University of Technology Chemnitz-Zwickau

Department of Computer Science

Theoretical Computer Science and Information Security

Technical Report

TR-94-9

Meta Message Recovery and Meta Blind signature schemes based on the discrete logarithm problem and their applications

Patrick Horster · Markus Michels · Holger Petersen

July 1994

Straße der Nationen 62
D-09111 Chemnitz, Germany

Phone: (+49) 371-531-1479
Fax: (+49) 371-531-1810

Meta Message Recovery and Meta Blind signature schemes based on the discrete logarithm problem and their applications

Patrick Horster · Markus Michels · Holger Petersen

Theoretical Computer Science and Information Security,
University of Technology Chemnitz-Zwickau,
Straße der Nationen 62, D-09111 Chemnitz, Germany
E-mail: {pho,hpe,mmi}@informatik.tu-chemnitz.de

Revised December 9, 1994

Abstract

There have been several approaches in the past to obtain signature schemes with message recovery based on the discrete logarithm problem. In this paper we generalize these approaches into a Meta-Message recovery scheme by applying the ideas of the Meta-ElGamal signature scheme. Furthermore we present Meta-blind signature schemes which have been developed from the ElGamal based blind signature scheme and the message recovery blind signature scheme discovered recently. From our Meta-schemes we get various variants from which some are more efficient then the already known ones. They can be recommended for practical use. Then we give interesting applications of the presented Meta-schemes like authentic encryption schemes, key distribution protocols and authentication schemes. Again, we can extract highly efficient variants.

1. Introduction

The concept of signature schemes giving message recovery has been proposed in 1978 [RiSA78]. In these schemes the message has to satisfy a given redundancy scheme (e.g. the english language) and can be recovered from the signature parameters. This has the advantage, that the signed message doesn't contain the message explicitly. The drawback of this concept is, that we can't sign a hash value of a large message.

There have been several approaches in the past to obtain signature schemes with message recovery based on the discrete logarithm problem [NyR193, NyR293, NyRu94, Pive93, HoP194, HoP294]. In this paper we integrate all these approaches into a Meta-Message recovery scheme by applying the ideas of the Meta-ElGamal signature scheme [HoP194, HMP394].

The concept of blind signature schemes was introduced by Chaum in 1982 [Chau82]. These schemes can be used in payment systems [Chau85] or for electronic voting schemes [FuOO92]. In a blind signature scheme an owner Alice wants to obtain a digital signature on a message

she signs. If she gets the message and the signature later, it must not be possible that Nancy can find a relationship between some blinded and unblinded parameters. Additionally there are other classes of blind signatures, hidden or weak blind signature [HoP394], in which the notary can find a relationship between some blinded and unblinded parameters if she gets a signed unblinded message.

Recently blind ElGamal based signature schemes and blind message recovery signature schemes were introduced [CaPS94]. We show how to generalize them using the ideas of the Meta-ElGamal and the Meta-Message recovery signature scheme. Note that this can be done with many but not all variants, e.g. it doesn't apply to the original ElGamal signature scheme. Then we give some interesting applications of these Meta-schemes like authentic encryption schemes, key distribution protocols and authentication schemes.

We first give a brief review of the Meta-ElGamal signature scheme and present the Meta-Message recovery scheme. After this we present Meta-ElGamal blind signatures and Meta-Message recovery blind signatures. Then we discuss the most efficient variants of both blind Meta-schemes and continue with some useful applications of all Meta-schemes.

2. The Meta-ElGamal signature scheme

The Meta-ElGamal signature scheme has been proposed in [HMP394].

The basic ElGamal signature scheme

For an ElGamal signature [ElGa84, ElGa85] the trusted authority chooses a large prime p and a generator $\alpha \in \mathbf{Z}_p^*$ with order $p-1$. p and α are public system parameters and authentically known to all users. The signer *Alice* chooses a random number $x_A \in \mathbf{Z}_{p-1}$ and computes $y_A := \alpha^{x_A} \pmod p$. She publishes y_A and keeps x_A secret. These values are constant for all messages to be signed. To sign a message $m \in \mathbf{Z}_{p-1}$ Alice chooses a random number $k \in \mathbf{Z}_{p-1}^*$. She computes $r := \alpha^k \pmod p$ and solves the congruence

$$m \equiv x_A r + ks \pmod{p-1} \tag{1}$$

for the parameter s. The triple $(m; r, s)$ is the signed message. It can be verified by checking the congruence

$$\alpha^m \equiv y_A^r r^s \pmod p. \tag{2}$$

The Meta-ElGamal signature scheme

Instead of signature generation by the equation (1) we can also choose the general equation

$$A \equiv x_A B + kC \pmod q \tag{3}$$

with $q \in \mathbf{P}$, $q|(p-1)$, and choose A, B, C as general functions $e, f, g : {\mathbf{Z}_q}^3 \to \mathbf{Z}_q$ with arguments m, r and s. As $m \in \mathbf{Z}_{p-1}$ we imply that m is reduced modulo q before it is used as an argument but in the following description we omit this for the sake of clearness.

The parameter s should either be used as argument in only one of the three functions or the functions have to be chosen carefully, such that the signature equation can be solved. Also all of the parameters m, r, s have to occur at least once. If two or three functions use exactly the same arguments, then they should be chosen as different operations. The occurrence of the insecure $rs-$ and $ms-$variants [HMP394], where the parameters r and s (m and s) occur exactly in one of the three functions e, f and g together but neither r nor s (m nor s) occurs in one of the two other, should be avoided. All four conditions apply also for equivalent variants, in which the signature equations can be transformed into each other. Furthermore none of the three functions should be equal to zero. To get efficient variants, in the functions should be chosen, such that s can be easily extracted (e.g. without inversion). It's also an advantage to choose one of the functions equal to one, to obtain an efficient signature verification. This verification is done by checking the equation

$$\alpha^A \equiv y_A^B r^C \pmod p. \tag{4}$$

of permutations, namely to choose A, B, C as a permutation of one of the following five types EG I – EG V, which have been analyzed in detail in [HMP394]:

$$\text{EG I: } (m, r, s), \text{EG II: } (f(m, r), s, 1), \text{EG III: } (f(m, r), g(m, s), 1),$$
$$\text{EG IV: } (f(m, r), g(r, s), 1), \text{EG V: } (f(m, s), g(r, s), 1).$$

The functions $f, g : {\mathbf{Z}_q}^2 \rightarrow \mathbf{Z}_q$ have to be invertible in the argument s to guarantee the solubility of the general signature equation (3) for the signature parameter s.

For every type we get one of the following six permutations of the coefficients, which are enumerated by $No.\ 1-6$:

$$1: (a, b, c) \quad 2: (a, c, b) \quad 3: (c, b, a)$$
$$4: (c, a, b) \quad 5: (b, c, a) \quad 6: (b, a, c)$$

For example $(a, b, c) = (m, r, s)$ in $Type$ EG I and $(a, b, c) = (f(m, r), s, 1)$ in $Type$ EG II. We can use more general $(+A, +B, +C), (+A, +B, -C), (+A, -B, +C), (+A, -B, -C)$ instead of (A, B, C) in the signature equation and refer to them with $\sigma(1), \sigma(2), \sigma(3)$ and $\sigma(4)$.

Additionally we can generalize the computation of the parameter r by choosing $r' := \alpha^k \pmod p$ and computing $r := d(r', m)$ with a suitable function $d : {\mathbf{Z}_p}^2 \rightarrow \mathbf{Z}_p$.

It also possible to vary the mode of operation that determines the group orders and the length of the parameters [HMP394]:

XL: ElGamal mode with $|p| = |q| = 512$,

L: Schnorr mode [Schn89, Schn91] with $|p| = 512, |q| = 160$,

M: DSA mode [NIST91] with $|p| = 512, |q| = 160$, r reduced modulo q, and

S: small mode [Schn89, Knob94] with $|p| = 512, |q| = 160$ and a q_1 bit number $h(r)$ $(50 \leq |q_1| \leq 160)$ reduced by any hash function h.

Some generalizations have already been proposed by Schnorr [Schn91]. They can be embedded in the Meta-ElGamal scheme [HMP294]. All these generalizations can also be applied to the ElGamal signature scheme with two message blocks ($Type$ EG VI – EG X) and the signature scheme with three message blocks ($Type$ EG XI) [ElGa84, HMP394]. Combining the described variations we get the Meta-ElGamal signature scheme which can be written as

$$MEG = (Mode.Type.No.\sigma, d, e, f, g).$$

The parameters are chosen in the following way:

- $Mode \in \{\text{XL, L, M, S}\}$ gives the mode of operation,
- $No \in \{1, 2, 3, 4, 5, 6\}$ gives the number of the permutation,
- $Type \in \{\text{EG I, EG II, } \ldots, \text{EG XI}\}$ gives the type of permutation,
- $\sigma \in \{\sigma(1), \sigma(2), \sigma(3), \sigma(4)\}$ fixes the signs,
- $d : {\mathbf{Z}_p}^2 \rightarrow \mathbf{Z}_p$ specifies the computation of r,
- $e, f, g : {\mathbf{Z}_q}^3 \rightarrow \mathbf{Z}_q$ invertible in the argument s.

In a simplified manner, we can also describe the Meta-ElGamal scheme by the tuple $(Mode, d, e, f, g)$ but then we loose useful structural information for the security analysis. Therefore we prefer the first notation even if it contains redundancy.

The basic Message recovery scheme
This scheme has been proposed by Nyberg and Rueppel [NyR193]. Let p and q be primes with $q|(p-1)$. Let $\alpha \in \mathbf{Z}_p^*$ be an element of order q. The signer *Alice* chooses x_A and y_A as in the ElGamal scheme. To sign the message $m \in \mathbf{Z}_{p-1}$, satisfying a redundancy scheme, she chooses a random $k \in \mathbf{Z}_q^*$, computes $r := \alpha^{-k} m \pmod p$ and solves the equation $s := k - x_A r \pmod q$. The tuple (r, s) is the signature on the message m, which can be recovered by computing $m := \alpha^s y_A^r r \pmod p$.

3.1 The Meta-Message recovery scheme for one message block

To develop a signature scheme giving message recovery from the Meta-ElGamal signature schemes, we can use the general message recovery approach, which has been described in [NyRu94]:

1. Multiply the exponential (or its inverse) in the commitment r with the message m (or m^{-1}),
2. replace the message m by 1 in equation (3),
3. rebuild the verification equation, such that the exponential α^k is computed and the message can be recovered from the commitment part r of the signature.

We can apply this approach to the Meta-ElGamal scheme.

General functions
Instead of computing $r' := \alpha^k \pmod p$ and $r := (r')^{-1} m \pmod p$ we can apply a general function $d : {\mathbf{Z}_p}^2 \to \mathbf{Z}_p$ to the arguments r', m, such that

$$r' := \alpha^k \pmod p,\ r := d(r', m),$$

where d is invertible in the second argument, that is $m := d^{-1}(r, r')$. The general signature equation is of the form

$$A \equiv x_A B + kC \pmod q \tag{5}$$

with A, B, C permutations of the general functions $e, f, g : {\mathbf{Z}_q}^2 \to \mathbf{Z}_q$ with arguments r and s. The message recovery can be done by verifying the equation

$$m \equiv d^{-1}\left(r, \alpha^{AC^{-1}} y_A^{-BC^{-1}} \pmod p\right) \tag{6}$$

and checking if m satisfies the given redundancy scheme. The correctness of the scheme can be verified by the following congruence:

$$\begin{aligned} d^{-1}\left(r, \alpha^{AC^{-1}} y_A^{-BC^{-1}} \pmod p\right) &\equiv d^{-1}\left(r, \alpha^{(A - x_A B)C^{-1}} \pmod p\right) \\ &\equiv d^{-1}\left(r, \alpha^{kCC^{-1}} \pmod p\right) \equiv d^{-1}(r, r') = m. \end{aligned}$$

Type of equation
If we look carefully on the necessary conditions on the functions e, f, g which are the same as described in chapter 2, we see that we get the following ten types of permutations:

Type	$(\pm A, \pm B, \pm C)$ permutation of		
MR I	1	r	s
MR II	1	s	$f(r,s)$
MR III	1	r	$f(r,s)$
MR IV	1	$f(r,s)$	$g(r,s)$
MR V	r	r	$f(r,s)$

Type	$(\pm A, \pm B, \pm C)$ permutation of		
MR VI	r	s	$f(r,s)$
MR VII	s	s	$f(r,s)$
MR VIII	r	$f(r,s)$	$g(r,s)$
MR IX	s	$f(r,s)$	$g(r,s)$
MR X	$e(r,s)$	$f(r,s)$	$g(r,s)$

efficient types are *Type* MR I – IV if we choose the parameter $C = 1$, because we need no inversion during message recovery. In *Type* MR II, IV, VI – X we have to choose suitable functions e, f, g to guarantee the solvability for the parameter s. In *Type* MR IV, we have to choose different functions f, g without homomorphic properties to guarantee the security of the signature scheme. *Type* MR I has been obtained from *Type* EG I of the Meta-ElGamal scheme, *Type* MR II from *Type* EG IV and *Type* MR III from *Type* EG V. The other types result from various variants of the Meta-ElGamal scheme which haven't been enumerated yet [HMP394].

Table 1 gives an overview about all permutations of the first four types with $d(r', m) = (r')^{-1}m \pmod p$, where we find the most efficient variants ($C = 1$). Variant MR I.3 has first been proposed in [NyR193], variant MR I.2 in [Pive93], variants MR I.1 and MR I.5 in [NyR293] and independently in [HoP294] and variants MR I.4 and MR I.6 in [NyR293, NyRu94].

No.	A	$-B$	C	signature	message recovery
MR I.1	1	r	s	$1 \equiv -x_A r + ks$	$m \equiv \alpha^{s^{-1}} y_A^{rs^{-1}} r$
MR I.2	1	s	r	$1 \equiv -x_A s + kr$	$m \equiv \alpha^{r^{-1}} y_A^{sr^{-1}} r$
MR I.3	s	r	1	$s \equiv -x_A r + k$	$m \equiv \alpha^{s} y_A^{r} r$
MR I.4	s	1	r	$s \equiv -x_A + kr$	$m \equiv \alpha^{sr^{-1}} y_A^{r^{-1}} r$
MR I.5	r	s	1	$r \equiv -x_A s + k$	$m \equiv \alpha^{r} y_A^{s} r$
MR I.6	r	1	s	$r \equiv -x_A + ks$	$m \equiv \alpha^{rs^{-1}} y_A^{s^{-1}} r$
MR II.1	s	$f(r,s)$	1	$s \equiv -x_A f(r,s) + k$	$m \equiv \alpha^{s} y_A^{f(r,s)} r$
MR II.2	s	1	$f(r,s)$	$s \equiv -x_A + kf(r,s)$	$m \equiv \alpha^{sf(r,s)^{-1}} y_A^{f(r,s)^{-1}} r$
MR II.3	1	$f(r,s)$	s	$1 \equiv -x_A f(r,s) + ks$	$m \equiv \alpha^{s^{-1}} y_A^{f(r,s)s^{-1}} r$
MR II.4	1	s	$f(r,s)$	$1 \equiv -x_A s + kf(r,s)$	$m \equiv \alpha^{f(r,s)^{-1}} y_A^{sf(r,s)^{-1}} r$
MR II.5	$f(r,s)$	1	s	$f(r,s) \equiv -x_A + ks$	$m \equiv \alpha^{f(r,s)s^{-1}} y_A^{s^{-1}}$
MR II.6	$f(r,s)$	s	1	$f(r,s) \equiv -x_A s + k$	$m \equiv \alpha^{f(r,s)} y_A^{s} r$
MR III.1	r	1	$f(r,s)$	$r \equiv -x_A + kf(r,s)$	$m \equiv \alpha^{rf(r,s)^{-1}} y_A^{f(r,s)^{-1}} r$
MR III.2	r	$f(r,s)$	1	$r \equiv -x_A f(r,s) + k$	$m \equiv \alpha^{r} y_A^{f(r,s)} r$
MR III.3	$f(r,s)$	1	r	$f(r,s) \equiv -x_A + kr$	$m \equiv \alpha^{f(r,s)r^{-1}} y_A^{r^{-1}}$
MR III.4	$f(r,s)$	r	1	$f(r,s) \equiv -x_A r + k$	$m \equiv \alpha^{f(r,s)} y_A^{r} r$
MR III.5	1	$f(r,s)$	r	$1 \equiv -x_A f(r,s) + kr$	$m \equiv \alpha^{r^{-1}} y_A^{f(r,s)r^{-1}} r$
MR III.6	1	r	$f(r,s)$	$1 \equiv -x_A r + kf(r,s)$	$m \equiv \alpha^{f(r,s)^{-1}} y_A^{rf(r,s)^{-1}} r$
MR IV.1	$f(r,s)$	$g(r,s)$	1	$f(r,s) \equiv -x_A g(r,s) + k$	$m \equiv \alpha^{f(r,s)} y_A^{g(r,s)} r$
MR IV.2	$f(r,s)$	1	$g(r,s)$	$f(r,s) \equiv -x_A + kg(r,s)$	$m \equiv \alpha^{f(r,s)g(r,s)^{-1}} y_A^{g(r,s)^{-1}} r$
MR IV.3	1	$g(r,s)$	$f(r,s)$	$1 \equiv -x_A g(r,s) + kf(r,s)$	$m \equiv \alpha^{f(r,s)^{-1}} y_A^{g(r,s)f(r,s)^{-1}} r$
MR IV.4	1	$f(r,s)$	$g(r,s)$	$1 \equiv -x_A f(r,s) + kg(r,s)$	$m \equiv \alpha^{g(r,s)^{-1}} y_A^{f(r,s)g(r,s)^{-1}} r$
MR IV.5	$g(r,s)$	1	$f(r,s)$	$g(r,s) \equiv -x_A + kf(r,s)$	$m \equiv \alpha^{g(r,s)f(r,s)^{-1}} y_A^{f(r,s)^{-1}}$
MR IV.6	$g(r,s)$	$f(r,s)$	1	$g(r,s) \equiv -x_A f(r,s) + k$	$m \equiv \alpha^{g(r,s)} y_A^{f(r,s)} r$

Table 1: Message recovery for one message block

Mode of operation

We also have to consider different modes of operation for the message recovery schemes. The *Mode* XL in which $|p| = |q| = 512$ is not very efficient. The *Mode* L has been referred as ElGamal* in [NyRu94]. For *Mode* M and S we get two different possibilities of computing the parameter r (of length q_1) as pointed out in [NyRu94] for *Mode* M:

1. $m \in \mathbf{Z}_{q_1}$, $d : \mathbf{Z}_{q_1}^2 \rightarrow \mathbf{Z}_{q_1}$, $r' := h\left(\alpha^k \pmod p\right)$, $r := d(r', m) = d\left(h(\alpha^k \pmod p)\right), m)$,
2. $m \in \mathbf{Z}_{p-1}$, $d : \mathbf{Z}_p^2 \rightarrow \mathbf{Z}_p$, $r' := \alpha^k \pmod p$, $\quad r := h\left(d(r', m)\right) = h\left(d(\alpha^k, m)\right)$.

$$m := d^{-1}\left(r, h(\alpha^{AC^{-1}}(y_A)^{-BC^{-1}})\right).$$

As the signed message is very small in this case we won't consider it any more.

The second variant was proposed by Schnorr [Schn89] and for *Mode* M by Nyberg and Rueppel and has been generalized in the seventh generalization of the Meta-ElGamal signature scheme in [HMP394]. It doesn't give message recovery and can thus be used only in text hashing mode (where we have to transmit m additionally) like the efficient DSA-variants proposed in [HMP394]. Summarizing the above results, we see that *Mode* L is best suited for message recovery schemes, because the expansionrate of the signature is minimal.

The Meta-scheme:
Combining the described variations we get the Meta-Message recovery scheme (MMR) for one message block which can be written as

$$\text{MMR}_1 = (Mode.Type.No.\sigma, d, e, f, g).$$

The parameters can be chosen out of the following:

- $Mode \in \{\text{XL, L, M, S}\}$ gives the mode of operation,
- $No \in \{1, 2, 3, 4, 5, 6\}$ gives the number of the permutation,
- $Type \in \{\text{MR I, MR II, ..., MR X}\}$ gives the type of permutation,
- $\sigma \in \{\sigma(1), \sigma(2), \sigma(3), \sigma(4)\}$ fixes the signs,
- $d : \mathbf{Z}_p{}^2 \to \mathbf{Z}_p$ invertible in the argument m,
- $e, f, g : \mathbf{Z}_q{}^2 \to \mathbf{Z}_q$ invertible in the argument s.

3.1.1 Security of the Meta-Message recovery scheme

The security of the Meta-Message recovery scheme is similar to the $mr-$variants of the Meta-ElGamal signature scheme, which has been analyzed in detail in [HMP394]. If we are substituting the parameter r in the signature equation by the function $d(r', m)$ then we see, that the equations in *Type* MR I are like the equations in *Type* EG II if we choose $d = f$. The same property holds for all other types, only the corresponding types in the Meta-ElGamal scheme haven't been enumerated yet.

The security analysis for a total break of the signature scheme and universal forgery of messages can be adapted from the Meta-ElGamal scheme. Only the existential forgery has to be considered again, because we get some obvious attacks, as described in [NyR193].

1. An attacker can choose signature parameters r, s at random and calculate the corresponding message m by the message recovery equation (6). To avoid this attack, the message m should be in a redundancy scheme, such that the probability of success for such an attack is negligible.

2. For some variants one can compute valid signatures $(r, s + t)$ from a given valid signature $S(m) = (r, s)$. For example the following equations hold

   $$\text{MR I.2:}\quad S\left(m(y_A^r{}^{-1})^t\right) = (r, s+t) \qquad \text{MR I.3:}\quad S(m\alpha^t) = (r, s+t)$$
   $$\text{MR I.4:}\quad S\left(m(\alpha^r{}^{-1})^t\right) = (r, s+t) \qquad \text{MR I.5:}\quad S(my_A^t) = (r, s+t)$$

   For special choices of the functions f and g this property also exists for some variants of the other types of signature schemes. This corresponds to the homomorphic property of the RSA signature scheme.

no two of the above messages are within this scheme.

Equivalent security of the variants

As already mentioned in [NyRu94, HMP394] some of the variants offer equivalent security. For arbitrary choices of the functions d, e, f, g the equations

- 1 and 6,
- 2 and 4,
- 3 and 5,

for all types provide equivalent security, because only the roles of the generator α and the public key y_A are changed. Additionally, some variants of the Meta-Message recovery scheme are strongly equivalent [NyRu94] to other variants of the Meta-ElGamal scheme for some special choices of the function d. For example for a message $m \in \mathbf{Z}_q$ in $Mode$ S, if we choose $d(r', m) = r'm^{-1}$ then $Type$ MR I is strongly equivalent to $Type$ EG I: If (r, s) is a signature for m in $Type$ EG I then $(rm^{-1} \pmod q, sm^{-1} \pmod q)$ is a signature for m in $Type$ MR I. Conversely, given a signature (r, s) in MR I, we first recover m and obtain the signature $(rm \pmod q, sm \pmod q)$ in EG I.

3.1.2 Performance of the Meta-Message recovery scheme

The most efficient variants are those, in which Alice doesn't need to compute any inversion modulo q for signature generation and additionally Bob doesn't need to compute any inverse for message recovery. These conditions are satisfied for variants MR I.3, MR II.1, MR III.4 if we choose a suitable function f, for which the inverse can be computed without computing multiplicative inverses modulo q.

3.2 The Meta-Message recovery scheme for two message blocks

The ideas of the Meta-ElGamal scheme can also be applied for two message blocks using the general construction principle described above. In this case, only one message block m_1 can be recovered, the other one m_2 has to be transmitted together with the signature parameters r, s. Thus we can put for example a hash value $h(m_1)$ and the identity ID_A of user A into this second block. This value has to be transferred in any case, by this approach it's already authenticated. This can be useful for e-mail distributions of messages. The block m_1 needs no longer to be within a suitable redundancy scheme and can be chosen at random in $\mathbf{Z}_{p-1}$ for $Mode$ L. If we choose for example a t bit hash value and l bits for the identity, then the message block m_2 can be submitted as a $l + t$ bit message. We can also add additional information, like timestamps to give a date of expire for the message.

The general signature equation is nearly the same as in section 3.1 with the difference that the functions $e, f, g : {\mathbf{Z}_q}^3 \to \mathbf{Z}_q$ use the arguments m_2, r, s. If we consider the efficient variants, we have to choose one function equal to 1. Hence we get the remaining ten types of permutations if we also consider the necessary conditions on the functions e, f, g described in chapter 2 .

$Type$	$(\pm A, \pm B, \pm C)$ permutation of		
MR XI	$f(m_2, r)$	s	1
MR XII	$f(m_2, r)$	$g(m_2, s)$	1
MR XIII	$f(m_2, r)$	$g(r, s)$	1
MR XIV	$f(m_2, s)$	$g(r, s)$	1
MR XV	$f(m_2, s)$	$g(m_2, r, s)$	1

$Type$	$(\pm A, \pm B, \pm C)$ permutation of		
MR XVI	$f(m_2, r)$	$g(m_2, r, s)$	1
MR XVII	$f(m_2, r, s)$	r	1
MR XVIII	$f(m_2, r, s)$	s	1
MR IXX	$f(m_2, r, s)$	$g(r, s)$	1
MR XX	$f(m_2, r, s)$	$g(m_2, r, s)$	1

The types MR XI, XII, XIII, XVI, XVII are solvable for all possible choices of f, g. The most efficient variants are those in which the functions f, g have only two arguments and the parameter $C = 1$, this is respective the case for two variants of $Type$ MR XI – XIV which are given in the following table 2.

No.	$\pm A$	$\mp B$	signature	message recovery
MR XI.3	s	$f(m_2,r)$	$s \equiv -x_A f(m_2,r) + k$	$m_1 \equiv \alpha^s y_A^{f(m_2,r)} r$
MR XI.6	$f(m_2,r)$	s	$f(m_2,r) \equiv -x_A s + k$	$m_1 \equiv \alpha^{f(m_2,r)} y_A^s r$
MR XII.3	$g(m_2,s)$	$f(m_2,r)$	$g(m_2,s) \equiv -x_A f(m_2,r) + k$	$m_1 \equiv \alpha^{g(m_2,s)} y_A^{f(m_2,r)} r$
MR XII.6	$f(m_2,r)$	$g(m_2,s)$	$f(m_2,r) \equiv -x_A g(m_2,s) + k$	$m_1 \equiv \alpha^{f(m_2,r)} y_A^{g(m_2,s)} r$
MR XIII.2	$f(m_2,r)$	$g(r,s)$	$f(m_2,r) \equiv -x_A g(r,s) + k$	$m_1 \equiv \alpha^{f(m_2,r)} y_A^{g(r,s)} r$
MR XIII.4	$g(r,s)$	$f(m_2,r)$	$g(r,s) \equiv -x_A f(m_2,r) + k$	$m_1 \equiv \alpha^{g(r,s)} y_A^{f(m_2,r)} r$
MR XIV.1	$f(m_2,s)$	$g(r,s)$	$f(m_2,s) \equiv -x_A g(r,s) + k$	$m_1 \equiv \alpha^{f(m_2,s)} y_A^{g(r,s)} r$
MR XIV.6	$g(r,s)$	$f(m_2,s)$	$g(r,s) \equiv -x_A f(m_2,s) + k$	$m_1 \equiv \alpha^{g(r,s)} y_A^{f(m_2,s)} r$

Table 2: Efficient variants for Message recovery with two message blocks

If we choose one coefficient out of the set

$$\{r, s, e(r,s), e(m_2,r), e(m_2,s), e(m_2,r,s)\}$$

instead of the coefficient 1, we get further 60 types (MR XXIII – MR LXXXII) which are not very efficient.

Combining the described variations we get the Meta-Message recovery scheme for two message blocks which can be written as

$$\mathrm{MMR}_2 = (Mode.Type.No.\sigma, d, e, f, g).$$

The parameters are the same as for one message block, except that $Type$ can be chosen out of {MR XI, MR XII, ..., MR XX, MR XXIII, ..., MR LXXX}.

3.3 The Meta-Message recovery scheme for three message blocks

In the case of three message blocks we can also recover only one message block m_1, such that we have to transmit the other two blocks m_2, m_3 together with the signature. The general signature equation is as above in section 3.1 with the difference that the functions $e, f, g : {\mathbf{Z}_q}^4 \rightarrow \mathbf{Z}_q$ have the arguments m_2, m_3, r, s. If we consider only the efficient variants we should choose one function equal to 1, such that we can choose the parameter $C = 1$ and don't need any inversion for message recovery. Among these variants the following one in table 3 is most efficient with suitable functions f, g. Because we can also apply the same attack as for the $ms-$ variant in the Meta-ElGamal signature scheme [HMP394] in which an attacker can universally forge message blocks m_3 with the knowledge of one signed message (m_1, m_2, m_3, r, s) we have to choose for example the last t bit of the message block m_2 as the hash value $h(m_1, m_2', m_3)$ of a suitable hash function (m_2' denotes a $|p| - t$ bit message, $m_2 = m_2' || h(m_1, m_2', m_3)$ and the parameter $t \in \mathbf{Z}_{p-1}$ an additional security parameter). The most secure scheme is the one, in which we choose $\pm A, \pm B, \pm C$ as a permutation of $m_2, f(m_3, r)$ and s. In this case an attacker can only start an existential forgery on the message block m_3, which can be prevented either by the choice of a suitable hash function or a redundancy scheme. This variant is also given in the following table 3:

4. Meta-ElGamal blind signatures

The basic scheme

We first give a short review of the first blind signature scheme, which has been presented by Okamoto in 1992 [Okam92]. Okamotos approach is best suited for signature schemes, where the message is only hidden in the function d, that means m doesn't appear as argument in the functions e, f and g. This is the case in Schnorr's signature scheme [Schn89], which is the basis for Okamoto's protocol.

No.	A	$-B$	C	signature	message recovery
MR XXI.1	1	$f(m_2,r)$	$g(m_3,s)$	$1 \equiv -x_A f(m_2,r) + k g(m_3,s)$	$\alpha^{g(m_3,s)^{-1}} y_A^{f(m_2,r)g(m_3,s)^{-1}} r$
MR XXI.2	1	$g(m_3,s)$	$f(m_2,r)$	$1 \equiv -x_A g(m_3,s) + k f(m_2,r)$	$\alpha^{f(m_2,r)^{-1}} y_A^{g(m_3,s)f(m_2,r)^{-1}} r$
MR XXI.3	$g(m_3,s)$	$f(m_2,r)$	1	$g(m_3,s) \equiv -x_A f(m_2,r) + k$	$\alpha^{g(m_3,s)} y_A^{f(m_2,r)} r$
MR XXI.4	$g(m_3,s)$	1	$f(m_2,r)$	$g(m_3,s) \equiv -x_A + k f(m_2,r)$	$\alpha^{g(m_3,s)f(m_2,r)^{-1}} y_A^{f(m_2,r)^{-1}} r$
MR XXI.5	$f(m_2,r)$	1	$g(m_3,s)$	$f(m_2,r) \equiv -x_A + k g(m_3,s)$	$\alpha^{f(m_2,r)g(m_3,s)^{-1}} y_A^{g(m_3,s)^{-1}} r$
MR XXI.6	$f(m_2,r)$	$g(m_3,s)$	1	$f(m_2,r) \equiv -x_A g(m_3,s) + k$	$\alpha^{f(m_2,r)} y_A^{g(m_3,s)} r$
MR XXII.1	m_2	$f(m_3,r)$	s	$m_2 \equiv -x_A f(m_3,r) + ks$	$\alpha^{m_2 s^{-1}} y_A^{f(m_3,r)s^{-1}} r$
MR XXII.2	m_2	s	$f(m_3,r)$	$m_2 \equiv -x_A s + k f(m_3,r)$	$\alpha^{m_2 f(m_3,r)^{-1}} y_A^{s f(m_3,r)^{-1}} r$
MR XXII.3	s	$f(m_3,r)$	m_2	$s \equiv -x_A f(m_3,r) + k m_2$	$\alpha^{s m_2^{-1}} y_A^{f(m_3,r)m_2^{-1}} r$
MR XXII.4	s	m_2	$f(m_3,r)$	$s \equiv -x_A m_2 + k f(m_3,r)$	$\alpha^{s f(m_3,r)^{-1}} y_A^{m_2 f(m_3,r)^{-1}} r$
MR XXII.5	$f(m_3,r)$	m_2	s	$f(m_3,r) \equiv -x_A m_2 + ks$	$\alpha^{f(m_3,r)s^{-1}} y_A^{m_2 s^{-1}} r$
MR XXII.6	$f(m_3,r)$	s	m_2	$f(m_3,r) \equiv -x_A s + k m_2$	$\alpha^{f(m_3,r)m_2^{-1}} y_A^{s m_2^{-1}} r$

Table 3: Message recovery for three message blocks

Okamoto's blind Schnorr signatures		
Owner *Alice*	Channel	Notary *Nancy*
$a, b \in \mathbf{Z}_q$		$\tilde{k} \in \mathbf{Z}_p^*$
$\tilde{r}'$	$\longleftarrow$	$\tilde{r}' := \alpha^{\tilde{k}} \pmod p$
$r' := \tilde{r}' y_N^{-a} \alpha^b \pmod p$		
$r := h(r', m)$		
$\tilde{r} := r + a$	$\longrightarrow$	$\tilde{r}$
$\tilde{s}$	$\longleftarrow$	$\tilde{s} := x_N \tilde{r} + \tilde{k} \pmod q$
$s := \tilde{s} + b \pmod q$		

The signature on the message m is given by (r, s). It's verification can be done by checking the equation

$$h(\alpha^s y_N^{-r}, m) = r.$$

This equation is true, because of the following congruence:

$$h(\alpha^s y_N^{-r} \pmod p, m) = h(\alpha^{\tilde{k}+\tilde{r}x_N+b}\alpha^{-x_N r} \pmod p, m) = h(\alpha^{\tilde{k}-ax_N+b}, m) = h(r', m) = r$$

The Meta-scheme

Now we present the Meta-ElGamal based blind signature scheme founded on the ElGamal based blind signature scheme in [CaPS94]. For the sake of clearness the function d is chosen as $d(r, m) := r$ and we only focus on the *Mode* L. The adoption to the other modes is straightforward.

The idea is that notary Nancy chooses the blinded parameter $\tilde{r} := \alpha^{\tilde{k}} \pmod p$ herself (with a random $\tilde{k} \in \mathbf{Z}_q{}^*$) and the owner Alice chooses the unblinded $r := \tilde{r}^a \alpha^b \pmod p$ (with random $a, b \in \mathbf{Z}_q{}^*$). Nancy signs the blinded message $\tilde{m}$ using the equation

$$\tilde{A} \equiv x_N \tilde{B} + \tilde{k}\tilde{C} \pmod q, \tag{7}$$

which is equivalent to $\tilde{k} \equiv \tilde{C}^{-1}(\tilde{A} - x_N \tilde{B}) \pmod q$ where x_N is the secret key of Nancy. The unblinded signed message is given by (m, r, s). Its validity is checked by the equation

$$\alpha^A \equiv y_N^B r^C \pmod p. \tag{8}$$

For the correctness of the signature scheme, it is necessary, that this equation is satisfied.

Thus we have

$$y_N^B r^C \equiv \alpha^{x_N B}\alpha^{(a\tilde{k}+b)C} \equiv \alpha^{x_N B + C(a\tilde{C}^{-1}(\tilde{A}-x_N\tilde{B})+b)} \equiv y_N^{(B-a\tilde{C}^{-1}\tilde{B}C)}\alpha^{a\tilde{C}^{-1}\tilde{A}C+bC} \pmod p$$

$$A = a\tilde{A}C\tilde{C}^{-1} + bC \pmod{q} \quad (9)$$

$$B = a\tilde{B}C\tilde{C}^{-1} \pmod{q} \quad (10)$$

If the value s does not appear in C then it is possible to transform these two equations to get $\tilde{m} := \psi(a, b, m, r, \tilde{r})$ and $s := \theta(a, b, m, r, \tilde{m}, \tilde{r}, \tilde{s})$. Note that s or $\tilde{s}$ are not allowed in the equation for $\tilde{m}$. Furthermore we can transform the signature equation (7) to get $\tilde{s} := \rho(x_N, \tilde{k}, \tilde{m}, \tilde{r})$. Hence follows the Meta-ElGamal blind signature scheme:

Meta-ElGamal blind signature scheme		
parameter: p, q prime, α generator, m message		
Owner *Alice*	Channel	Notary *Nancy*
$a, b \in_R \mathbf{Z}_q^*$		$\tilde{k} \in_R \mathbf{Z}_p{}^*$
$\tilde{r}$	$\longleftarrow$	$\tilde{r} := \alpha^{\tilde{k}} \pmod{p}$
$r := \tilde{r}^a \alpha^b \pmod{p}$		
$\tilde{m} := \psi(a, b, m, r, \tilde{r})$	$\longrightarrow$	$\tilde{m}$
$\tilde{s}$	$\longleftarrow$	$\tilde{s} := \rho(x_N, \tilde{k}, \tilde{m}, \tilde{r})$
$s := \theta(a, b, m, r, \tilde{m}, \tilde{r}, \tilde{s})$		

The signature on the message m is given by (r, s). Its verification can be done by checking the equation

$$\alpha^A \equiv y_N^B r^C \pmod{p}.$$

We illustrate the Meta-scheme by giving equations for some efficient variants in table 4.

No.	equation (9)	equation (10)
MB I.2	$m \equiv a\tilde{r}^{-1}\tilde{m}r + br$	$s \equiv a\tilde{r}^{-1}\tilde{s}r$
MB I.3	$s \equiv a\tilde{m}^{-1}\tilde{s}m + bm$	$r \equiv a\tilde{m}^{-1}\tilde{r}m$
MB I.4	$s \equiv a\tilde{r}^{-1}\tilde{s}r + br$	$m \equiv a\tilde{r}^{-1}\tilde{m}r$
MB I.5	$r \equiv a\tilde{m}^{-1}\tilde{r}m + bm$	$s \equiv a\tilde{m}^{-1}\tilde{s}m$
MB II.2	$1 \equiv af(\tilde{m}, \tilde{r})^{-1}f(m, r) + bf(m, r)$	$s \equiv af(\tilde{m}, \tilde{r})^{-1}\tilde{s}f(m, r)$
MB II.3	$s \equiv a\tilde{s} + b$	$f(m, r) \equiv af(\tilde{m}, \tilde{r})$
MB II.4	$s \equiv af(\tilde{m}, \tilde{r})^{-1}\tilde{s}f(m, r) + bf(m, r)$	$1 \equiv af(\tilde{m}, \tilde{r})^{-1}f(m, r)$
MB II.5	$f(m, r) = af(\tilde{m}, \tilde{r}) + b$	$s \equiv a\tilde{s}$
MB III.2	$1 \equiv ag(\tilde{m}, \tilde{s}) + bf(m, s)$	$g(m, s) \equiv af(\tilde{m}, \tilde{r})^{-1}g(\tilde{m}, \tilde{s})f(m, r)$
MB III.3	$g(m, s) \equiv ag(\tilde{m}, \tilde{s}) + b$	$f(m, r) \equiv af(\tilde{m}, \tilde{r})$
MB III.4	$g(m, s) \equiv af(\tilde{m}, \tilde{r})^{-1}g(\tilde{m}, \tilde{s})f(m, s) + bf(m, s)$	$1 \equiv af(\tilde{m}, \tilde{r})^{-1}f(m, r)$
MB III.5	$f(m, r) \equiv af(\tilde{m}, \tilde{r}) + b$	$g(m, s) \equiv ag(\tilde{m}, \tilde{r})$
MB IV.2	$1 \equiv af(\tilde{m}, \tilde{r})^{-1}f(m, r) + bf(m, r)$	$g(r, s) \equiv af(\tilde{m}, \tilde{r})^{-1}g(\tilde{r}, \tilde{s})f(m, r)$
MB IV.3	$g(r, s) \equiv ag(\tilde{r}, \tilde{s}) + b$	$f(m, r) \equiv af(\tilde{m}, \tilde{r})$
MB IV.4	$g(r, s) \equiv af(\tilde{m}, \tilde{r})^{-1}g(\tilde{r}, \tilde{s})f(m, r) + bf(m, r)$	$1 \equiv af(\tilde{m}, \tilde{r})^{-1}f(m, r)$
MB IV.5	$f(m, r) \equiv af(\tilde{m}, \tilde{r}) + b$	$g(r, s) \equiv ag(\tilde{r}, \tilde{s})$
MB V.2	$1 \equiv af(\tilde{m}, \tilde{s})^{-1}f(m, s) + bf(m, s)$	$g(r, s) \equiv af(\tilde{m}, \tilde{s})^{-1}g(\tilde{r}, \tilde{s})f(m, s)$
MB V.3	$g(r, s) \equiv ag(\tilde{r}, \tilde{s}) + b$	$f(m, s) \equiv af(\tilde{m}, \tilde{s})$
MB V.4	$g(r, s) \equiv af(\tilde{m}, \tilde{s})^{-1}g(\tilde{r}, \tilde{s})f(m, s) + bf(m, s)$	$1 \equiv af(\tilde{m}, \tilde{s})^{-1}f(m, s)$
MB V.5	$f(m, s) \equiv af(\tilde{m}, \tilde{s}) + b$	$g(r, s) \equiv ag(\tilde{r}, \tilde{s})$

Table 4: Meta-ElGamal blind signature schemes

Note that for those schemes in which the parameter s appears in C we can't get blind signature schemes for general functions f and g, because s and $\tilde{s}$ are not allowed as arguments in the function ψ. Thus we can't get a blind signature scheme using the basic ElGamal signature scheme.

A signature scheme is called blind, if all (blinded) parameters which are known by Nancy are statistically independent from the unblinded parameters of the signature. If it can be shown that for any blinded and unblinded parameters there are unique a and b which are chosen at random

can show that all variants are blind signature schemes:

Theorem 1: For any pair of triples $(\tilde{m}, \tilde{s}, \tilde{r})$, (m, s, r) with $m, r, \tilde{r} \in \mathbf{Z}_p^*, \tilde{r} \equiv \alpha^{\tilde{k}} \pmod{p}$, $\tilde{A} \equiv \tilde{B} x_N + \tilde{C}\tilde{k} \pmod{q}$, $\alpha^A \equiv y_N^B r^C \pmod{p}$ and A, B, C chosen from the table above, there exist unique $a, b \in \mathbf{Z}_q$ with

$$r \equiv \tilde{r}^a \alpha^b \pmod{p} \tag{11}$$
$$A = a\tilde{C}^{-1}\tilde{A}C + bC \pmod{q} \tag{12}$$
$$B = a\tilde{C}^{-1}\tilde{B}C \pmod{q} \tag{13}$$

Proof: Choose $a, b \in \mathbf{Z}_q$ with

$$a = B\tilde{C}\tilde{B}^{-1}C^{-1} \pmod{q} \tag{14}$$
$$b = (A - \tilde{A}B\tilde{B}^{-1})C^{-1} \pmod{q} \tag{15}$$

Using the signature equation (7) from above we get

$$\begin{aligned} a\tilde{k} + b &\equiv (B\tilde{C}\tilde{B}^{-1}C^{-1})\tilde{k} + (A - \tilde{A}B\tilde{B}^{-1})C^{-1} \\ &\equiv C^{-1}(A + B(\tilde{C}\tilde{B}^{-1}\tilde{k} - \tilde{A}\tilde{B}^{-1})) \\ &\equiv C^{-1}(A + B((\tilde{A} - x_N\tilde{B})\tilde{B}^{-1} - \tilde{A}\tilde{B}^{-1})) \\ &\equiv C^{-1}(A - Bx_N) \pmod{q} \end{aligned} \tag{16}$$

Thus we have

$$\tilde{r}^a \alpha^b \equiv \alpha^{a\tilde{k}+b} \equiv \alpha^{C^{-1}(A - Bx_N)} \equiv (\alpha^A y_N^{-B})^{C^{-1}} \equiv r^{CC^{-1}} \equiv r \pmod{p}$$

The validity of the relations for a and b is trivial. Moreover the choice of a and b is unique, because congruence (16) must be satisfied. Hence the Meta-ElGamal blind signature scheme can be written as

$$\text{MEB} = (Mode.Type.No, d, e, f, g)$$

similar to the Meta-ElGamal scheme (MEG).

4.1 Security considerations

Total break of the scheme

To avoid a total break of the scheme, which means that an attacker can compute the secret key x_N of the notary Nancy, Nancy should be aware that she doesn't sign a blinded message $\tilde{m}$ if the coefficient $\tilde{B}$ or $\tilde{C}$ is equal to zero or $(p-1)/2$ in *Mode* XL.

- As already described in [HMP294], the variants of the ElGamal signature scheme can be totally broken, if the coefficient C is chosen equal to $(p-1)/2 \pmod{p}$ in *Mode* XL (or equal to $0 \pmod{q}$ in modes L, M and S). In these cases every verifier can compute the secret key x_N. The signature equation is $A \equiv x_N B + kC \pmod{p-1}$. If $C = (p-1)/2$ then this equation simplifies to $A \equiv x_N B \pmod{p-1}$ if k is even and to $A \equiv x_N B + (p-1)/2 \pmod{p-1}$ if k is odd. In both cases x_A can be computed if $\gcd(B, p-1) = 1$. In the case of $C \equiv 0 \pmod{q}$ the equation simplifies to $A \equiv x_N B \pmod{q}$, which can always be solved for x_N if $B \neq 0$.

  To avoid this kind of attack, it is necessary, that either the parameter C can't be chosen equal to $(p-1)/2$ or 0 without knowledge of the notary, or the parameter B should also be equal to $(p-1)/2$ or 0 in this case, such that x_N can't be extracted, as the condition $\gcd(B, p-1) = 0$ is not satisfied.

  If we look at the equation (10), we see that if $C = 0$ then the parameter B or $\tilde{B}$ must also be equal to zero, such that either the notary gets a zero coefficient ($(p-1)/2$ respectively) or the equation is trivial and leaks no information about x_N.

(or $B \equiv (p-1)/2$ in *Mode* XL) and $C \neq 0$. In this case, only k can be extracted, which is a random number used only once, such that there is no immediate use from this knowledge. But with the knowledge of k the parameter $\tilde{k} := (k-b)a^{-1}$ can also be computed. This can be used, to solve the blinded equation $\tilde{A} \equiv x_N \tilde{B} + \tilde{k}\tilde{C} \pmod{q}$ for the parameter x_N, if this equation is not trivial in the sense, that $\tilde{B}$ is also equal to $(p-1)/2$ or 0 in this case.

If we look again at the equation (10), we see that with the choice of $B = 0$, the coefficient $\tilde{B}$ or C must also be equal to zero. If $\tilde{B}$ is equal to zero then the notary won't sign the message, in the other case k can't be extracted as $C = 0$.

Universal and existential forgery

There are three different persons with different views who are able to cheat:

- The notary Nancy knows the blinded parameters and perhaps later some unblinded ones. She wants to find out some relationship between the blinded and unblinded parameters and she does not need to follow the protocol.
- The verifier Bob knows some unblinded and blinded parameters but not necessarily the related ones and can try to forge a signature. He cannot influence the protocol.
- The owner Alice knows the related blinded and unblinded parameters and her aim is to get more valid signatures to arbitrarily chosen messages than is allowed to get. She does not need to follow the protocol.

We have already proved that the scheme is truly blind. Thus Nancy doesn't get additional informations and is not able to cheat.

The blinded parameters don't help verifier Bob because as we have seen there are unique parameters a, b such that a blinded and an unblinded triple of signature parameters correspond to each other. Hence we get no further informations from the unblinded parameters and can reduce this case to the Meta-ElGamal scheme. This type of possible cheating has already been analyzed in [HMP394].

Last we examine the case that Alice try to cheat. If she follows the protocol then we can reduce this case to the problem how to get an additional valid signature triple out of t given signature triples. We assume that t pairs of signature triples are known, these are $(m_1, s_1, r_1), (\tilde{m_1}, \tilde{r_1}, \tilde{s_1}), \cdots, (m_t, s_t, r_t), (\tilde{m_t}, \tilde{r_t}, \tilde{s_t})$. She can try to choose the parameter $k \equiv a\tilde{k} + b \pmod{q}$ such that $k \equiv k_1 \pmod{p}$ with $k_1 \equiv a_1\tilde{k}_1 + b_1 \pmod{q}$. Then using the equations $A = x_N B + kC$ and $A = x_N B + k_1 C$ Alice can compute the secret key x_N because the parameters m_1, s_1, r_1, m, s, r which appear in A, B, C are known and we have two equations and two unknown variables k and x_N. Note that k is still unknown for Alice (Otherwise she can compute $\tilde{k}$ and x_N using the blinded signature equation $\tilde{A} = \tilde{B}x_N + \tilde{C}\tilde{k}$.). Now she can solve the equations to get x_N and k.

If k and k_1 are equal then r and r_1 are equal either. Thus the problem is how to choose a, b, a_1, b_1 such that

$$\tilde{r_1}^{a_1}\alpha^{b_1} \equiv \tilde{r}^a \alpha^b \pmod{p}.$$

or

$$1 \equiv \tilde{r_1}^{-a_1}\tilde{r}^a\alpha^{b-b_1} \pmod{p}.$$

This is the representation problem [Bran93] and is equivalent to the discrete logarithm problem. Thus this attack is not successful.

But how can Alice cheat if she doesn't follow the protocol ? Note that she can compute the parameters $r, \tilde{m}$ and s totally different. But then she doesn't get any valid signature on any message m and it will be hard for her to combine several non-signatures into one signature.

Instead of using $d(r,m) := r$ we can also use the general suitable function $d(r',m)$ where $r' := \tilde{r}^a\alpha^b \pmod p$ if m does not appear as argument in the functions e, f, g. The resulting Meta-scheme is given in the following table:

Meta-blind scheme for d-variants		
parameter: p, q prime, α generator, m message		
Owner *Alice*	Channel	Notary *Nancy*
$a, b \in_R \mathbf{Z}_q^*$		$\tilde{k} \in_R \mathbf{Z}_p{}^*$
$\tilde{r}'$	$\longleftarrow$	$\tilde{r}' := \alpha^{\tilde{k}} \pmod p$
$r' := \tilde{r}'^a\alpha^b \pmod p$		
$r := d(r', m)$		
$\tilde{r} := \psi(a, b, r)$	$\longrightarrow$	$\tilde{r}$
$\tilde{s}$	$\longleftarrow$	$\tilde{s} := \rho(x_N, \tilde{k}, \tilde{r})$
$s := \theta(a, b, r, \tilde{r}, \tilde{s})$		

Further, instead of using the equation (A1) $r' := (\tilde{r}')^a\alpha^b \equiv \alpha^{\tilde{k}a+b} \pmod p$ we can also use (A2) $r' := \tilde{r}'y_N^{-a}\alpha^b \equiv \alpha^{\tilde{k}-x_Na+b} \pmod p$ as suggested by Okamoto in [Okam92], (A3) $r' := (\tilde{r}')^a y_N^b \equiv \alpha^{\tilde{k}a+x_Nb} \pmod p$ or (A4) $r' := (\tilde{r}')^a y_N\alpha^b \equiv \alpha^{\tilde{k}a+x_N+b} \pmod p$ which haven't been proposed before. This leads to slightly modified general equations (9), (10) from which we obtain many additional efficient variants. We get the following equations:

$$(A2) \qquad A = bC + \tilde{A}C\tilde{C}^{-1} \pmod q \tag{17}$$
$$B = aC - \tilde{B}\tilde{C}^{-1} \pmod q \tag{18}$$

$$(A3) \qquad A = a\tilde{A}C\tilde{C}^{-1} \pmod q \tag{19}$$
$$B = a\tilde{B}C\tilde{C}^{-1} + bC \pmod q \tag{20}$$

$$(A4) \qquad A = a\tilde{A}C\tilde{C}^{-1} + bC \pmod q \tag{21}$$
$$B = a\tilde{B}C\tilde{C}^{-1} - 1 \pmod q \tag{22}$$

The approach (A3) is interesting for the $d-$variants and the message recovery variants as we can choose here $A = s, B = r, C = 1$ and $d(r', m) = m + r$. Then we get the following equations for $\tilde{r}$ and s:

$$r := \tilde{r}a + b, s = \tilde{s}a \text{ and } s = x_N r + k.$$

Security considerations

The attack described above for the total break also applies here, if the notary doesn't examine the coefficients $\tilde{B}$ and $\tilde{C}$ carefully.

5. Meta-Message recovery blind signatures

Now we present the Meta-Message recovery blind signature scheme based on the message recovery blind signatures [CaPS94]. The idea is that notary Nancy chooses the blinded parameter $\tilde{t} := \alpha^{\tilde{k}}$ herself (with a random $\tilde{k} \in \mathbf{Z}_q$) and the owner Alice chooses the unblinded $r := m\tilde{t}^a\alpha^b$ (with random $a, b \in \mathbf{Z}_q$). Nancy signs the blinded parameter $\tilde{r}$ using the equation

$$\tilde{A} \equiv x_N\tilde{B} + \tilde{k}\tilde{C} \pmod q, \tag{23}$$

which is equivalent to $\tilde{k} \equiv \tilde{C}^{-1}(\tilde{A} - x_N\tilde{B}) \pmod q$, where x_N is the secret key of Nancy. The signature on the unblinded message m is given by (r, s). Its validity is checked by the message recovery equation

$$m := \alpha^{AC^{-1}} y_N^{-BC^{-1}} r \pmod p. \tag{24}$$

$$\alpha^{AC^{-1}} y_N^{-BC^{-1}} r \equiv \alpha^{AC^{-1} - x_N BC^{-1} + a\tilde{k} + b} m \equiv$$

$$\alpha^{AC^{-1} - x_N BC^{-1} + a(\tilde{C}^{-1}(\tilde{A} - x_N \tilde{B})) + b} m \equiv \alpha^{(C^{-1}A + a\tilde{C}^{-1}\tilde{A} + b) - x_N (C^{-1}B + a\tilde{C}^{-1}\tilde{B})} m \pmod{p}$$

and this should be equivalent to m modulo p. Hence we get the equations

$$A = -a\tilde{A}C\tilde{C}^{-1} - bC \pmod{q} \tag{25}$$

$$B = -a\tilde{B}\tilde{C}^{-1}C \pmod{q} \tag{26}$$

Note that we get nearly the same equations as in the Meta-ElGamal blind signature scheme. If the value s does not appear in C then it is possible to transform these two equations to get $\tilde{r} := \psi(a, b, r, \tilde{t})$ and $s := \theta(a, b, r, \tilde{r}, \tilde{s}, \tilde{t})$. Note that s and $\tilde{s}$ are not allowed in the equation for $\tilde{r}$. Furthermore we can transform the signature equation (23) to get $\tilde{s} := \rho(x_N, \tilde{k}, \tilde{r})$. From this the Meta-Message recovery scheme follows:

Meta-Message recovery blind signature scheme		
Parameter: p, q prime, α generator, m message		
Owner *Alice*	Channel	Notary *Nancy*
$a, b \in_R \mathbf{Z}_q$		$\tilde{k} \in_R \mathbf{Z}_p{}^*$
$\tilde{t}$	$\longleftarrow$	$\tilde{t} := \alpha^{\tilde{k}} \pmod{p}$
$r := m\tilde{t}^a \alpha^b \pmod{p}$		
$\tilde{r} := \psi(a, b, r, \tilde{t})$	$\longrightarrow$	$\tilde{r}$
$\tilde{s}$	$\longleftarrow$	$\tilde{s} := \rho(x_N, \tilde{k}, \tilde{r})$
$s := \theta(a, b, r, \tilde{r}, \tilde{s}, \tilde{t})$		

The signature of the message m is (r, s). The message recovery is done by calculating

$$m := \alpha^{AC^{-1}} y_N^{-BC^{-1}} r \pmod{p}$$

and checking, if m satisfies the redundancy scheme. We summarize the equations for some efficient types in table 5:

No.	signature	equation(25)	equation(26)
MB I.2	$1 \equiv -x_N\tilde{s} + \tilde{k}\tilde{r}$	$1 \equiv -a\tilde{r}^{-1}r - br$	$s \equiv -a\tilde{r}^{-1}\tilde{s}r$
MB I.3	$\tilde{s} \equiv -x_N\tilde{r} + \tilde{k}$	$s \equiv -a\tilde{s} - b$	$r \equiv -a\tilde{r}$
MB I.4	$\tilde{s} \equiv -x_N + \tilde{k}\tilde{r}$	$s \equiv -a\tilde{r}^{-1}\tilde{s}r - br$	$1 \equiv -a\tilde{r}^{-1}r$
MB I.5	$\tilde{r} \equiv -x_N\tilde{s} + \tilde{k}$	$r \equiv -a\tilde{r} - b$	$s \equiv -a\tilde{s}$
MB II.1	$\tilde{s} \equiv -x_N f(\tilde{r}, \tilde{s}) + \tilde{k}$	$s \equiv -af(\tilde{r}, \tilde{s}) - b$	$f(r, s) \equiv -af(\tilde{r}, \tilde{s})$
MB II.6	$f(\tilde{r}, \tilde{s}) \equiv -x_N\tilde{s} + \tilde{k}$	$f(r, s) \equiv -a\tilde{s} - b$	$s \equiv -a\tilde{s}$
MB III.2	$\tilde{r} \equiv -x_N f(\tilde{r}, \tilde{s}) + \tilde{k}$	$r \equiv -a\tilde{r} - b$	$f(r, s) \equiv -af(\tilde{r}, \tilde{s})$
MB III.3	$f(\tilde{r}, \tilde{s}) \equiv -x_N + \tilde{k}\tilde{r}$	$f(r, s) \equiv -a\tilde{r}^{-1}f(\tilde{r}, \tilde{s})r - br$	$1 \equiv -a\tilde{r}^{-1}r$
MB III.4	$f(\tilde{r}, \tilde{s}) \equiv -x_N\tilde{r} + \tilde{k}$	$f(r, s) \equiv -af(\tilde{r}, \tilde{s}) - b$	$r \equiv -a\tilde{r}$
MB III.5	$1 \equiv -x_N f(\tilde{r}, \tilde{s}) + \tilde{k}\tilde{r}$	$1 \equiv -a\tilde{r}^{-1}r - br$	$f(r, s) \equiv -a\tilde{r}^{-1}f(\tilde{r}, \tilde{s})r$

Table 5: Efficient variants of the Message recovery blind signature scheme

Note that in the schemes listed in table 5 we can get the functions ψ and θ by transforming the equations (25) and (26) with a general function f. In any other variant of *Type* MB I – IV this is not possible for general functions f and g.

The proof of blindness is similar to the proof given for the Meta-ElGamal blind signature scheme above.

$\tilde{A} \equiv \tilde{B}x_N + \tilde{C}\tilde{k} \pmod q$, $\alpha^{AC^{-1}} y_N^{-BC^{-1}} r \equiv m \pmod p$ and A, B, C chosen from the table above, there exist unique $a, b \in \mathbf{Z}_q$ with

$$r \equiv m\tilde{t}^a \alpha^b \pmod p \tag{27}$$
$$A = -a\tilde{C}^{-1}\tilde{A}C - bC \pmod q \tag{28}$$
$$B = -a\tilde{C}^{-1}\tilde{B}C \pmod q \tag{29}$$
$$\tag{30}$$

Proof: Choose $a, b \in \mathbf{Z}_q$ with

$$a = -B\tilde{C}\tilde{B}^{-1}C^{-1} \pmod q \tag{31}$$
$$b = (-A + \tilde{A}B\tilde{B}^{-1})C^{-1} \pmod q \tag{32}$$

Using the signature equation (23) from above we get

$$\begin{aligned} a\tilde{k} + b &\equiv (-B\tilde{C}\tilde{B}^{-1}C^{-1})\tilde{k} + (-A + \tilde{A}B\tilde{B}^{-1})C^{-1} \\ &\equiv C^{-1}(-A - B(\tilde{C}\tilde{B}^{-1}\tilde{k} - \tilde{A}\tilde{B}^{-1})) \\ &\equiv C^{-1}(-A - B((\tilde{A} - x_N\tilde{B})\tilde{B}^{-1} - \tilde{A}\tilde{B}^{-1})) \\ &\equiv C^{-1}(-A + Bx_N) \pmod q \end{aligned} \tag{33}$$

Thus we have

$$m\tilde{t}^a\alpha^b \equiv m\alpha^{a\tilde{k}+b} \equiv m\alpha^{C^{-1}(-A+Bx_N)} \equiv m\alpha^{-AC^{-1}} y_N^{BC^{-1}} \equiv r \pmod p$$

The validity of the relations for a and b is trivial. Moreover the choice of a and b is unique, because congruence (33) must be satisfied.

Hence we can define Meta-Message recovery blind signature scheme

$$MRB = (Mode.Type.No, d, e, f, g)$$

similar to the Meta-Message recovery scheme.

5.1 Security considerations

5.2 Efficiency considerations

For a detailed security and efficiency analysis we refer to the final version of this paper.

6. Efficient variants

Efficient Meta-ElGamal blind signatures

Recently Harn published a digital signature scheme [Harn94], which is variant EG II.3 of the Meta-ElGamal scheme. From this we get the following blind signature scheme:

The signature equation for notary Nancy is

$$\tilde{s} \equiv x_N(\tilde{m} + \tilde{r}) - \tilde{k} \pmod q.$$

Thus we have $A := s, B := (m + r), C := -1$ and we can substitute the equations (10) and (9) to get $m + r = a(\tilde{m} + \tilde{r})$ and $s = a\tilde{s} - b$. We have $\tilde{m} := \psi(a, b, r, \tilde{r}, m) = a^{-1}(m + r) - \tilde{r}$, $s := \theta(a, b, \tilde{s}, \tilde{r}, r, \tilde{m}, m) = a\tilde{s} - b$ and $\tilde{s} := \rho(\tilde{m}, \tilde{r}, x_N, \tilde{k}) = x_N(\tilde{m} + \tilde{r}) - \tilde{k}$.

We get the following scheme:

parameter: p, q prime, α generator, m message

owner Alice		notary Nancy
$a, b \in_R \mathbf{Z}_q$		$\tilde{k} \in_R \mathbf{Z}_p{}^*$ $\tilde{r} := \alpha^{\tilde{k}} \pmod p$
$\tilde{r}$ $r := \tilde{r}^a \alpha^b \pmod p$	$\longleftarrow$	$\tilde{r}$
$\tilde{m} := a^{-1}(m + r) - \tilde{r}$	$\longrightarrow$	$\tilde{m}$ $\tilde{s} \equiv x_N(\tilde{m} + \tilde{r}) + \tilde{k} \pmod q.$
$\tilde{s}$ $s := a\tilde{s} + b$	$\longleftarrow$	$\tilde{s}$

Performance analysis: Owner Alice needs to compute two on–line exponentiations modulo p with a $|q|$ bit exponent and one off–line inverse modulo q. Notary Nancy needs just one off-line exponentiation modulo p with a $|q|$ bit exponent. A verifier needs to compute two exponentiations instead of three as usual for an ElGamal signature. Note that this scheme is more efficient than the variant proposed in [CaPS94]. In that variant owner Alice additionally computes two on-line inversions modulo q. There are other variants with high efficiency (EB II.4, EB III.3, EB III.4, EB IV.3, EB IV.4, EB V.3, EB V.4) where the addition is used for the functions f and g [HMP394].

Efficient Meta-Message recovery blind signatures
The most efficient variants for the message recovery blind signature scheme are the variants MB I.3, MB I.5, MB II.1, MB II.6, MB III.2 and MB III.4. Here the owner needs one on-line and one off-line exponentiation modulo p with a $|q|$ bit exponent, the notary one off-line exponentiation modulo p with a $|q|$ bit exponent and the verifier two exponentiations modulo p with a $|p|$ bit exponent.

7. Applications of the Meta schemes

7.1 Authentic message encryption

The basic encryption scheme which can be developed from the ElGamal encryption scheme [ElGa84] has been proposed in [NyRu94]. In this scheme we have to submit three ciphertext blocks. An improvement of this scheme was proposed in [HMP194] in which we need to transmit only two message blocks of length $|p| + |q|$ instead of $2|p| + |q|$ in the basic encryption scheme. This scheme can also be generalized for the Meta-Message recovery scheme.

The trusted third party chooses two large primes $p, q \in \mathbf{P}$ with $q|(p-1)$, an element α of order q and a one-way function $h : \mathbf{Z}_p{}^* \to \mathbf{Z}_p{}^*$. These public parameters are authentic to all users. Each user $i \in \{A, B\}$ chooses a secret key $x_i \in \mathbf{Z}_q{}^*$ and computes his public key $y_i := \alpha^{x_i} \pmod p$. He publishes y_i which is certified by a trusted third party and keeps x_i secret. To send a message $m \in \mathbf{Z}_{p-1}$ within a suitable redundancy scheme, the user *Alice* chooses a random $k \in \mathbf{Z}_q$ and computes $r := h(y_B^k)^{-1} m \pmod p$, $\tilde{r} := r \pmod q$ and $s := k - x_A \tilde{r} \pmod q$. Then she sends (r, s) to the receiver *Bob*, who computes $\tilde{r} := r \pmod q$, recovers the message $m := h(y_B^s y_A^{\tilde{r} x_B}) r \pmod p$ and checks if m satisfies the redundancy scheme. We prove that the scheme is correct:

$$h(y_B^s y_A^{x_B \tilde{r}}) \equiv h(y_B^{k - x_A \tilde{r}} \alpha^{x_A x_B \tilde{r}}) r \equiv h(y_B^k) \left(h(y_B^k)\right)^{-1} m \equiv m \pmod p$$

Alice has to compute one inverse modulo p, one exponentiation modulo p with a $|q|$ bit exponent and executes g one time. Bob computes two exponentiations modulo p with a $|q|$ bit exponent and executes g one time. The ciphertext for a $|p|$ bit plaintext within the redundancy scheme has a length of $|p| + |q|$ bit. Thus the expansionrate is only $(|p| + |q|)/|p| = 1 + |q|/|p|$.

7.1.2 Security considerations

An attacker, Carol, can try to forge the signature or break the encryption scheme. Forging the signature scheme seems to be as hard as forging the Nyberg-Rueppel scheme and breaking the encryption scheme seems to be as hard as the Diffie-Hellman problem [DiHe76] (that is: given α^{x_A} and α^{x_B}, compute $\alpha^{x_A x_B}$). If the Diffie-Hellman problem can be solved then the scheme is no longer secure.

Note that the one-way property of the function g is necessary because otherwise an attacker Carol can get the Diffie-Hellman-Key $K := y_A^{x_B} \equiv y_B^{x_A} \pmod{p}$ with just one known plaintext–ciphertext pair (m, r, s) by computing

$$K \equiv (g^{-1}(r^{-1}m)y_B^{-s})^{(r')^{-1}} \pmod{p}.$$

Then Carol can read every further message with given ciphertext (r, s) by computing $r' := r \pmod{q}$ and recovering the message $m := h(y_B^s K^{r'})r \pmod{p}$.

Meta authentic encryption schemes

We can adopt the ideas of this kind of schemes to all Meta-Message recovery schemes. The equations $r' := h(y_B^k)$ and $r := d(r', m) \pmod{p}$ are the same in every variant. Furthermore the signature generation is like in the general equation (5). The message recovery can be done by the equation

$$m \equiv d^{-1}\left(r, h(y_B^{AC^{-1}} y_A^{-x_B BC^{-1}})\right) \pmod{p}. \tag{34}$$

The correctness of this equation is shown by the following congruence:

$$\begin{aligned} d^{-1}\left(r, h(y_B^{AC^{-1}} y_A^{-x_B BC^{-1}})\right) &\equiv d^{-1}\left(r, h(y_B^{(A-x_A B)C^{-1}})\right) \\ &\equiv d^{-1}\left(r, h(y_B^{kCC^{-1}})\right) \equiv d^{-1}(r, r') \equiv m \pmod{p}. \end{aligned}$$

Note that the variants in [HMP194] which are based on the $rs-$variants are not authentic as pointed out by Lim [Lim 94]. His attack can be countermeasured if the receiver checks whether the session key $h(y_B^k)$ has been used before. As this solution is not very practical we recommend not to use them.

7.2 Self-certified public keys

By applying the ideas described in [BaKn89, Guen89] we can create identity based certificates for the public keys of identified users with distinguished names, which results in the self-certified public keys [Gira91]. They have the following properties:

- The public key can be computed as a function of the identity, the public parameters (generator α, prime modul p, public key of the trusted authority y_Z) and the signature parameter r, which isn't necessarily authentic. The secret key is computed by the trusted authority as the signature parameter s.
- The authenticity of the public key is not directly verified, but only the authorized user knows the corresponding secret key and thus can benefit of this public key.

The trusted authority Z signs the identity ID_A of user Alice with the signature scheme giving message recovery proposed in [NyR193]. Z chooses a random number $k_A \in \mathbf{Z}_q^*$, computes $r_A := \alpha^{-k_A} ID_A \pmod p$ and solves the signature equation for the parameter x_A:

$$x_A := k_A - x_Z r_A \pmod q.$$

The tuple (r_A, x_A) is a signature on the message ID_A. The parameter r_A is published and the parameter x_A is kept as secret key. The corresponding public key can be computed as

$$y_A := \alpha^{x_A} \equiv \alpha^{k_A - x_Z r_A} \equiv \alpha^{k_A} y_Z^{-r_A} \equiv y_Z^{-r_A} r_A^{-1} ID_A \pmod p.$$

Thus it can be computed as a function u_A with arguments α, y_Z, r_A and ID_A, which are all public known.

In the following, we will describe the general approach to obtain the self-certified public key y_A of user *Alice*. For the sake of clearness we restrict the description to the *Mode* L.

Using the Meta-Message recovery scheme

The trusted authority Z, who uses his public key y_Z and his secret key x_Z signs the identity ID_A of Alice with the Meta-Message recovery scheme by choosing a random $k_A \in \mathbf{Z}_q$, computing $r'_A := \alpha^{k_A} \pmod p$, $r_A := d(r'_A, ID_A)$ and solving the general signature equation for the parameter x_A:

$$A \equiv x_Z B + k_A C \pmod q \tag{35}$$

with $\pm A, \pm B, \pm C$ permutations of general functions $e, f, g : {\mathbf{Z}_q}^2 \to \mathbf{Z}_q$ and arguments r_A and x_A. This gives an equation of the form $x_A =: a_A(x_Z, k_A, r_A) b_A(x_Z, k_A, r_A)^{-1} \pmod q$, where the functions a_A and b_A are implicit defined by the chosen variant. The tuple (r_A, x_A) is a signature on the message ID_A. The public key y_A can be computed as $y_A := \beta_A^{x_A} \pmod p$ with generator $\beta_A := \alpha^{b_A(x_Z, k_A, r_A)} =: t_A(\alpha, y_Z, r_A, ID_A) \pmod p$, such that the following congruence holds for y_A:

$$y_A \equiv \beta_A^{x_A} \equiv \alpha^{b_A(x_Z,k_A,r_A) a_A(x_Z,k_A,r_A) b_A(x_Z,k_A,r_A)^{-1}} \equiv \alpha^{a_A(x_Z,k_A,r_A)} =: u_A(\alpha, y_Z, r_A, ID_A) \pmod p.$$

(Note, that $\alpha^{x_Z} \equiv y_Z \pmod p$ and $\alpha^{k_A} \equiv d^{-1}(r_A, ID_A) \pmod p$!) This means, that y_A can be computed as a function u_A of the authentic public parameters α, y_Z, ID_A and the parameter r_A, which doesn't need to be authentic. To guarantee, that the last equivalence holds, the functions e, f and g have to be chosen as a composition of arithmetic operations.

If we consider for example *Type* MR I with function d as $d(r', m) = r' \cdot m^{-1}$ and suitable signs we get the following public keys:

No.	signature	$\beta_A = t_A(\alpha, y_Z, r_A, ID_A)$	$y_A = u_A(\alpha, y_Z, r_A, ID_A)$
MR I.1	$1 \equiv -x_Z r_A + k_A x_A$	$r_A ID_A$	$\alpha y_Z^{r_A}$
MR I.2	$1 \equiv x_Z x_A - k_A r_A$	y_Z	$\alpha (r_A ID_A)^{r_A}$
MR I.3	$x_A \equiv x_Z r_A + k_A$	α	$y_Z^{r_A} r_A ID_A$
MR I.4	$x_A \equiv x_Z + k_A r_A$	α	$y_Z (r_A ID_A)^{r_A}$
MR I.5	$r_A \equiv x_Z x_A - k_A$	y_Z	$\alpha^{r_A} r_A ID_A$
MR I.6	$r_A \equiv -x_Z + k_A x_A$	$r_A ID_A$	$\alpha^{r_A} y_Z$

The main drawback of this general approach is, that the trusted authority creates the secret keys of all users and therefore requires unconditional trust by the users.

Using Meta blind signature schemes

This drawback can be prevented by signing the secret key x_A with a hidden signature [HoKn91, HoP394, HMP494], such that it can be modified by the user after signature generation without the trusted authority knowing it [HoP294]. It can also be signed with one of the two Meta blind signature schemes described above, to obtain a self-certified public key for one pseudonym of a user.

message $m := h(ID_A)y_A$ with a public hash function h to obtain a certificate (r, s) and recovering the public key as $y_A := mh(ID_A)^{-1} \pmod p$ by using the identity ID_A of the user [NyR193]. This method can be applied to all Meta-Message recovery schemes.

Performance in computing self-certified public keys

Note that the computation of the self-certified public-keys is more efficient in some variants than in the previously known protocols. For example in variants MR I.3 and MR I.4 we just need one exponentiation modulo p where in the proposed schemes in [BaKn89, Guen89] we need two exponentiations and one inversion.

7.2.1 Authentication schemes

In the authentication schemes, the user *Alice* who wants to authenticate herself to the verifier *Bob* proves the knowledge of the discrete logarithm x_A of her self-certified public key $y_A := u_A(\alpha, y_Z, r_A, ID_A)$ to the basis $\beta_A = t_A(\alpha, y_Z, r_A, ID_A)$ by the Zero-Knowledge-proof of Chaum, Evertse and van de Graaf [ChEG87]. This can be done with all self-certified public keys obtained from the Meta-Message recovery scheme [HoP294]. It can also be applied to other authentication protocols, e.g. [ChEG87, BrMc90, Okam92].

Meta-Zero-knowledge authentication scheme		
User *Alice*	Channel	Verifier *Bob*
$v_A \in \mathrm{RAND}(\mathbf{Z}_q)$		
$w_A := \beta_A^{v_A} \pmod p$	$\longrightarrow$	(ID_A, r_A, w_A)
		$\beta_A := t_A(\alpha, y_Z, r_A, ID_A)$
		$y_A := u_A(\alpha, y_Z, r_A, ID_A)$
c_A	$\longleftarrow$	$c_A \in \mathrm{RAND}[0 : 2^t - 1]$
$z_A := v_A + x_A c_A \pmod q$	$\longrightarrow$	v
		Accepted, if
		$w_A \equiv \beta_A^{v_A} y_A^{-c_A} \pmod p$

7.2.2 Authenticated key exchange

To obtain a general authenticated key exchange protocol, we can adapt the ideas from [BaKn89, Guen89] or the SELANE-protocols [HoKn91] which both use self-certified public keys.

The *first approach* combines mutual authentication and key exchange by using the (authentic) data exchanged during the authentication protocol to construct the session key. The session key can be computed from the authentic parameters y_A, y_B in the Meta-authentication scheme above and some exchanged, randomly chosen parameters $e_A := \beta_A^{d_A} \pmod p, e_B := \beta_B^{d_B} \pmod p$ in the following manner:

$$K := e_B^{v_A} w_B^{d_A} \equiv \beta_B^{v_A d_B + v_B d_A} \equiv \beta_A^{v_A d_B + v_B d_A} \equiv w_A^{d_B} e_A^{v_B} \pmod p.$$

Obviously this is only correct if $\beta_A = \beta_B$ and thus some variants can't be used.

In the *second approach*, a Diffie-Hellman key is computed, which can only be known by the two parties. The key has to be verified after computation, to ensure that both sides know the same key. *Alice* computes $y_B := u_B(\alpha, y_Z, r_B, ID_B)$ and $\beta_B := t_B(\alpha, y_Z, r_B, ID_B)$. Then she chooses a random number $x_A \in \mathbf{Z}_q{}^*$ and computes $y_A := \beta_B^{x_A} \pmod p$. y_A is transmitted to *Bob* who computes $y_A := u_A(\alpha, y_Z, r_A, ID_A)$ and $\beta_A := t_A(\alpha, y_Z, r_A, ID_A)$. He chooses a random number $x_B \in \mathbf{Z}_q{}^*$ and computes $y_B := \beta_A^{x_B} \pmod p$, which he transmits to *Alice*. *Alice* computes her session key $K_A := y_B^{x_A} y_B^{x_A} \pmod p$ and *Bob* his session key $K_B := y_A^{x_B} y_A^{x_B} \pmod p$. If the transmission of the values y_A and y_B hasn't been disturbed by any attacker, the two session keys are identically as proven by the following congruence:

$$K_A := w_B^{x_A} y_B^{v_A} \equiv (\beta_A^{v_B})^{x_A} (\beta_B^{x_B})^{v_A} \equiv (\beta_B^{v_A})^{x_B} (\beta_A^{x_A})^{v_B} \equiv w_A^{x_B} y_A^{v_B} := K_B \pmod p.$$

recovery scheme to compute their self-certified public keys y_i and the corresponding generators β_i ($i \in \{A, B\}$).

Both solutions are vulnerable to the triangle attack [Burm94] where under special circumstances the session key of two users can be calculated using eavesdropped information and additional information which is voluntarily given by the users to the attacker. This attack can be countermeasured, but the amortized security can't be proven [YaSh89].

A *third possibility* to exchange an authentic session key, is to sign as message m a random value $w_A := \alpha^{v_A} \pmod p$ by *Alice* with one of the Meta-Message recovery schemes and also a value $w_B := \alpha^{v_B} \pmod p$ by *Bob*. Both parties can exchange their signatures, recover the values q_A and w_B and compute the session key as in the Diffie-Hellman key exchange [DiHe76, NyRu94]:

$$K_A := w_B^{v_A} \equiv \alpha^{v_A v_B} \equiv w_A^{v_B} := K_B \pmod p.$$

7.3 Further applications

From the Meta-Message recovery scheme we can develop hidden and weak blind signature schemes giving message recovery as presented in [HoP394, HMP494]. Other applications of the Meta-Message recovery scheme are multisignatures, blind multisignatures [HMP694] and threshold cryptosystems as proposed in [HMP594, HMP294]. Details on this topic are very extensive and can be found in the literature.

8. Conclusion

We have presented Meta-Message recovery signature schemes for one, two and three message blocks based on the Meta-ElGamal signature scheme. All previously known message recovery schemes based on the discrete logarithm problem can be integrated into this approach. For further discussions we have to consider only this Meta-scheme. We have shown how to generalize the ElGamal- and the Message recovery blind signature schemes and discussed the most efficient variants. Furthermore some interesting applications like Meta-authentic encryption schemes, identity based public keys used for authentication and authenticated key exchange schemes have been mentioned.

References

[BaKn89] F.Bauspieß, H.-J.Knobloch, "How to keep authenticity alive in a computer network", Lecture Notes in Computer Science 434, Advances in Cryptology: Proc. Eurocrypt '89, Berlin: Springer Verlag, (1990), pp. 38–46.

[Bran93] S. Brands, "An efficient off-line electronic cash system based on the representation problem", Lecture Notes in Computer Science , Advances in Cryptology: Proc. Eurocrypt '93, (1993), S. 26.1–26.15

[BrMc90] E.F.Brickell, K.S.McCurley, "An Interactive Identification scheme based on discrete logarithms and factoring", Lecture Notes in Computer Science 473, Advances in Cryptology: Proc. Eurocrypt '90, Berlin: Springer Verlag, (1991), pp. 63–71.

[Burm94] M.Burmester, "On the risk of opening distributed keys", Lecture Notes in Computer Science 839, Advances in Cryptology: Proc. Crypto '94, Berlin: Springer Verlag, (1994), pp. 308 – 317.

[CaPS94] J.L.Camenisch, J.-M.Piveteau, M.A.Stadler, "Blind signature schemes based on the discrete logarithm problem", Preprint, presented at the Rump session of Eurocrypt '94, (1994), 5 pages.

[Chau82] D. Chaum, "Blind signatures for untraceable payments", Advances in Cryptology: Proc. Crypto '82, New York: Plenum Press, (1983), pp. 199–203.

obsolete", Communications of the ACM, Vol. 28, (1985), pp. 1030–1044.

[ChEG87] D.Chaum, J.H.Evertse, J.van de Graaf, "Demonstrating possession of a discrete logarithms and some generalizations", Lecture Notes in Computer Science 304, Advances in Cryptology: Proc. Eurocrypt '87, Berlin: Springer Verlag, (1988), pp. 127–141.

[DiHe76] W.Diffie, M.Hellman,"New directions in cryptography", IEEE Transactions on Information Theory, Vol. IT-22, No. 6, November, (1976), pp. 644–654.

[ElGa84] T.ElGamal, "Cryptography and logarithms over finite fields", Stanford University, CA., UMI Order No. DA 8420519, (1984), 119 pages.

[ElGa85] T.ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms", IEEE Transactions on Information Theory, Vol. IT-30, No. 4, July, (1985), pp. 469–472.

[FuOO92] A.Fujioka, T.Okamoto, K.Ohta, "A practical secret voting scheme for large scale elections", Auscrypt '92, Abstracts, Gold Coast, Australia, 13.–16. December, (1992), S. 6-15–6-19.

[Gira91] M.Girault, "Self-Certified Public Keys", Lecture Notes in Computer Science 547, Advances in Cryptology: Proc. Eurocrypt '91, Berlin: Springer Verlag, (1991), pp. 490–497.

[Guen89] C.G.Günter, "An identity based key exchange protocol", Lecture Notes in Computer Science 434, Advances in Cryptology: Proc. Eurocrypt '89, Berlin: Springer Verlag, (1990), pp. 29–37.

[Harn94] L.Harn, "New digital signature scheme based on discrete logarithm", Electronics Letters, Vol. 30, No. 5, (1994), pp. 396 – 398.

[HoKn91] P.Horster, H.-J.Knobloch, "Discrete Logarithm based protocols", Lecture Notes in Computer Science 547, Advances in Cryptology: Proc. Eurocrypt '91, Berlin: Springer Verlag, (1992), pp. 399–408.

[HMP194] P.Horster, M.Michels, H.Petersen, "Authenticated encryption schemes with low communication costs", Electronics Letters, Vol. 30, No. 15, July, (1994), pp. 1230–1231.

[HMP294] P.Horster, M.Michels, H.Petersen, "Generalized ElGamal signature schemes for one message block", Proc. 2nd Int. Workshop on IT-Security, Vienna, Sep. 22.–23., (1994), 16 pages.

[HMP394] P.Horster, M.Michels, H.Petersen, "Meta-ElGamal signature schemes", Proc. 2nd ACM conference on Computer and Communications security, Fairfax, Virginia, Nov. 2–4, (1994), pp. 96–107.

[HMP494] P.Horster, M.Michels, H.Petersen, "Hidden and weak blind signature schemes and related concepts", Technical Report TR-94-10, University of Technology Chemnitz-Zwickau , August, (1994), 8 pages[1].

[HMP594] P.Horster, M.Michels, H.Petersen, "Meta-Multisignature schemes based on the discrete logarithm problem", Technical Report TR-94-12, University of Technology Chemnitz-Zwickau, September, (1994), 11 pages[1].

[HMP694] P.Horster, M.Michels, H.Petersen, "Blind multisignature schemes based on the discrete logarithm problem", Presented at Rump Session of this conference, Nov. 29, (1994), 4 pages.

Informationssystemen, Proceedings of SIS '94, Verlag der Fachvereine Zürich, (1994), pp. 89–106.

[HoP294] P.Horster, H.Petersen, "Signature and authentication schemes based on the discrete logarithm" (in German), Internal Report 94–9, RWTH Aachen, ISSN 0935–3232, March, (1994), 96 pages[1].

[HoP394] P.Horster, H.Petersen, "Classification of blind signature schemes and examples of hidden and weak blind signatures", Presented at the Rump Session of Eurocrypt '94, Perugia, Italy, (1994), 6 pages[1].

[Knob94] H.-J.Knobloch, "A remark on the size of ElGamal-type digital signatures", EISS Report 94/1, European Institute for System Security (EISS), University of Karlsruhe, (1994), 5 pages.

[Lim 94] C.H.Lim, E-mail to the authors, August 20, (1994).

[NIST91] National Institute of Standards and Technology, Federal Information Process. Standard, FIPS Pub XX: Digital Signature Standard (DSS), (1991).

[NyR193] K.Nyberg, R.Rueppel, "A new signature scheme based on the DSA giving message recovery", Proc. 1st ACM Conference on Computer and Communications Security, Fairfax, Virginia, Nov. 3–5., (1993), 4 pages.

[NyR293] K.Nyberg, R.Rueppel, "New signature schemes based on the discrete logarithm problem or how to add message recovery to the DSA", Preprint, November 13, (1993), 13 pages.

[NyRu94] K.Nyberg, R.Rueppel, "Message recovery for signature schemes based on the discrete logarithm problem", Pre-proceedings of Eurocrypt '94, University of Perugia, Italy, (1994), pp. 175 – 190.

[Okam92] E.Okamoto, "Provable secure and practical identification schemes and corresponding signature schemes", Lecture Notes in Computer Science 740, Advances in Cryptology: Proc. Crypto '92, Berlin: Springer Verlag, (1993), pp. 31–53.

[Pinc94] G.E.Pinch, "Comment: New signature scheme with message recovery", Electronics Letters, Vol. 30, No. 11, (1994), S. 852.

[Pive93] J.M.Piveteau, "New signature scheme with message recovery", Electronics Letters, Vol. 29, No. 25, (1993), pp. 2185.

[RiSA78] R.L. Rivest, A. Shamir, L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", Comm. of the ACM, Vol. 21, (1978), S. 120–126.

[Schn89] C.P.Schnorr, "Efficient identification and signatures for smart cards", Lecture Notes in Computer Science 435, Advances in Cryptology: Proc. Crypto '89, Berlin: Springer Verlag, (1990), pp. 239–251.

[Schn91] C.P.Schnorr, "Comment on DSA: Comparison of the Digital Signature Algorithm and the Signature schemes of ElGamal and Schnorr", Letter to the Director of CSL/NIST, October 25th, (1991), 7 pages.

[YaSh89] Y.Yacobi, Z.Shmuely, "On key distribution systems", Lecture Notes in Computer Science 435, Advances in Cryptology: Proc. Crypto '89, Berlin: Springer Verlag, (1990), pp. 344–355.