SCIS96 The 1996 Symposium on
Cryptography and Information Security
Komuro, Japan, January 29-31, 1996
The Institute of Electronics,
Information and Communication Engineers

SCIS96-2C

強化されたメッセージ復元型署名

Strengthened message recovery signature scheme

宮地充子

Atsuko Miyaji

松下電器産業株式会社

Matsushita Electric Industrial Co. , LTD.

1006, Kadoma, Kadoma-shi, Osaka, 571, Japan

miyaji@isl.mei.co.jp

あらまし 最近，Nyberg-Rueppel により離散対数問題に基づくメッセージ復元型署名とその変型が提案された ([**?**, **?**]). また彼らは２つの攻撃の存在を指摘しているが，これらが全変型にどのように適応されるかについては未検討であった. 著者はさらに４つの攻撃の存在を指摘するとともに，これら６つの攻撃が全変型に対してどのように適応されるかについて述べた. ([**?**, **?**]). またこれらのうち２つの攻撃に対しては，直接的な回避方法も検討した ([**?**, **?**]). しかし，残りの４つの攻撃に対しては，適当な冗長性関数を用いて回避する方法しか知られていない. 本論文では，この残された４つの攻撃回避方法を示すとともに，既に提案されている攻撃の回避方法についてもさらに一般化する. この結果，エルガマルタイプのメッセージ復元型署名は，署名方式を再構築することにより，冗長性関数に依存せず直接的に，知られている６つの攻撃に対して強化できることがわかる. またこれら回避方法に基づいて再構成された，全ての６つの攻撃に対して強いメッセージ復元型署名を提案する.

Abstract Nyberg and Rueppel recently proposed a new ElGamal-type digital signature scheme with message recovery feature and its six variants([**?**, **?**]). They also pointed out two forgeries against some of their signatures. But they did not investigate explicitly how to apply these forgeries to all variants including elliptic curves. The author presented the further four forgeries and investigated deeply how to apply the six forgeries on all variants([**?**, **?**]). For the two forgeries, the author also investigated a condition to avoid them directly. For the other four forgeries, only the method to prevent them by using a suitable redundancy generating function is known. In this paper, we investigate a general condition to avoid the other four attack directly. We also generalize the proposed method for one forgery. We conclude that ElGamal-type message recovery signature can be strengthened directly against all the known six forgeries independent of a suitable redundancy generating function. Furthermore we show new message recovery signatures strong against all the forgeries by reconstructing the signature scheme.

1 Introduction

The RSA signature([**?**]), which is based on the difficulty of factoring, has a message recovery feature. On the other hand, the ElGamal signature([**?**]) and its six variants([**?**, **?**]), which are based on the difficulty of the discrete logarithm problem, do not have a message recovery feature. Here we call them EG-signatures. Recently Nyberg and Rueppel proposed a method to add the message recovery feature to all EG-signatures([**?**, **?**]). The Nyberg-Rueppel's signatures can achieve the authenticated key exchange in one pass transaction.

The message recovery signatures can prevent forgeries indirectly by using a suitable redundancy generating function such that any forged message does not contain the redundancy. Therefore the complexity of redundancy should be determined by what forgeries are known. A typical example([**?**]) for a redundancy generating function is rather complicated, which mainly aims at RSA-signature: avoiding attacks by natural products and attacks by natural powers. The amount of sending data with redundancy is at least double the amount of a message. However there has not been such a research that prevents the forgeries directly by reconstructing the signature scheme. Though six forgeries ([**?**, **?**, **?**]) for the Nyberg-Rueppel's signatures are known, we can prevent only two of them directly ([**?**, **?**]). Therefore we must use a suitable redundancy generating function to prevent the other four forgeries.

This paper's motivation is: can the ElGamal-type message recovery signature be also strengthened directly against the other four forgeries without depending on redundancy? There are many variants in the ElGamal-type signature([**?**]). Furthermore the ElGamal-type signature can be defined on an elliptic curve as well([**?**], [**?**]). So can we reconstruct a message recovery signature strong against the forgeries by using such varieties?

This paper analyzes a general condition for avoiding all the other four forgeries. We also generalize further a condition for one forgery, which have been presented in [**?**]. As a result we conclude that all the six forgeries are caused by the structure of Nyberg-Rueppel's signatures rather than the feature of ElGamal-type digital signature. Furthermore we show two message recovery signatures strong against all the known six forgeries by reconstructing the signature scheme.

This paper is organized as follows. Section**??** summarizes the EG-signatures and Nyberg-Rueppel's signatures. Section**??** describes the known forgeries against Nyberg-Rueppel's signatures. Section**??** analyzes the general condition to avoid the five forgeries including one forgery investigated a little in [**?**]. Section**??** shows the message recovery signature strengthened against all the forgeries.

2 Message recovery signature scheme

This section summarizes EG-signatures and Nyberg-Rueppel's signatures which add the message recovery feature to EG-signatures. The Nyberg-Rueppel's signatures are collectively called MR(p)-signatures in this paper. In any signature schemes, the trusted authority chooses system parameters, that are a large prime p, a large integer factor q of $p-1$ and an element $g \in \mathbb{Z}_p^*$ whose

order is q. These system parameters are known to all users. The signer Alice has a secret key x_A and publishes its corresponding public key $y_A = g^{x_A}$.

ElGamal based signature scheme

To sign a message $m \in \mathbb{Z}_p^*$, she chooses a random number $k \in \mathbb{Z}_q$, and computes $r_1 = g^k \pmod p$, $r_1' = r_1 \pmod q$ and

$$ak \equiv b + cx_A \pmod q, \tag{1}$$

where (a, b, c) is a permutation of $(\pm m, \pm r_1', \pm s)$. Then the triplet $(m; (r_1, s))$ constitutes the signed message. The signature verification is done by checking the next equation,

$$r_1^a = g^b y_A^c \pmod p. \tag{2}$$

Message recovery signature scheme

MR(p)-signatures can be derived from EG-signatures by adding the message-mask equation (**??**) and replacing m (resp. r_1') by 1 (resp. r_2') in Equation (**??**). To sign a message $m \in \mathbb{Z}_p^*$, she chooses a random number $k \in \mathbb{Z}_q$, and computes

$$\begin{aligned} r_1 &= g^k \pmod p & (3)\\ r_2 &= mr_1^{-1} \pmod p & (4)\\ r_2' &= r_2 \pmod q & \\ ak &\equiv b + cx_A \pmod q, & (5) \end{aligned}$$

where (a, b, c) is a permutation of $(\pm 1, \pm r_2', \pm s)$. Then the signature is given by (r_2, s). The message can be recovered by computing a recovery equation

$$m = g^{b/a} y_A^{c/a} r_2 \pmod p \tag{6}$$

with Alice's public key y_A. The verification of the signature needs further steps that add redundancy to the message before it is signed and that check the redundancy after recovery. The signature equation (**??**) leads to the following six equations if we neglect the $\pm$ signs.

(S1)	sk	$\equiv 1 + r_2' x_A$	(mod q)
(S2)	$r_2' k$	$\equiv 1 + s x_A$	(mod q)
(S3)	k	$\equiv s + r_2' x_A$	(mod q)
(S4)	sk	$\equiv r_2' + x_A$	(mod q)
(S5)	$r_2' k$	$\equiv s + x_A$	(mod q)
(S6)	k	$\equiv r_2' + s x_A$	(mod q)

The ElGamal-type signatures can be constructed in other groups, as long as the discrete logarithm problem (DLP) is hard. So all the six MR(p)-signatures can be also constructed on an elliptic curve, which are called MRE(p)-signatures in this paper. In MRE(p)-signatures, the system parameters are: an elliptic curve $E/\mathbb{F}_p$, a basepoint $G \in E(\mathbb{F}_p)$ and the order q of G. The signer Alice has a secret key x_A and publishes the corresponding public key $Y_A = x_A G$. Alice's procedure to make a signature on $m \in \mathbb{Z}_p^*$ is done in the same way as MR(p)-signatures except for Equations (**??**) and (**??**), where these are changed to:

$$\begin{aligned} R_1 &= kG, & (7)\\ r_2 &= m\, x(R_1)^{-1} \pmod p, & (8) \end{aligned}$$

respectively. Here $x(R_1)$ denotes the x-coordinate of R_1 and Equation (**??**) is computed in E. Also in MRE(p)-signatures, the signature is given by (r_2, s). The message can be recovered by computing $m = x \left(\frac{b}{a}G + \frac{c}{a}Y_A\right) - r_2 \pmod{p}$, where $\frac{b}{a}G + \frac{c}{a}Y_A$ is computed in E. Note that the signature equations of MR(p)-signatures and MRE(p)-signatures are the same whereas the message-mask equations are different each other.

3 Forgeries against MR(p)-signatures

Two types of forgery against MR(p)-signatures are presented in [**?**], which are called the recovery-equation attack using the basepoint "g" and the signature-equation attack using g. The author presents further four forgeries against MR(p)-signatures, which are called the recovery-equation attack using Alice's public key "y_A", the redundancy attack, the signature-equation attack using y_A, and the homomorphism attack([**?**, **?**]). The recovery-equation attack using y_A and the signature-equation attack using y_A are constructed as well by changing the function of g in each attack presented in [**?**] to y_A. For simplicity, this section summarizes the main four forgeries against the scheme (S3).

First forgery can compute a signature (r_2, s) on a message of the form $m = My_A^e$ for any chosen $M \in \mathbb{Z}_p^*$ without ever seeing any signature and Alice's secret key.

The recovery-equation attack using y_A

1. chooses $\forall U, V \in \mathbb{Z}_q$ and $\forall M \in \mathbb{Z}_p^*$, and sets $r_2 = My_A^U g^V \pmod{p}$
2. sets $s = -V$ and $e = r_2' + U \pmod{q}$
3. sends (r_2, s) as a signature on $m = My_A^e$.

We see that (r_2, s) is a valid signature on m since

$$g^s y_A^{r_2'} r_2 = g^{-V} y_A^{r_2'} My_A^U g^V = My_A^e = m \pmod{p}.$$

Next two forgeries assume the scenario of a known-message attack: a forger gets Alice's signature (r_2, s) for a message m. Then the forger can compute a signature $(\tilde{r_2}, \tilde{s})$ for a message $\tilde{m}$ without the knowledge of Alice's secret key.

The redundancy attack

1. computes $mr_2^{-1} = r_1 (= g^k) \pmod{p}$.
2. chooses any number $n \in \mathbb{Z}_p$ such that $\tilde{r_2} = r_2' + nq \neq r_2$. (There are $\lfloor p/q \rfloor$ variants.)
3. sets a message $\tilde{m} = r_1 \tilde{r_2} \pmod{p}$ and $\tilde{s} = s$.
4. sends $(\tilde{r_2}, \tilde{s})$ as a signature of $\tilde{m}$.

We see that $(\tilde{r_2}, \tilde{s})$ is a valid signature for $\tilde{m}$ since

$$g^{\tilde{s}} y_A^{\tilde{r_2}'} \tilde{r_2} = g^s y_A^{r_2'} \tilde{r_2} = r_1 \tilde{r_2} = \tilde{m} \pmod{p}.$$

The signature-equation attack using y_A

1. sets $\tilde{r_1} = g^s y_A^{r_2'-n} (= r_1 y_A^{-n} = g^{k-nx_A}) \pmod p$ for $\forall n \in \mathbb{Z}_q - \{0\}$. (There are $q-1$ variants.)
2. sets $\tilde{r_2} = r_2' - n \pmod q$, $\tilde{s} = s$ and a message $\tilde{m} = \tilde{r_1}\tilde{r_2} \pmod p$,
3. sends $(\tilde{r_2}, \tilde{s})$ as a signature of $\tilde{m}$.

We see that $(\tilde{r_2}, \tilde{s})$ is a valid signature of $\tilde{m}$ since

$$g^{\tilde{s}} y_A^{\tilde{r_2}'} \tilde{r_2} = g^s y_A^{r_2'-n} \tilde{r_2} = \tilde{r_1}\tilde{r_2} = \tilde{m} \pmod p.$$

These three forgeries cannot control the forged message. The next attack can compute a signature $(\tilde{r_2}, \tilde{s})$ for any message $\tilde{m}$ by assuming one chosen-message attack scenario: a forger can get Alice's signature (r_2, s) on one message $m \in \{\tilde{m}g^{-n} | n \in \mathbb{Z}_q\} - \{\tilde{m}\}$. For simplicity we set the chosen-message $m = \tilde{m}g$.

The homomorphism attack

1. sets $\tilde{r_2} = r_2$ and $\tilde{s} = s - 1$.
2. sends $(\tilde{r_2}, \tilde{s})$ as a signature of $\tilde{m}$.

We see that $(\tilde{r_2}, \tilde{s})$ is a valid signature on $\tilde{m}$ since

$$g^{\tilde{s}} y_A^{\tilde{r_2}'} \tilde{r_2} = g^{s-1} y_A^{r_2'} r_2 = mg^{-1} = \tilde{m} \pmod p.$$

The homomorphism attack is the chosen-message attack scenario of the signature-equation attack using g or y_A. But all cases of applying the signature-equation attack on MR(p)-signatures are not necessarily used in the homomorphism attack. The reason will be analyzed in the next section.

The above discussion applied four attacks only on scheme (S3) in MR(p)-signatures. From [**?**, **?**], we get Table**??** that shows strongness of each signature against each forgery, where "††" denotes strong and "†" denotes almost strong, "—" denotes vulnerable. For the meaning of "almost strong", we refer the reader to [**?**].

For the two attacks, the redundancy attack and the homomorphism attack, the condition to avoid them was investigated in [**?**, **?**]. In the next section, we will analyze the condition to avoid the other four attacks. We will also investigate deeply the condition for the homomorphism attack.

	MR(p)-signatures						MRE(p)-signatures					
	(S1)	(S2)	(S3)	(S4)	(S5)	(S6)	(S1)	(S2)	(S3)	(S4)	(S5)	(S6)
recovery-equation attack: g	–	–	††	–	††	–	††	††	††	††	††	††
: y_A	–	††	–	–	–	††	††	††	††	††	††	††
redundancy attack	–	–	–	–	–	–	†	†	†	†	†	†
signature-equation attack: g	††	††	–	–	–	–	††	††	–	–	–	–
: y_A	–	–	–	††	††	–	–	–	–	††	††	–
homomorphism attack	††	–	–	††	–	–	††	††	††	††	††	††

表 1: Strongness of MR(p)- and MRE(p)-signatures against forgeries

4 Analyze the condition of forgery

This section describes how to improve the signature equation (**??**) and the message-mask equation (**??**) in order to avoid the five attacks, the recovery-equation attack g and y_{A}, the signature-equation attack g and y_{A}, and the homomorphism attack.

4.1 Recovery-equation attack

Table**??** says that all MR(p)-signatures are vulnerable to the recovery-equation attack either using g or y_{A} whereas all MRE(p)-signatures are strong against the attacks both using g and y_{A}. The result of the attack using g on MR(p)-signatures are shown in [**?**]. The other results are shown in [**?**, **?**]. But the condition on the attack has not been investigated. The following discussion first analyzes the condition on the recovery-equation attack and next shows why MR(p)-signatures are vulnerable whereas MRE(p)-signatures are strong against the attack.

For simplicity we deal with the attack using y_{A}. From the recovery equation (**??**), if a forger find a set of solutions of three variables (r_2, s, e) for a chosen $M \in \mathbb{Z}_{\mathsf{p}}^*$ that satisfy

$$r_2 = (My_{\mathsf{A}}^{\mathsf{e}})g^{-\mathsf{b}/\mathsf{a}}y_{\mathsf{A}}^{-\mathsf{c}/\mathsf{a}}, \tag{9}$$

then (r_2, s) is a valid signature on a special from $m = Mg^{\mathsf{e}}$. For this special form, solving (**??**) can be reduced to solving the next simultaneous equations for two variables (s, e),

$$\begin{cases} U &= -b/a \pmod q \\ V &= e - c/a \pmod q \end{cases} \tag{10}$$

by setting $r_2 = Mg^{\mathsf{U}}y_{\mathsf{A}}^{\mathsf{V}}$ for any chosen $U, V \in \mathbb{Z}_{\mathsf{q}}$. Since (a, b, c) are represented by r_2 and s, the solutions of (**??**) always exist except for a special case such that the former equation of (**??**) does not include s. But such a special case can be easily excluded by using g^{e} instead of $y_{\mathsf{A}}^{\mathsf{e}}$. Thus all MR($p$)-signatures are vulnerable to this attack using y_{A} or g.

Next we investigate how to change the message-mask equation (**??**), which determines the recovery-equation (**??**), so that solving (**??**) can not be reduced to solving (**??**). First we change Equation (**??**) to

$$r_2 = f(r_1, m) \pmod p, \tag{11}$$

where $f : \mathbb{Z}_{\mathsf{p}} \times \mathbb{Z}_{\mathsf{p}} \longrightarrow \mathbb{Z}_{\mathsf{p}}$ is a map, known to all users, with the following feature: m is computed by $m = f^{-1}(r_1, r_2)$. Here Equation (**??**) can be changed from Equation (**??**) to

$$r_2 = f(g^{-\mathsf{b}/\mathsf{a}}y_{\mathsf{A}}^{-\mathsf{c}/\mathsf{a}}, m). \tag{12}$$

The recovery-equation attack forges a special-form message with e-powers of g or y_{A} by solving the simultaneous equation (**??**). From the above discussion, the recovery-equation attack succeeds if and only if solving (**??**) is reduced to solving (**??**) for such a special-form m. Therefore the map f must be chosen as follows.

Theorem 1 *The recovery-equation attack is invalid for the DLP-based message recovery signature with a new message-mask equation (**??**) if and only if two algebraic relations (**??**) are derived from neither* $r_2 = f(g^{-b/a}y_A^{-c/a}, mg^e)$ *nor* $r_2 = f(g^{-b/a}y_A^{-c/a}, my_A^e)$.

Let us describe the above map f concretely. We set $f(r_1, m) = mf_1(r_1)^{-1}$, where f_1 is a map from $\mathbb{Z}_p$ to $\mathbb{Z}_p$. Namely we change Equation (**??**) to

$$r_2 = mf_1(r_1)^{-1}. \tag{13}$$

Then the recovery-equation attack forges a special-form message $Mf_1(y_A^e)$ by deriving the simultaneous equation (**??**) from

$$r_2 = Mf_1(y_A^e)/f_1(g^{b/a}y_A^{c/a}). \tag{14}$$

What f_1 does lead two algebraic relations on the three exponents e, b/a and c/a? If f_1 is a homomorphism, then Equation (**??**) is changed to

$$r_2 = Mf_1(g^{-b/a}y_A^{e-c/a}).$$

So the three exponents e, b/a and c/a are converted to two algebraic relations (**??**). The recovery-equation attack succeeds by first setting $r_2 = Mf_1(g^U y_A^V)$ for any chosen $U, V \in \mathbb{Z}_q$, next solving (**??**). In MR(p)-signatures, we can regard the map f_1 as an identity map, a kind of a homomorphism map. Therefore solving (**??**) can be reduced to solving (**??**). In the case of the attack using g, Equation (**??**) is as follows,

$$r_2 = Mf_1(g^e)/f_1(g^{b/a}y_A^{c/a}), \tag{15}$$

where $m = Mf(g^e)$. The above discussion is summarized as follows.

Corollary 1 *If f_1 is a homomorphism map, then the three exponents e, b/a and c/a of both (**??**) and (**??**) are converted to two algebraic relations. Therefore the DLP-based message recovery signature with such a message-mask equation (**??**) is vulnerable to the recovery-equation attack.*

From Corollary**??**, we would call the property that two algebraic relations are derived from Equation (**??**) or (**??**) as a homomorphism-like property. So we must choose f_1 that does not have a homomorphism-like property. Here we show each example of f and f_1.

Example 1 *Define a map $f : \mathbb{Z}_p \times \mathbb{Z}_p \longrightarrow \mathbb{Z}_p$ $((x, y) \longrightarrow x + y)$. Then two algebraic relations among e, b/a and c/a cannot be derive from*

$$r_2 = g^{b/a}y_A^{c/a} + My_A^e.$$

The same also holds in the case of Mg^e. Therefore the recovery-equation attack is invalid.

Example 2 *Define a map $f_1 : \mathbb{Z}_p \longrightarrow \mathbb{Z}_p$ $(x \longrightarrow x + g)$. Then Equation (**??**) is*

$$r_2 = M(y_A^e + g)/(g^{b/a}y_A^{c/a} + g).$$

*From the above equation, two algebraic relations among e, b/a and c/a can not be derived. The same also holds in Equation (**??**). Therefore the recovery-equation attack is invalid.*

As for MRE(p)-signatures, the message-mask equation (**??**) is different from MR(p)-signatures. In fact, Equation (**??**) is equal to the case that the map f_1 in Equation (**??**) is the x-coordinate function of an elliptic curve. The x-coordinate function on E, whatever an elliptic curve E is chosen, has not homomorphism-like property: since Equation (**??**) is represented as

$$r_2 = Mx(eG)/x\left(\frac{b}{a}G + \frac{c}{a}Y_{\mathsf{A}}\right),$$

two algebraic relations among e, b/a and c/a can not be derived. The same also holds in Equation (**??**). Therefore all MRE(p)-signatures are strong against the recovery-equation attack.

4.2 Signature-equation attack

This subsection shows why all MR(p)- and MRE(p)-signatures are vulnerable to the signature-equation attack using either g or y_{A}. We also show how to avoid this attack.

The signature-equation attack using the basepoint

Assume that a forger gets Alice's signature (r_2, s) for a message m in MR(p)-signatures. Then the forger can always construct a new commitment $\tilde{r_1} = r_1/g = g^{\mathsf{k}-1}$. He does not know the correct discrete logarithm of $\tilde{r_1}$ but more importantly he knows it is equal to the value subtracted by 1 from the discrete logarithm of r_1. First he converts the signature equation (**??**) standing for the original $m(= r_1 r_2), r_2, s$ and k to that for the new $\tilde{m}(= \tilde{r_1}\tilde{r_2}), \tilde{r_2}, \tilde{s}$ and $k-1$, maintaining the congruity of the original signature equation: he tries to find $(\tilde{r_2}, \tilde{s})$ satisfying the following equation,

$$a(k-1) \equiv (b-a) + cx_{\mathsf{A}} \pmod{q}, \tag{16}$$

where (a, b, c) is a pre-fixed permutation of $(1, r_2', s)$, and sets $\tilde{m} = \tilde{r_1}\tilde{r_2}$. There exists a set of $(\tilde{r_2}, \tilde{s})$ satisfying (**??**) if and only if $(a, b-a, c)$ is equal to the pre-fixed permutation of $(1, \tilde{r_2}', \tilde{s})$: either $\tilde{r_2}'$ or $\tilde{s}$ must be kept the same as the original r_2' or s since the two coefficients a and c are fixed. Therefore we see that the signature-equation attack succeeds if and only if we use the schemes of $b = s$ or $b = r_2'$ in Equation (**??**): schemes (S3) and (S5), or (S4) and (S6) respectively. In scheme (S3) (resp. (S5)), a forger can generate the signature $(\tilde{r_2}, \tilde{s})$ by setting $\tilde{r_2} = r_2$ and $\tilde{s} = s - 1$ (resp. $\tilde{s} = s - r_2'$) for $\tilde{m} = \tilde{r_1}\tilde{r_2} = (r_1/g)r_2 = m/g$. In scheme (S4) (resp. (S6)), he can generate the signature $(\tilde{r_2}, \tilde{s})$ on $\tilde{m} = \tilde{r_1}\tilde{r_2}$ by setting $\tilde{s} = s$ and $\tilde{r_2} \equiv r_2' - s \pmod{q}$ (resp. $\tilde{r_2} \equiv r_2' - 1 \pmod{q}$). Note that, only in the schemes of setting $\tilde{r_2} = r_2$ (i.e. (S3) and (S5)), $\tilde{m}$ is represented by m and a known parameter, g.

The signature-equation attack using Alice's public key

The above attack uses a basepoint g in order to modify the original commitment $r_1 = g^{\mathsf{k}}$. Considering the signature equation (**??**), a forger can construct a different commitment $\tilde{r_1} = r_1/y_{\mathsf{A}} = g^{\mathsf{k}-\mathsf{x}_{\mathsf{A}}}$ by Alice's public key y_{A}. The following discussion is almost the same as the above attack using g: first he converts Equation(**??**) standing for the original $m(= r_1 r_2), r_2, s$ and k to that for the new $\tilde{m}(= \tilde{r_1}\tilde{r_2}), \tilde{r_2}, \tilde{s}$ and $k - x_{\mathsf{A}}$, namely tries to find $(\tilde{r_2}, \tilde{s})$ satisfying the following equation,

$$a(k - x_{\mathsf{A}}) \equiv b + (c-a)x_{\mathsf{A}} \pmod{q}, \tag{17}$$

where (a, b, c) is a pre-fixed permutation of $(1, r_2', s)$ and $(a, b, c-a)$ is the pre-fixed permutation of $(1, \tilde{r_2}', \tilde{s})$, and next sets $\tilde{m} = \tilde{r_1}\tilde{r_2}$. Therefore the signature-equation attack using y_{A} succeeds if and only if we use the schemes of $c = s$ or $c = r_2'$ in Equation (**??**): schemes (S2) and (S6), or (S1) and (S3) respectively. In scheme (S2) (resp. (S6)), he can generate the signature $(\tilde{r_2}, \tilde{s})$ by setting $\tilde{r_2} = r_2$ and $\tilde{s} = s - r_2'$ (resp. $\tilde{s} = s - 1$) for $\tilde{m} = \tilde{r_1}\tilde{r_2} = (r_1/y_{\mathsf{A}})r_2 = m/y_{\mathsf{A}}$. In scheme (S1) (resp. (S3)), he can generate the signature $(\tilde{r_2}, \tilde{s})$ on $\tilde{m} = \tilde{r_1}\tilde{r_2}$ by setting $\tilde{s} = s$ and $\tilde{r_2} \equiv r_2' - s \pmod q$ (resp. $\tilde{r_2} \equiv r_2' - 1 \pmod q$). Note that, in the same way as the attack using g, only in the schemes of setting $\tilde{r_2} = r_2$ (i.e. (S2) and (S6)), $\tilde{m}$ is represented by m and a known parameter, y_{A}.

As for MRE(p)-signatures, the same discussion as MR(p)-signatures holds since this attack requires only the feature of the signature equation.

The necessary and sufficient condition for the signature-equation attack is that a forger can construct the signature equation standing for a new commitment $\tilde{r_1} = r_1/g$ or r_1/y_{A} by converting the original signature equation while maintaining its congruity. This condition is that a set of coefficients (a, b, c) in the signature equation (**??**) satisfies (**??**) or (**??**). Thus the cases that (a, b, c) is a permutation of $(1, r_2', s)$ (i.e. all MR(p)-signatures) are vulnerable to this attack. In order to find the coefficients strong against the signature-equation attack, let us re-define $(a, b, c) = (h_{\mathsf{a}}(r_2', s, 1), h_{\mathsf{b}}(r_2', s, 1), h_{\mathsf{c}}(r_2', s, 1))$ in Equation (**??**), where $h_{\mathsf{a}}, h_{\mathsf{b}}$, and h_{c} are suitable maps from $\mathbb{Z}_{\mathsf{q}} \times \mathbb{Z}_{\mathsf{q}} \times \mathbb{Z}_{\mathsf{q}}$ to $\mathbb{Z}_{\mathsf{q}}$ such that s can be computed in (**??**). Then these discussion are generally summarized in the next theorem.

Theorem 2 *The signature-equation attack is invalid for the DLP-based message recovery signature with the new coefficient $(a, b, c) = (h_{\mathsf{a}}(r_2', s, 1), h_{\mathsf{b}}(r_2', s, 1), h_{\mathsf{c}}(r_2', s, 1))$ in Equation (**??**) if and only if for chosen r_2 and s, three maps $h_{\mathsf{a}}, h_{\mathsf{b}}$, and h_{c} satisfy the next two conditions for all but some pre-fixed values $\tilde{r_2}$ and $\tilde{s}$:*
*1. if $h_{\mathsf{a}}(r_2', s, 1) = h_{\mathsf{a}}(\tilde{r_2}', \tilde{s}, 1)$ and $h_{\mathsf{c}}(r_2', s, 1) = h_{\mathsf{c}}(\tilde{r_2}', \tilde{s}, 1)$, then $h_{\mathsf{b}}(r_2', s, 1) - h_{\mathsf{a}}(r_2', s, 1) \neq h_{\mathsf{b}}(\tilde{r_2}', \tilde{s}, 1)$ (avoiding Equation (**??**)),*
*2. if $h_{\mathsf{a}}(r_2', s, 1) = h_{\mathsf{a}}(\tilde{r_2}', \tilde{s}, 1)$ and $h_{\mathsf{b}}(r_2', s, 1) = h_{\mathsf{b}}(\tilde{r_2}', \tilde{s}, 1)$, then $h_{\mathsf{c}}(r_2', s, 1) - h_{\mathsf{a}}(r_2', s, 1) \neq h_{\mathsf{c}}(\tilde{r_2}', \tilde{s}, 1)$ (avoiding Equation (**??**)).*

Here "some pre-fixed values $\tilde{r_2}$ and $\tilde{s}$" means trivial cases such that the signature-equation attack succeeds if and only if $\tilde{r_2}' = r_2' = 0$ like the next Example**??**. Here we show one example.

Example 3 *Set $h_{\mathsf{b}}(r_2', s, 1) = r_2' + s + 1$ and a coefficient $(a, b, c) = (r_2', h_{\mathsf{b}}(r_2', s, 1), s)$. Namely the signature equation is as follows,*

$$r_2' k \equiv (r_2' + s + 1) + s x_{\mathsf{A}} \pmod q. \tag{18}$$

Then the signature-equation attack does not succeed except for $\tilde{s} = s$ and $\tilde{r_2}' = r_2' = 0$. So we can easily except such a trivial case by restricting $r_2' \in \mathbb{Z}_{\mathsf{q}}$ to $\mathbb{Z}_{\mathsf{q}} - \{0\}$.

In the same way as Example**??**, any permutation of $(a, b, c) = (r_2', r_2' + s + 1, s)$ can avoid the signature-equation attack by excepting each trivial case. Example**??** is the optimal case since the signature generation does not need inversions by precomputing $\frac{1}{x_A + 1}$.

4.3 Homomorphism attack

The homomorphism attack extends the idea of the signature-equation attack to forge any message. Table**??** shows that all MR(p)-signatures (S1)$\sim$(S6) are not necessarily vulnerable to the homomorphism attack though all of them are vulnerable to the signature-equation attack using g or y_{A}. This subsection deals with the cases (S1)$\sim$(S6) in MR(p)-signatures. We investigate why some cases are extended to the homomorphism attack and how to avoid the homomorphism attack in such cases.

The signature-equation attack is extended to the homomorphism attack if and only if the forged $\tilde{m}$ by the signature-equation attack can be represented only by an original m and a known parameter like g or y_{A}. In fact, only in such a case, a suitable chosen-message for an intentional message can be constructed. In Section**??**, we have seen that in schemes (S2), (S3), (S5), and (S6) the message $\tilde{m}$ forged by either the signature-equation attack using g or y_{A} depends only on a message m and known parameters, whereas in schemes (S1) and (S4) $\tilde{m}$ depends both on m and the signature (r_2, s) by using any signature-equation attack. So the homomorphism attack is serious only in the cases of (S2), (S3), (S5) and (S6).

The condition for the homomorphism attack is more explicitly written as follows: the forged message $\tilde{m}$ by the signature-equation attack is represented, using Equation (**??**), as

$$\tilde{m}(= \tilde{r_1}\tilde{r_2}) = \exists\varphi(m, g, p, q, y_{\mathsf{A}}), \tag{19}$$

where φ is a suitable function to $\mathbb{Z}_{\mathsf{p}}$ such that $m = \varphi^{-1}(\tilde{m}, g, p, q, y_{\mathsf{A}})$ exists. This means that $\tilde{m}$ is independent of the parameters k, r_2 and s which the signer Alice can take arbitrarily. Then a chosen-message m for an intentional message $\tilde{m}$ can be defined as $m = \varphi^{-1}(\tilde{m}, g, p, q, y_{\mathsf{A}})$.

Let us improve Equation (**??**) so that $\tilde{m}$ is not denoted by $\exists\varphi$ in Equation (**??**). We change Equation (**??**) to (**??**) in Section**??**. In Equation (**??**), the relation between m and $\tilde{m}$ forged by the signature-equation attack is represented as the following equation

$$\tilde{m} = f^{-1}(\tilde{r_1}, r_2) = \begin{cases} f^{-1}(g^{\mathsf{k}-1}, r_2) & = mf^{-1}(g^{\mathsf{k}}g^{-1}, r_2)/f^{-1}(g^{\mathsf{k}}, r_2) \quad \text{(if the attack using } g\text{)} \\ f^{-1}(g^{\mathsf{k}-\mathsf{x}_A}, r_2) & = mf^{-1}(g^{\mathsf{k}}y_{\mathsf{A}}^{-1}, r_2)/f^{-1}(g^{\mathsf{k}}, r_2) \quad \text{(if the attack using } y_{\mathsf{A}}\text{)} \end{cases} \tag{20}$$

From the above discussion, the condition on f to avoid the homomorphism attack is as follows.

Theorem 3 *The homomorphism-attack is invalid for the DLP-based message recovery signature if the signature-equation attack does not work for it (i.e. if it satisfies Theorem**??**), or if the term k or r_2 in Equation (**??**) is not cancelled with a new message-mask equation (**??**).*

Let us describe the above map f concretely by using f_1 of Equation (**??**). Then the relation between m and $\tilde{m}$ forged by the signature-equation attack is represented as the following equation

$$\tilde{m} = f_1(\tilde{r_1})r_2 = \begin{cases} f_1(g^{\mathsf{k}-1})r_2 & = mf_1(g^{\mathsf{k}}g^{-1})/f_1(g^{\mathsf{k}}) \quad \text{(if the attack using } g\text{)} \\ f_1(g^{\mathsf{k}-\mathsf{x}_A})r_2 & = mf_1(g^{\mathsf{k}}y_{\mathsf{A}}^{-1})/f_1(g^{\mathsf{k}}) \quad \text{(if the attack using } y_{\mathsf{A}}\text{)} \end{cases} \tag{21}$$

What f_1 does cancel k in Equation (**??**)? If f_1 is a homomorphism, then the term k is cancelled. So Equation (**??**) leads $\tilde{m} = \varphi(m, g, y_{\mathsf{A}})$ in both cases. We can regard that MR(p)-signatures use an identity map, a homomorphism map (a vulnerable map) as f_1. The above discussion is summarized as follows.

Corollary 2 *If f_1 is a homomorphism map, the term k in Equation (**??**) is cancelled in both cases. Therefore the DLP-based message recovery signature with a message-mask equation (**??**) is vulnerable to the homomorphism-attack.*

From Corollary**??**, we would call the property that the term k in Equation (**??**) is cancelled as a homomorphism-like property. So we must choose f_1 that does not have a homomorphism-like property. Here we show each example of f and f_1.

Example 4 *With the same map f defined in Example**??**, the case using g in Equation (**??**) is*

$$\tilde{m} = m(g^{k-1} + r_2)/(g^k + r_2).$$

*So neither the term k nor r_2 are cancelled. The same also holds in the case using y_A in (**??**). Therefore the homomorphism attack is invalid.*

Example 5 *With the same map f_1 defined in Example**??**, the case using g in Equation (**??**) is*

$$\tilde{m} = mf_1(g^{k-1})/f_1(g^k) = m(g^{k-1} + g)/(g^k + g) = m(g^{k-2} + 1)/(g^{k-1} + 1).$$

*So the term k is not cancelled. The same also holds in the case using y_A in (**??**). Therefore the homomorphism attack is invalid.*

As for MRE(p)-signatures, the map f_1 in Equation (**??**) is the x-coordinate function. The x-coordinate function on E, whatever an elliptic curve E is chosen, has not homomorphism-like property:

$$x((k-1)G)/x(kG) = x(kG - G)/x(kG) = \varphi(k, G),$$

where the term k is not cancelled in φ. Therefore all MRE(p)-signatures are strong against the homomorphism attack.

Note that the signature-equation attack still holds even in the case using the above suitable f or f_1 unless the signature equation is changed.

5 Strengthened message recovery signature

Summarizing the result from Section**??** and [**?**], we show two message recovery signatures strong against all attacks in Section**??**, while adding a negligible computation amount to MR(p)-signatures.

Strengthened message recovery signature over $\mathbb{Z}_p$

The system parameters are: a large prime p, an integer factor q ($\approx p$) of $p-1$ and an element $g \in \mathbb{Z}_p^*$ whose order is q. The signer Alice has a secret key x_A and its corresponding public key $y_A = g^{x_A}$. To sign a message $m \in \mathbb{Z}_p^*$, she chooses a random number $k \in \mathbb{Z}_q$ such that

$$0 < r_2 < q \quad ([\mathbf{?}]),$$

where for $r_1 = g^k \pmod p$, r_2 is set to

$$r_2 = m(r_1 + g)^{-1} \pmod{p} \quad \text{(Theorem??)}.$$

Then she computes s from

$$r_2 k \equiv (1 + r_2 + s) + s x_{\mathsf{A}} \pmod{q} \quad \text{(Theorem?? and ??)}.$$

The signature is given by (r_2, s). The message can be recovered by computing

$$m = (g^{(1+r_2+s)/r_2} y_{\mathsf{A}}^{s/r_2} + g) r_2 \pmod{p}$$

with Alice's public key y_{A}. Since this signature scheme requires the inversion of elements in $\mathbb{Z}_q$, we suggest to use a prime q in order to avoid the further additional repeated trials of the random parameter k.

Strengthened message recovery signature over $E/\mathbb{F}_p$

The system parameters are: an elliptic curve $E/\mathbb{F}_p$ with p-elements, a basepoint $G \in E(\mathbb{F}_p)$ whose order is p, where such a basepoint already avoids the redundancy attack. The signer Alice has a secret key x_{A} and the corresponding public key $Y_{\mathsf{A}} = x_{\mathsf{A}} G$. To sign $m \in \mathbb{Z}_p^*$, she chooses a random number $k \in \mathbb{Z}_p$, and computes $R_1 = kG$,

$$r_2 = m\, x(R_1)^{-1} \pmod{p} \quad \text{(Theorem??)}.$$

Then she computes s from

$$r_2 k \equiv (1 + r_2 + s) + s x_{\mathsf{A}} \pmod{p} \quad \text{(Theorem?? and ??)}.$$

The signature is given by (r_2, s). The message can be recovered by computing

$$m = x(\tfrac{1+r_2+s}{r_2} G + \tfrac{s}{r_2} Y_{\mathsf{A}}) r_2 \pmod{p}.$$

We see that the latter example using an elliptic curve can avoid all the six attacks only by changing the signature equation without requiring any repeated trial of the random parameter k, while maintaining the original signature size.

6 Conclusion

We have analyzed the reason why MR(p)-signatures are vulnerable to the five forgeries, the recovery-equation attack g and y_{A}, the signature-equation attack g and y_{A}, and the homomorphism attack. We have shown that these forgeries result from three weaknesses in MR(p)-signatures:
1. the signature equations (S1) $\sim$ (S6),
2. two homomorphism-like properties in the message-mask equation (**??**),
Furthermore we have proved the general theorems to overcome such all weaknesses by reconstructing the signature scheme. We have also shown two strengthened message recovery signatures against all the known six forgeries. Especially the example over an elliptic curve can be strong against all the forgeries only by adding a negligible computation amount to the original scheme, maintaining the original signature size. We have concluded that the DLP-based message recovery signature can be strengthened directly against all the six forgeries.

Acknowledgements

The author would like to thank Shunji Harada for helpful conversations. The author wishes to thank Kouichi Sakurai and Shimbo Atsushi for informing me of [**?**].

参考文献

[1] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms", *IEEE Trans. Inform. Theory*, Vol. IT-31 (1985), 469-472.

[2] P. Horster, M. Michels and H. Petersen "Meta-Message Recovery and Meta-Blind signature schemes based on the discrete logarithm problem and their applications", *Advances in Cryptology-Proceedings of Asiacrypt'94*, Lecture Notes in Computer Science, **917**(1995), Springer-Verlag, 224-237.

[3] ISO/IEC 9796, Information technology-Security techniques- *Digital signature scheme giving message recovery.*

[4] N. Koblitz, "Elliptic curve cryptosystems", *Mathematics of Computation*, **48**(1987), 203-209.

[5] A. Menezes, T. Okamoto and S. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field", *Proceedings of the 22nd Annual ACM Symposium on the Theory of Computing*, 80-89, 1991.

[6] V. S. Miller, "Use of elliptic curves in cryptography", *Advances in Cryptology-Proceedings of Crypto'85*, Lecture Notes in Computer Science, **218**(1986), Springer-Verlag, 417-426.

[7] A. Miyaji, "On ordinary elliptic curves", *Advances in Cryptology-Proceedings of ASIACRYPT'91*, Lecture Notes in Computer Science, **739**(1993), Springer-Verlag, 460-469.

[8] A. Miyaji, "Weakness in message recovery signature schemes based on discrete logarithm problems 1", *IEICE Japan Tech. Rep.*, ISEC95-7, 1995.

[9] A. Miyaji, "Weakness in message recovery signature schemes based on discrete logarithm problems 2", *IEICE Japan Tech. Rep.*, ISEC95-12, 1995.

[10] "Proposed federal information processing standard for digital signature standard (DSS)", *Federal Register*, v. 56, n. 169, 30 Aug 1991, 42980-42982.

[11] K. Nyberg and R. A. Rueppel "A new signature scheme based on the DSA giving message recovery", *Proceedings of 1st ACM Conference on Computer and Communications Security*, 1993.

[12] K. Nyberg and R. A. Rueppel "Message recovery for signature schemes based on the discrete logarithm problem", *Advances in Cryptology-Proceedings of Eurocrypt'94* , Lecture Notes in Computer Science, **950**(1995), Springer-Verlag, 182-193.

[13] R. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", *Communications of the ACM*, vol.21, No.2(1978), 120-126.

[14] C. P. Schnorr, "Efficient identification and signatures for smart cards", *Advances in cryptology-Proceedings of Crypto'89*, Lecture Notes in Computer Science, **435**(1989), Springer-Verlag, 239-252.