mathbb{Z}$ (bilinearity). We also use a cryptographic hash functions $H : \{0,1\}^* \to \mathbb{G}$.

2.2.1 MDVS1

MDVS1 is a multiple designated verifiers signature scheme [LV04] based on Joux's tripartite key exchange [Jou00], and this is why MDVS1 allows only two designated verifiers. Figure 1 describes algorithms of MDVS1, where algorithms SKeyGen and VKeyGen are proceeded by KeyGen1. MDVS1 is proved to be strongly secure in the random oracle model [LV04].

When designated verifiers V_1 and V_2 are colluded in MDVS1, they can generate a forged signature σ' on an arbitrary document as follows:

1. Generate $r,\ r' \leftarrow \mathbb{Z}/p\mathbb{Z}$ randomly.
2. Set $u \leftarrow e(\mathsf{pk}_S, \mathsf{pk}_1)^{\mathsf{sk}_2}$ and $M \leftarrow H(m, u^r)$.
3. Set $Q'_S \leftarrow r'P$ and $Q'_V \leftarrow (\mathsf{sk}_1 + \mathsf{sk}_2)^{-1}(M - r'\mathsf{pk}_S)$.
4. Output a forged signature $\sigma' = (Q'_S, Q'_V, r)$.

Here, the correctness of the above forgery can be verified by the following equation:

$$
\begin{aligned}
& e(Q'_S, \mathsf{pk}_S) \cdot e(Q'_V, \mathsf{pk}_1 + \mathsf{pk}_2) \\
= \ & e(r'P, \mathsf{pk}_S) \\
& \quad \cdot e((\mathsf{sk}_1 + \mathsf{sk}_2)^{-1}(M - r'\mathsf{pk}_S), \mathsf{pk}_1 + \mathsf{pk}_2) \\
= \ & e(P, \mathsf{pk}_S)^{r'} \\
& \quad \cdot e(M - r'\mathsf{pk}_S, (\mathsf{sk}_1 + \mathsf{sk}_2)P)^{(\mathsf{sk}_1 + \mathsf{sk}_2)^{-1}} \\
= \ & e(P, r'\mathsf{pk}_S) \cdot e(M - r'\mathsf{pk}_S, P) \\
= \ & e(P, r'\mathsf{pk}_S) \cdot e(M, P)/e(r'\mathsf{pk}_S, P) \\
= \ & e(M, P).
\end{aligned}
$$

2.2.2 MDVS2

MDVS2 is a multiple designated verifiers signature scheme [LV07] as an extension of MDVS1 to n designated verifiers (however, MDVS2 cannot be obtained from MDVS1 even when the number of designated verifiers are limited to two). Figure 2 describes algorithms of MDVS2, where algorithms SKeyGen and VKeyGen are proceeded by KeyGen2, and Setup2, KeyGen2 are same as Setup1, KeyGen1. MDVS2 is proved to be strongly secure in the random oracle model [LV07].

When all designated verifiers $V_1, \ldots, V_n$ are colluded in MDVS2, they can generate a forged

signature σ' on an arbitrary document as follows:

1. Generate $r,\ r' \leftarrow \mathbb{Z}/p\mathbb{Z}$ randomly.
2. Set $Y \leftarrow rP$, $Y_i \leftarrow r\mathsf{pk}_i$ and $M \leftarrow H(m, \mathsf{pk}_1, \ldots, \mathsf{pk}_n, Y)$.
3. Set $Q'_S \leftarrow r'P$ and $Q'_V \leftarrow (\mathsf{sk}_1 + \cdots + \mathsf{sk}_n)^{-1}(M - r'\mathsf{pk}_S)$.
4. Output a forged signature $\sigma' = (Q'_S, Q'_V, Y_1, \ldots, Y_n)$.

Here, the correctness of the above forgery can be verified by the following equation:

$$\begin{aligned} & e(Q'_S, \mathsf{pk}_S) \cdot e(Q'_V, \mathsf{pk}_1 + \cdots + \mathsf{pk}_n) \\ = \ & e(r'P, \mathsf{pk}_S) \\ & \quad \cdot e((\mathsf{sk}_1 + \cdots + \mathsf{sk}_n)^{-1}(M - r'\mathsf{pk}_S), \\ & \quad \mathsf{pk}_1 + \cdots + \mathsf{pk}_n) \\ = \ & e(P, \mathsf{pk}_S)^{r'} \\ & \cdot e(M - r'\mathsf{pk}_S, (\mathsf{sk}_1 + \cdots + \mathsf{sk}_n)P)^{(\mathsf{sk}_1 + \cdots + \mathsf{sk}_n)^{-1}} \\ = \ & e(P, r'\mathsf{pk}_S) \cdot e(M - r'\mathsf{pk}_S, P) \\ = \ & e(P, r'\mathsf{pk}_S) \cdot e(M, P)/e(r'\mathsf{pk}_S, P) \\ = \ & e(M, P). \end{aligned}$$

3 Proposed Forgery Attacks

This section describes how to forge a signature in MDVS1 and MDVS2. An idea of the forgery is to use the redundancy of the verification equation: an adversary can revise $Q_S,\ Q_V$ in the signature if a product of two bilinear maps are unchanged.

3.1 Forgery Attack against MDVS1

When an adversary obtains a valid (V_1, V_2)-designated verifiers signature $\sigma = (Q_S, Q_V, r)$ on a document m, he can forge a (V_1, V_2)-designated verifiers signature $\overline{\sigma} = (\overline{Q}_S, \overline{Q}_V, r)$ on the same document as follows:

1. Generate $\varepsilon \leftarrow \mathbb{Z}/p\mathbb{Z}$ randomly.
2. Set $\overline{Q}_S \leftarrow Q_S + \varepsilon(\mathsf{pk}_1 + \mathsf{pk}_2)$ and $\overline{Q}_V \leftarrow Q_V - \varepsilon\mathsf{pk}_S$.
3. Output a forged signature $\overline{\sigma} = (\overline{Q}_S, \overline{Q}_V, r)$.

Here, the correctness of the above attack can be verified by the following equation:

$$\begin{aligned} & e(\overline{Q}_S, \mathsf{pk}_S) \cdot e(\overline{Q}_V, \mathsf{pk}_1 + \mathsf{pk}_2) \\ = \ & e(Q_S + \varepsilon(\mathsf{pk}_1 + \mathsf{pk}_2), \mathsf{pk}_S) \\ & \quad \cdot e(Q_V - \varepsilon\mathsf{pk}_S, \mathsf{pk}_1 + \mathsf{pk}_2) \\ = \ & e(Q_S, \mathsf{pk}_S) \cdot e(\varepsilon(\mathsf{pk}_1 + \mathsf{pk}_2), \mathsf{pk}_S) \\ & \quad \cdot e(Q_V, \mathsf{pk}_1 + \mathsf{pk}_2)/e(\varepsilon\mathsf{pk}_S, \mathsf{pk}_1 + \mathsf{pk}_2) \\ = \ & e(M, P). \end{aligned}$$

Let m be a fixed document, $\Sigma(m) = \{\sigma\}$ be a set of valid (V_1, V_2)-designated verifiers signatures on m generated by the signer, $\Sigma'(m) = \{\sigma'\}$ be a set of (V_1, V_2)-designated verifiers signatures on m generated by colluded designated verifiers, and $\overline{\Sigma}(m) = \{\overline{\sigma}\}$ be a set of valid (V_1, V_2)-designated verifiers signatures on m forged by the adversary. Then, all signatures in $\Sigma \cup \Sigma' \cup \overline{\Sigma}$ are judged to be valid by the signer and each designated verifier. When a signature on m is given to the signer on , he can convince whether it is in Σ (namely, it is generated by the signer). However, when it is not in Σ, he cannot distinguish whether it is in Σ' or $\overline{\Sigma}$. This is critical for MDVS because the scheme is based on the trusty relationship between the signer and the designated verifiers.

3.2 Forgery Attack against MDVS2

A forgery attack against MDVS2 is very similar to that against MDVS1. When an adversary obtains a valid $(V_1, \ldots, V_n)$-designated verifiers signature $\sigma = (Q_S, Q_V, Y_1, \ldots, Y_n)$ on a document m, he can forge a $(V_1, \ldots, V_n)$-designated verifiers signature $\overline{\sigma} = (\overline{Q}_S, \overline{Q}_V, Y_1, \ldots, Y_n)$ on the same document as follows:

1. Generate $\varepsilon \leftarrow \mathbb{Z}/p\mathbb{Z}$ randomly.

2. Set $\overline{Q}_S \leftarrow Q_S + \varepsilon(\mathsf{pk}_1 + \cdots + \mathsf{pk}_n)$ and $\overline{Q}_V \leftarrow Q_V - \varepsilon\mathsf{pk}_S$.

3. Output a forged signature $\overline{\sigma} = (\overline{Q}_S, \overline{Q}_V, Y_1, \ldots, Y_n)$.

Here, the correctness of the above attack can be verified by the following equation:

$$\begin{aligned} & e(\overline{Q}_S, \mathsf{pk}_S) \cdot e(\overline{Q}_V, \mathsf{pk}_1 + \cdots + \mathsf{pk}_n) \\ = \; & e(Q_S + \varepsilon(\mathsf{pk}_1 + \cdots + \mathsf{pk}_n), \mathsf{pk}_S) \\ & \cdot e(Q_V - \varepsilon\mathsf{pk}_S, \mathsf{pk}_1 + \cdots + \mathsf{pk}_n) \\ = \; & e(Q_S, \mathsf{pk}_S) \cdot e(\varepsilon(\mathsf{pk}_1 + \cdots + \mathsf{pk}_n), \mathsf{pk}_S) \\ & \cdot e(Q_V, \mathsf{pk}_1 + \cdots + \mathsf{pk}_n) \\ & / e(\varepsilon\mathsf{pk}_S, \mathsf{pk}_1 + \cdots + \mathsf{pk}_n) \\ = \; & e(M, P) \end{aligned}$$

Similarly to the forgery attack against MDVS1, the signer in MDVS2 cannot convince designated verifiers since he cannot distinguish who generated the signature, the adversary or the colluded designated verifiers, when the forged signature is given. Again, this is critical for MDVS.

4 Concluding Remarks

This paper proposes a new forgery attack against the multiple designated verifiers signature schemes MDVS1 and MDVS2 proposed by Laguillaumie-Vergnaud [LV04, LV07]. As a result of the proposed forgery attack, the signer cannot convince designated verifier(s) because he cannot distinguish who generated the signature when the forged signature is given. This is critical for both MDVS schemes.

Based on MDVS2, Ohyama-Tanaka proposed a designated verifier signature scheme [OT07]. An idea of their scheme is to regard colluded all designated verifiers in MDVS2 as the single designated verifier in their DVS. Thus, our proposed forgery attack described in the previous section can be applied to their DVS scheme. Becase of the limitation, we omit detailed discussion on their DVS scheme. However, the signer in Ohyama-Tanaka's DVS scheme cannot convince designated verifiers since he cannot distinguish who generated the signature, the adversary or the designated verifier, when the forged signature is given.

One possible solution for avoiding our forgery attack is to construct the strongly existential unforgeability in MDVS and DVS. However, since (colluded) designated verifier(s) can generate a signature in addition to the signature, the definition of the strongly existential unforgeability in MDVS and DVS should be somewhat weakened from that in the standard digital signature.

参考文献

[Cha96] D. Chaum, "Private Signature and Proof System", US Patent no. 5493614, 1996.

[Des03] Y. Desmedt, "Verifier-designated Signatures", *CRYPTO 2003*, rump session, 2003.

[Jou00] A. Joux, "A One Round Protocol for Tripartite Diffie-Hellman", *ANTS IV*, LNCS 1838, pp. 385-394, Springer-Verlag, 2000.

[JSI96] M. Jakobsson, K. Sako and R. Impagliazzo, "Designated Verifier Proofs and Their Applications", *EUROCRYPT 1996*, LNCS 1070, pp. 142-154, Springer-Verlag, 1996.

[LV04] F. Laguillaumie and D. Vergnaud, "Multi-designated Verifiers Signatures", *ICICS 2004*, LNCS 3269, pp. 495-507, Springer-Verlag, 2004.

[LV07] F. Laguillaumie and D. Vergnaud, "Multi-designated Verifiers Signatures: Anonymity without Encryption", *Information Processing Letters*, vol. 102, issues 2-3, pp. 127-132, 2007.

[OT07] C. Ohyama and K. Tanaka, "Privacy of Verifier's Identity on Designated-verifier Signature", *SCIS 2004*, 3C4-5, p. 285, 2007.

図 1: Description of MDVS1

Setup1
Input: Security parameter k.
Output: Public parameters $(p, P, \mathbb{G}, \mathbb{G}_T, H, e)$, where p is a prime, P is a generator of an additive cyclic group $\mathbb{G}$ with p elements, $\mathbb{G}_T$ is a multiplicative cyclic group with p elements, H is a hash function from $\{0,1\}^*$ to $\mathbb{G}$, and e is a bilinear map from $\mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$.

KeyGen1
1. Generate $\mathsf{sk} \leftarrow \mathbb{Z}/p\mathbb{Z}$ randomly and set $\mathsf{pk} \leftarrow \mathsf{sk}P$.
Output: A pair of secret and public keys $(\mathsf{sk}, \mathsf{pk}) \in \mathbb{Z}/p\mathbb{Z} \times \mathbb{G}$

Sign1
Input: Document $m \in \{0,1\}^*$, signer's secret key $\mathsf{sk}_S \in \mathbb{Z}/p\mathbb{Z}$, designated verifiers' public keys $\mathsf{pk}_1, \mathsf{pk}_2 \in \mathbb{G}$
1. Generate $r,\ r' \leftarrow \mathbb{Z}/p\mathbb{Z}$ randomly. 2. Set $u \leftarrow e(\mathsf{pk}_1, \mathsf{pk}_2)^{\mathsf{sk}_S}$ and $M \leftarrow H(m, u^r)$ 3. Set $Q_S \leftarrow \mathsf{sk}_S^{-1}(M - r'(\mathsf{pk}_1 + \mathsf{pk}_2))$ and $Q_V \leftarrow r'P$
Output: A (V_1, V_2)-designated verifiers signature $\sigma = (Q_S, Q_V, r) \in \mathbb{G} \times \mathbb{G} \times \mathbb{Z}/p\mathbb{Z}$

Verify1 (by the t-th designated verifier)
Input: Document $m \in \{0,1\}^*$, a (V_1, V_2)-designated verifiers signature $\sigma = (Q_S, Q_V, r) \in \mathbb{G} \times \mathbb{G} \times \mathbb{Z}/p\mathbb{Z}$, the t-th designated verifier's secret key $\mathsf{sk}_t \in \mathbb{Z}/p\mathbb{Z}$ $(t \in \{1,2\})$ and public keys $\mathsf{pk}_S,\ \mathsf{pk}_1,\ \mathsf{pk}_2 \in \mathbb{G}$
1. Set $u \leftarrow e(\mathsf{pk}_S, \mathsf{pk}_{3-t})^{\mathsf{sk}_t}$ and $M \leftarrow H(m, u^r)$ 2. Check whether $e(M, P) = e(Q_S, \mathsf{pk}_S) \cdot e(Q_V, \mathsf{pk}_1 + \mathsf{pk}_2)$ holds. If not, output invalid and terminate, otherwise output valid and terminate.

図 2: Description of MDVS2

Setup2 = Setup1, KeyGen2 = KeyGen1

Sign2
Input: Document $m \in \{0,1\}^*$, signer's secret key $\mathsf{sk}_S \in \mathbb{Z}/p\mathbb{Z}$, designated verifiers' public keys $\mathsf{pk}_1, \ldots, \mathsf{pk}_n \in \mathbb{G}$
1. Generate $r,\ r' \leftarrow \mathbb{Z}/p\mathbb{Z}$ randomly. 2. Set $Y \leftarrow rP$, $Y_i \leftarrow r\mathsf{pk}_i$ and $M \leftarrow H(m, \mathsf{pk}_1, \ldots, \mathsf{pk}_n, Y)$ 3. Set $Q_S \leftarrow \mathsf{sk}_S^{-1}(M - r'(\mathsf{pk}_1 + \cdots + \mathsf{pk}_n))$ and $Q_V \leftarrow r'P$
Output: A $(V_1, \ldots, V_n)$-designated verifiers signature $\sigma = (Q_S, Q_V, Y_1, \ldots, Y_n) \in \mathbb{G} \times \mathbb{G} \times \mathbb{G} \times \cdots \times \mathbb{G}$

Verify2 (by the t-th designated verifier)
Input: Document $m \in \{0,1\}^*$, a $(V_1, \ldots, V_n)$-designated verifiers signature $\sigma = (Q_S, Q_V, Y_1, \ldots, Y_n) \in \mathbb{G} \times \mathbb{G} \times \mathbb{G} \times \cdots \times \mathbb{G}$, the t-th designated verifier's secret key $\mathsf{sk}_t \in \mathbb{Z}/p\mathbb{Z}$ $(t \in \{1, \ldots, n\})$ and public keys $\mathsf{pk}_S,\ \mathsf{pk}_1, \ldots, \mathsf{pk}_n \in \mathbb{G}$
1. Set $Y \leftarrow \mathsf{sk}_t^{-1} Y_t$ and $M \leftarrow H(m, \mathsf{pk}_1, \ldots, \mathsf{pk}_n, Y)$ 2. Check whether $e(M, P) = e(Q_S, \mathsf{pk}_S) \cdot e(Q_V, \mathsf{pk}_1 + \cdots + \mathsf{pk}_n)$ holds. If not, output invalid and terminate, otherwise output valid and terminate.