Attacking M&M Collective Signature Scheme

Michal Rjaško and Martin Stanek*

Department of Computer Science
Comenius University, Slovak Republic
{rjasko,stanek}@dcs.fmph.uniba.sk

Abstract. A collective signature scheme aims to solve the problem of signing a message by multiple signers. Recently, Moldovyan and Moldovyan [1] proposed a scheme for collective signatures based on Schnorr signatures. We show some security weaknesses of the scheme.

1 Introduction

Digital signature schemes are important cryptographic constructions with wide and diverse applications. A collective signature scheme aims to solve the problem of signing a message by multiple signers (in a more efficient manner than concatenating individual signatures of the signers). Various constructions of such schemes are known, often satisfying additional requirements, e.g. threshold signatures, blind signatures, etc.

Recently, Moldovyan and Moldovyan [1] proposed a scheme (we denote it as M&M scheme) for collective signatures and its variants – blind collective signature scheme, and multi-signature scheme for simultaneous signing a package of contracts. The scheme is based on well known Schnorr digital signature scheme [2]. The authors of M&M scheme claim the security of their construction, assuming the security of Schnorr's signatures.

Results. We analyze the security and show several security weaknesses of M&M scheme. In particular we demonstrate:

- how two or more participants can add themselves to any collective signature (without a consent or participation of the original signers);
- how malicious participants can (in what we call a "related public key attack") include arbitrary party in a collective signature using just the knowledge of his/her public key.

We discuss how these weaknesses affect variants of M&M scheme (blind signatures and simultaneous contract signing). In addition, we propose possible modifications of the scheme that fix identified vulnerabilities.

* Research supported by VEGA grant No. 1/0266/09.

2 M&M Scheme

Let p be a large prime, and q be a prime such that $q \mid (p-1)$. Let g be an element with order q in the multiplicative group $(\mathbb{Z}_p^*, \cdot)$. We denote the participants (signers) by $P_1, \ldots, P_m$. A private key of the participant P_i is a randomly chosen value x_i from $\mathbb{Z}_q$. The corresponding public key is computed as $Y_i = g^{x_i} \bmod p$. Let H be a hash function.

In order to collectively sign a message M, the participants $P_1, \ldots, P_m$ perform the following computation:

1. Each signer P_i chooses random $t_i \in \mathbb{Z}_q$, and computes $R_i = g^{t_i} \bmod p$.
2. The signers compute the first part of the signature: $E = H(M \,||\, R)$, where $R = R_1 R_2 \cdot \ldots \cdot R_m \bmod p$.
3. Each signer P_i computes $S_i = t_i + x_i E \bmod q$.
4. The signers compute the second part of the signature: $S = S_1 + \ldots + S_m \bmod q$.
5. The collective signature is the pair $\langle E, S \rangle$.

The validity of the collective signature $\langle E, S \rangle$ of M is tested in the following way (knowing/assuming participants $P_1, \ldots, P_m$ as signers):

1. Compute the collective public key $Y = Y_1 Y_2 \cdot \ldots \cdot Y_m \bmod p$.
2. The signature is valid, if and only if $H(M \,||\, Y^{-E} g^S) \stackrel{?}{=} E$.

Remark 1. The scheme can be stated more generally, in any group G of order q.

3 Security Problems of M&M Scheme

Although the authors of M&M scheme perform a security analysis (see [1]), we were able to find some weaknesses of the scheme. We present our findings in this section.

3.1 Joining a Collective Signature

Let us assume that $P_1, \ldots, P_m$ collectively signed a message M, and the signature is $\langle E, S \rangle$. According the construction in M&M scheme we know that $S = \sum_{i=1}^{m} t_i + x_i E \bmod q$. Any pair of participants, we denote them P_{m+1} and P_{m+2} can join the signature without a consent or participation of the original signers:

1. They select arbitrary t_{m+1}, t_{m+2} satisfying $t_{m+2} \equiv -t_{m+1} \pmod{q}$. Then $R_{m+1} R_{m+2} \bmod p = 1$, and the value R is unchanged. Subsequently, E is unchanged as well.
2. P_{m+1} and P_{m+2} construct new signature:

$$\langle E^*, S^* \rangle = \langle E, S + t_{m+1} + t_{m+2} + E(x_{m+1} + x_{m+2}) \bmod q \rangle.$$

The verification of $\langle E^*, S^* \rangle$ for signers $P_1, \ldots, P_{m+2}$ will be successful (we denote $Y = Y_1 \cdot \ldots \cdot Y_m \bmod p$, and $Y^* = Y \cdot Y_{m+1} Y_{m+2} \bmod p$):

$$\begin{aligned} H(M \,||\, Y^{*-E^*} g^{S^*}) &= H\left(M \,||\, Y^{-E} Y_{m+1}^{-E} Y_{m+2}^{-E} \cdot g^{S+t_{m+1}+t_{m+2}+E(x_{m+1}+x_{m+2})}\right) \\ &= H\left(M \,||\, Y^{-E} \cdot g^{S+t_{m+1}+t_{m+2}}\right) \\ &= H\left(M \,||\, Y^{-E} \cdot g^{S}\right) \\ &= H(M \,||\, R) = E = E^* \end{aligned}$$

It is straightforward to extend this attack to more than two participants.

Remark 2. There are applications of collective signatures, where such "free joining" property can be desirable (such as signing a petition). However, in other scenarios a signer can have an objection to sign a document when arbitrary/unknown participants can join the signature.

Remark 3. The problem can be easily fixed by adding the number of signers or their public keys into the computation of E value, i.e. $E = H(M \,||\, R \,||\, m)$ or $E = H(M \,||\, R \,||\, Y_1 \,||\, \ldots \,||\, Y_m)$. Certainly, the signers must check the correctness of E value (or compute it for themselves) when signing, just like they must do such check in the original scheme.

3.2 Related Public Key Attack

Assume a collaborating group of malicious participants $P_1, \ldots, P_{m-1}$. Let P_m be an arbitrary participant (a victim) outside of the group. This group can create a collective signature of any message M for $P_1, \ldots, P_m$ in the following way:

1. P_1 sets/registers his public key to $Y_1 = Y_m^{-1} \bmod p$. Thus, $x_1 \equiv -x_m \pmod{q}$ although P_1 does not known the value of x_1.
2. $P_2, \ldots, P_{m-1}$ sign the message M using M&M scheme. They obtain a signature $\langle E, S \rangle$.
3. The final collective signature of M for $P_1, \ldots, P_m$ is $\langle E, S \rangle$.

The verification of $\langle E, S \rangle$ for signers $P_1, \ldots, P_m$ will be successful (we denote $Y = Y_1 \cdot \ldots \cdot Y_m \bmod p$, and $\tilde{Y} = Y_2 \cdot \ldots \cdot Y_{m-1}$):

$$\begin{aligned} H(M \,||\, Y^{-E} g^S) &= H(M \,||\, \tilde{Y}^{-E} \cdot (Y_1 Y_m)^{-E} \cdot g^S) \\ &= H(M \,||\, \tilde{Y}^{-E} \cdot g^S) \\ &= H(M \,||\, R) = E \end{aligned}$$

Strictly speaking, the attack does not require a collaboration of $P_1, \ldots, P_{m-1}$, and can be carried by P_1 alone as long as someone does not detect a suspicious public key. On the other hand, the fixes proposed in Section 3.1 do not help in preventing this attack, since the collaborating group of attackers can construct E in any way they need. Moreover, the group can collaborate even more and hide the suspicious (related) public key:

- They choose a random nonempty subset $\mathcal{A} \subset \{P_1, \ldots, P_{m-1}\}$.
- The attackers from $\mathcal{A}$ selects their public keys so that $\prod_{i \in \mathcal{A}} Y_i \equiv Y_m^{-1} \pmod{p}$. Notice that this can be done such that exactly one attacker from $\mathcal{A}$ does not know the secret key corresponding to his public key.
- The attack proceeds as before – signature of M created by $\{P_1, \ldots, P_{m-1}\} \setminus \mathcal{A}$ is again a valid signature of M for $P_1, \ldots, P_m$.

In this case the checking of suspicious public keys requires testing of all possible sets $\mathcal{A}$ – which is infeasible (exponential in m).

An easy fix to related public key attack is to sum participants' secret keys into S in a nonuniform way. For example, each participant can compute his/her S_i as $S_i = t_i + Ew_i x_i \bmod q$, where $w_i = H(Y_i \,||\, E)$. Computation of S is unchanged: $S = S_1 + \ldots + S_m \bmod q$. The signature is still the pair $\langle E, S\rangle$. Notice that the values $w_1, \ldots, w_m$ can be computed from E and public keys, therefore they are not part of the signature. However, the verification must be changed accordingly:

$$H(M \,||\, (Y_1^{w_1} \ldots Y_m^{w_m})^{-E} \cdot g^S) \stackrel{?}{=} E.$$

Certainly, the security properties of this modification must be analyzed in detail. Moreover, the modification increases computational complexity of signing and verification.

3.3 Impact on M&M Variants

The authors [1] proposed two variants of the original collective signature scheme: the blind collective signature scheme, and the multi-signature scheme for simultaneous signing a package of contracts.

Since the "blind" variant creates exactly the same signatures as the original scheme, both attacks described in previous sections can be applied to this scheme as well. Moreover, also the fixes proposed there can be included into this variant.

The problem of simultaneous signing a package of contracts is to produce a collective signature of m participants $P_1, \ldots, P_m$ for n documents $M_1, \ldots, M_n$, where each participant P_i signs the document M_{α_i}, $\alpha_i \in \{1, \ldots, n\}$. The variant of M&M scheme dealing with this problem works in the following way:

1. Each signer P_i chooses random $t_i \in \mathbb{Z}_q$, and computes $R_i = g^{t_i} \bmod p$.
2. The signers compute the first part of the signature: $E = f(R)$, where $R = R_1 R_2 \cdot \ldots \cdot R_m \bmod p$ and f is some compression function, e.g. $f(R) = R \bmod q$.
3. Each signer P_i computes $S_i = t_i + x_i h_i E \bmod q$, where $h_i = H(M_{\alpha_i})$.
4. The signers compute the second part of the signature: $S = S_1 + \ldots + S_m \bmod q$.
5. The collective signature is the pair $\langle E, S\rangle$.

The validity of such collective signature $\langle E, S\rangle$ is verified as follows:

1. Compute the collective "data-dependent" public key $Y = Y_1^{h_1} \cdot \ldots \cdot Y_m^{h_m} \bmod p$, where $h_i = H(M_{\alpha_i})$.

2. The signature is valid, if and only if $f(Y^{-E}g^{S}) = E$.

Since the value R is computed in the same way as in the original scheme, the "joining" attack presented in the Section 3.1 works here as well. Moreover, the participants P_{m+1} and P_{m+2} can choose, which document they sign (even outside the set $\{M_1, \ldots, M_n\}$).

The related public key attack described in the Section 3.2 can be applied to this variant with one restriction. The malicious signer P_1 (or all signers from the set $\mathcal{A}$ in a more general scenario) must sign the same document as the victim P_m. In this case $h_1 = h_m$, so if P_1 sets his public key $Y_1 = Y_m^{-1} \bmod p$ then the collective "data-dependent" public key $Y = Y_1^{h_1} Y_2^{h_2} \cdot \ldots \cdot Y_{m-1}^{h_{m-1}} Y_m^{h_m} \bmod p$ for $P_1, \ldots, P_m$ is the same as for $P_2, \ldots, P_{m-1}$. Hence, as described in the Section 3.2 the malicious participants $P_1, \ldots, P_{m-1}$ can force the victim P_m to sign an arbitrary document.

References

1. N.A. Moldovyan, A.A. Moldovyan: Blind Collective Signature Protocol Based on Discrete Logarithm Problem, *International Journal of Network Security*, Vol. 11, No. 2, pp. 106–113, 2010.
2. C.P. Schnorr: Efficient Signature Generation by Smart Cards, *Journal of Cryptology*, Vol. 4, No. 3, pp. 161–174, 1991.