Tatra Mt. Math. Publ. **25** (2002), 23–37

Tatra Mountains
Mathematical Publications

OBSTACLES IN TWO PUBLIC KEY CRYPTOSYSTEMS BASED ON GROUP FACTORIZATIONS

María Isabel González Vasco — Rainer Steinwandt

ABSTRACT. Recently S. Magliveras, D. Stinson, and Tran Van Trung proposed two approaches for deriving a public key cryptosystem from the symmetric cipher PGM. The first proposal, MST_1, uses the same kind of group factorizations as PGM (so-called logarithmic signatures), and the second one, MST_2, makes use of a different kind called meshes. We point out security problems in MST_1 and MST_2 that arise if the keys are not chosen carefully enough. In fact, at the writing of this paper we are not aware of a practical method for deriving keys with acceptable security.

1. Introduction

Most primitives in public key cryptography are based on the assumption that a certain number theoretic problem is hard. Not only owing to the polynomial time algorithms for factoring and computing discrete logarithms on quantum computers [17] it is indeed desirable to broaden the scope of mathematical tools used in designing public key cryptosystems. Several attempts to use problems from mathematical areas like combinatorics or group theory have been made; good examples are cryptosystems based on combinatorial group theory—ranging from more conceptual approaches based on the word problem [5, 18] to more practical ones based on the conjugacy problem in braid groups [1, 7].

The systems studied in this paper make use of certain factorizations of finite groups, which are essentially sequences of finite sets leading to a representation of each group element as a product consisting of one element from each set. Several attempts to derive cryptosystems using such factorizations can be found in the literature; some are conceptually similar to those based on the knapsack problem [16, 19] and have been identified as insecure [6, 14, 15].

2000 Mathematics Subject Classification: 94A60.
Keywords: public key cryptography, group factorizations, cryptanalysis.

In [8] S. Magliveras proposed a private key cryptosystem called PGM (*Permutation Group Mappings*), which makes use of a very special kind of factorization sets for finite permutation groups called *logarithmic signatures*. Although many questions about PGM still remain unanswered, several positive results concerning the security [9, 10, 11, 12] of this system have been achieved.

The two public key systems MST_1 and MST_2 proposed in [13] are attempts to use the idea behind PGM for public key cryptography. Namely, instead of using two ways of factoring the group elements which are easily computable and should therefore be kept secret, the idea behind MST_1 is to use two ways of factoring, one of which should be hard to compute, to construct a one way permutation. The cryptosystem MST_2 is inspired by the ElGamal cryptosystem [4] and uses a different kind of factorization sets. Even though both approaches are indeed interesting, we have serious doubts concerning their feasibility; the problems described below demonstrate that deriving secure keys is quite a subtle task. In fact, presently we do not know of a practical procedure for generating keys with an acceptable security level.

2. Group factorizations

In this section we introduce the basic terminology about logarithmic signatures which will be used throughout the paper—it is essentially adopted from [12, 13].

Usually, a factorization of a finite group G is defined as a sequence $[A_1, \dots, A_s]$ of subsets of G such that, given any group element g, there is exactly one way to express it as a product $g = a_1 \cdots a_s$, where each $a_i \in A_i$. In the sequel, we will mainly deal with so-called permutation groups.—Recall that a finite group G is a *permutation group* of degree $n \in \mathbb{N}$ if it is isomorphic to some subgroup of the symmetric group S_n; from Cayley's theorem we know that each finite group G can be regarded as a permutation group of degree $|G|$.

For MST_2 we will also admit certain types of factorizations which do not exactly match the aforementioned definition, but nevertheless provide a way of factoring the group elements. We start by defining so-called *logarithmic signatures*, the kind of factorizations which we will be dealing with in MST_1:

Definition 2.1. Denote by G some permutation group of degree n. Next, let $\alpha = [\alpha_1, \dots, \alpha_s]$ be a sequence where $s \in \mathbb{N}_0$ and each α_i $(1 \leq i \leq s)$ is itself a sequence $\alpha_i = [\alpha_{i0}, \dots, \alpha_{ir_i-1}]$ with $r_i \in \mathbb{N}_0$ and $\alpha_{ij} \in S_n$ $(0 \leq j < r_i)$. Then we call α a *logarithmic signature* for G if $|G| = r_1 \cdots r_s$, and each $g \in G$ is uniquely represented as a product

$$g = \alpha_{1j_1} \cdots \alpha_{sj_s}$$

with $\alpha_{ij_i} \in \alpha_i$ $(1 \leq i \leq s)$.

If $\alpha = [\alpha_1, \ldots, \alpha_s]$ is a logarithmic signature, then we refer to the sequences $\alpha_i = [\alpha_{i0}, \ldots, \alpha_{ir_i-1}]$ $(1 \leq i \leq s)$ as *blocks* of α, and we call the vector $r = (r_1, \ldots, r_s)$ *type* of α. Finally, we denote the set of all logarithmic signatures for G by $\Lambda(G)$ (or simply Λ if there is no ambiguity).

The hardness of actually computing the unique factorization of group elements with respect to some logarithmic signatures is the basis for the cryptosystem MST_1.

Given a group G and a logarithmic signature $\alpha = [\alpha_1, \ldots, \alpha_s]$ of type $r = (r_1, \ldots, r_s)$ for it, say $\alpha_i = [\alpha_{i0}, \ldots, \alpha_{ir_i-1}]$ $(1 \leq i \leq s)$, we can construct the mappings

$$\begin{aligned} \lambda\colon \mathbb{Z}_{r_1} \times \cdots \times \mathbb{Z}_{r_s} &\longrightarrow \mathbb{Z}_{|G|}\,, \\ (n_1, \ldots, n_s) &\longmapsto \sum_{i=1}^{s} \left(n_i \cdot \prod_{j=1}^{i-1} r_j \right) \qquad (\text{“mixed radix repres.”}) \end{aligned}$$

and

$$\begin{aligned} \Theta_\alpha\colon \mathbb{Z}_{r_1} \times \cdots \times \mathbb{Z}_{r_s} &\longrightarrow G\,, \\ (n_1, \ldots, n_s) &\longmapsto \alpha_{1n_1} \cdots \alpha_{sn_s}\,. \end{aligned}$$

It is straightforward to prove both mappings to be bijective, and we can construct a third bijection

$$\begin{aligned} \breve{\alpha}\colon \mathbb{Z}_{|G|} &\longrightarrow G\,, \\ n &\longmapsto (\Theta_\alpha \lambda^{-1})(n) = \Theta_\alpha\big(\lambda^{-1}(n)\big)\,, \end{aligned}$$

which essentially materializes the factorization of the group elements induced by α:

$$\breve{\alpha}\colon \mathbb{Z}_{|G|} \xrightarrow{\lambda^{-1}} \mathbb{Z}_{r_1} \times \cdots \times \mathbb{Z}_{r_s} \xrightarrow{\Theta_\alpha} G\,.$$

For the public key cryptosystem MST_1 it is of crucial importance whether $\breve{\alpha}$ can be inverted efficiently. This motivates the following definition:

Definition 2.2. Let G be a permutation group of degree n and α a logarithmic signature for G. Then we say that α is *tame* if $\breve{\alpha}^{-1}$ can be computed in polynomial time (of n), otherwise we call it *wild*. If $\breve{\alpha}^{-1}$ can be computed in time $\mathcal{O}(n^2)$, we say that α is *supertame*.

As for MST_1 the cryptographic relevance of a logarithmic signature α will essentially rely on the induced bijection $\breve{\alpha}$, we also make.

Definition 2.3. Let G be a permutation group and α, β, two logarithmic signatures for G. Then we call α and β *equivalent* if $\breve{\alpha} = \breve{\beta}$.

Now we are left with the problem of actually constructing logarithmic signatures for permutation groups. A very simple method of doing so is the following: let G be a permutation group of degree n, and consider some subgroup chain

$$G = G_0 > G_1 > \cdots > G_s = \{1\}.$$

Now take $\alpha = [\alpha_i \colon i = 1, \ldots, s]$, a sequence of finite sequences over G such that $\alpha_i = [\alpha_{ij} \colon j = 0, \ldots, r_i - 1]$ is a complete system of left coset representatives of G_{i-1} modulo G_i. Under suitable assumptions α is a tame logarithmic signature for G (cf. [12]). Such logarithmic signatures are called *exact l-transversals*, and those constructed analogously taking right coset representatives are called *exact r–transversals*. Similarly, one can build a logarithmic signature from a subgroup chain for which each block is either a complete set of left or right coset representatives of a quotient in the chain. Such logarithmic signatures are usually refered to as *mixed transversals*. If a logarithmic signature is exact l-, r-, or mixed transversal, we say it is *exact transversal*; we denote by $\mathcal{E}(G)$ the set of all exact transversal logarithmic signatures for G.

Different ways of constructing new logarithmic signatures from a given one are studied in [12], here we will only consider some constructions which will be necessary in the sequel.

Let G be a permutation group of degree n and α a logarithmic signature for G. Given permutations $t_1, \ldots, t_{s-1} \in S_n$ and $t_0, t_s \in G$ we construct the sequence $\beta = [\beta_1, \ldots, \beta_s]$ where for $1 \leq i \leq s$ we have $\beta_i = t_{i-1}^{-1} \alpha_i t_i$. Clearly, β is also a logarithmic signature for G. If $t_0 = t_s = 1$, we say that β is a *sandwich* of α. The following proposition is proven in [12]:

Proposition 2.4. *Two logarithmic signatures of the same type r are equivalent if and only if one is a sandwich of the other.*

We are now able to give a small classification of the set $\Lambda(G)$:

Definition 2.5. Let G be a permutation group of degree n. A logarithmic signature for G is called

- *transversal,* if it is the sandwich of an exact transversal logarithmic signature,
- *non-transversal,* if it is not transversal,
- *totally-non-transversal,* if none of its blocks is a coset of a non-trivial subgroup of G,
- *totally aperiodic,* if each block is an aperiodic set in G, i.e., a set which is not the union of (either left or right) cosets of a non-trivial subgroup H of G,

- *permutably transversal,* if a transversal logarithmic signature can be constructed permuting its blocks.

The sets of all transversal, non-transversal, totally-non-transversal, totally aperiodic, and permutably transversal logarithmic signatures for G are respectively denoted $\mathcal{T}(G)$, $\mathcal{NT}(G)$, $\mathcal{TNT}(G)$, $\mathcal{TA}(G)$, and $\mathcal{PT}(G)$. When no ambiguity is possible, we will simply write $\mathcal{T}$, $\mathcal{NT}$, etc.

Logarithmic signatures for a group can be used to generate permutations on the set $\mathbb{Z}_{|G|}$, for given a fixed $\eta \in \Lambda$, $\hat{\alpha} = \breve{\eta}^{-1}\breve{\alpha} \in S_{|G|}$, for all $\alpha \in \Lambda$. In fact, according to [12, Theorem 6.5] for any $\eta \in \mathcal{T}$, the set $\mathcal{T}_\eta = \{\hat{\alpha} = \breve{\eta}^{-1}\breve{\alpha} \colon \alpha \in \mathcal{T}\}$ generates "almost always" the full symmetric group. The importance of this result will become clear later.

The second way of factoring the elements of a finite group we will be interested in is induced by some other type of factorization sets, called $[s,r]$-meshes. The cryptosystem MST_2 is constructed under the assumption that such factorizations are hard to compute.

Definition 2.6. Let G be a finite group, and let $\alpha = [\alpha_1, \dots, \alpha_s]$ be a sequence where $s \in \mathbb{N}_0$ and for some $r \in \mathbb{N}_0$, each α_i $(1 \le i \le s)$ is itself a sequence $\alpha_i = [\alpha_{i0}, \dots, \alpha_{ir-1}]$ with $\alpha_{ij} \in G$ $(0 \le j < r)$. Then we call α an $[s,r]$-*mesh* for G provided that each $g \in G$ is represented as a product

$$g = \alpha_{1j_1} \cdots \alpha_{sj_s}$$

with $\alpha_{ij_i} \in \alpha_i$. Also, if there are a_g different factorizations for each element g, the probability distribution $\{P_g = a_g/r^s \colon g \in G\}$ should be approximately uniform.

Just as we used a logarithmic signature α for a permutation group G to define a bijection $\breve{\alpha} \colon \mathbb{Z}_{|G|} \longrightarrow G$, we can use an $[s,r]$-mesh β for G to define a surjection $\breve{\beta} \colon \mathbb{Z}_{r^s} \longrightarrow G$. Surjections of this kind are used in the encryption procedure of MST_2.

3. The public key cryptosystem MST_1

We briefly describe the system MST_1 which is introduced in [13], see also [3]: let G be a permutation group of degree n and fix a supertame logarithmic signature η for G (both public). Alice's public key consists of a pair $(\alpha, \beta) \in \Lambda \times \Lambda$ with α wild and β tame. Her private key is a sequence $[\theta_1, \dots, \theta_k] \in \mathcal{T}^k$ (with $k \ge 2$ a small integer) such that

$$\hat{\beta}^{-1}\hat{\alpha} = \hat{\theta}_1 \cdots \hat{\theta}_k .$$

To send a plaintext message $m \in \mathbb{Z}_{|G|}$ to Alice, Bob computes the ciphertext $\mathbf{c} = \hat{\beta}^{-1}\hat{\alpha}(m)$, which can be decrypted by Alice in polynomial time, for

$$m = \hat{\alpha}^{-1}\hat{\beta}(\mathbf{c}) = \hat{\theta}_k^{-1} \cdots \hat{\theta}_1^{-1}(\mathbf{c}).$$

This choice of a private key is motivated by the fact that the permutation group generated by transversal signatures is often the whole $S_{|G|}$ (see [12, Theorem 6.5]), though in principle any additional information that would allow Alice to compute $\hat{\alpha}^{-1}$ could be used as a private key.

Of course the question arises of how to specify the aforementioned parameters for deriving concrete public key/private key pairs. Unfortunately, to our knowledge no concrete proposals in this direction have been made so far—e.g., in [12] a possible group size of 48! is mentioned, but no concrete suggestion for choosing the permutation group G has been made. In fact, in [12, Section 4.2] the authors of MST_1 write "Consequently, at the present time, we do not know whether the trap doors indicated in the proposed scheme can be constructed efficiently."

In the sequel we want to point out some obstacles that arise when trying to generate concrete instances of MST_1. But before doing so, we want to give some comments on the possible key sizes in MST_1.

3.1 Minimal length for a logarithmic signature

Suppose the group G is fixed, what can we tell then about the size of the public key $(\alpha, \beta) \in \Lambda \times \Lambda$? If we define the *length* of a logarithmic signature $\alpha = [\alpha_1, \dots, \alpha_s]$ as the integer $\ell(\alpha) = \sum_{i=1}^{s} r_i$, where $|\alpha_i| = r_i$, $1 \le i \le s$, how small can we expect $\ell(\alpha)$ to be? The following lower bound is straightforward to prove:

Remark 3.1. Let G be a permutation group and $|G| = \prod_{j=1}^{t} p_j^{a_j}$ the prime factorization of the order of G (with $p_1, \dots, p_t$ the different prime factors of $|G|$). Then for any $\alpha \in \Lambda$ we have $\ell(\alpha) \ge \sum_{j=1}^{t} a_j p_j$.

Proof. Let $\alpha = [\alpha_1, \dots, \alpha_s]$ with $|\alpha_i| = r_i$ for $1 \le i \le s$. Then from $\alpha \in \Lambda$ we conclude that $|G| = r_1 \cdots r_s$, i.e., the prime factorization of r_i $(1 \le i \le s)$ has the form $r_i = \prod_{j=1}^{t} p_j^{a_{ij}}$, where $0 \le a_{ij} \le a_j$, and for $1 \le j \le t$ we have $\sum_{i=1}^{s} a_{ij} = a_j$.

Now the claim follows from the trivial inequality $r_i \ge \sum_{j=1}^{t} a_{ij} p_j$ $(1 \le i \le s)$. □

For solvable and symmetric groups we give an elementary constructive proof of the sharpness of this bound:

Proposition 3.2. *Let G be a group of order $|G| = \prod_{i=1}^{t} p_i^{a_i}$ (with $p_1, \dots, p_t$ the different prime factors of $|G|$), and define the integer l_G as $l_G = \sum_{i=1}^{t} a_i p_i$. Then if G is*

(i) *a solvable group*

or

(ii) *a symmetric group S_k,*

it is possible to construct a logarithmic signature α for G of length $\ell(\alpha) = l_G$.

P r o o f. (i) As the empty product evaluates to 1, for $G = \{1\}$ we can set $\alpha := []$ (the empty sequence); in the sequel we assume w.l.o.g. $G \neq \{1\}$.

The composition factors of G are cyclic groups of prime order. Therefore, if we take a composition series of G, say $\gamma \equiv G = G_0 \triangleright G_1 \triangleright \dots \triangleright G_k = \{1\}$, we have for $i = 1, \dots, k$, $[G_{i-1} : G_i] \in \{p_1, \dots, p_t\}$ and

$$\prod_{i=1}^{k} [G_{i-1} : G_i] = \prod_{i=1}^{t} p_i^{a_i}.$$

Clearly, any exact transversal logarithmic signature we construct from γ will have length $\sum_{i=1}^{k} [G_{i-1} : G_i] = \sum_{i=1}^{t} a_i p_i = l_G$, and the result follows.

(ii) The proof is made by induction on the degree k of the symmetric group.[1] For $k \leq 4$ the statement follows from i., as in this case S_k is solvable. So let us now suppose we have constructed a logarithmic signature $\alpha^{k-1} = [\alpha_1^{k-1}, \dots, \alpha_m^{k-1}]$ for S_{k-1} of length $l_{S_{k-1}}$ already. Next, recall that if for $1 \leq i \leq k$ we are given elements $g_i \in S_k$ such that $g_i(i) = k$, then we have

$$S_k = \bigcup_{i=1}^{k} S_{k-1} g_i.$$

Writing k in the form $k = p_1 \cdots p_r$, with $p_1, \dots, p_r$ not necessarily distinct prime numbers, we will construct r new blocks, $\alpha_1, \dots, \alpha_r$, with $|\alpha_i| = p_i$ and such that for $1 \leq i \leq k$ we can construct a permutation g_i with $g_i(i) = k$ as a product comprised of one element of each block. Then the sequence $[\alpha_1^{k-1}, \dots, \alpha_m^{k-1}, \alpha_1, \dots, \alpha_r]$ will be a logarithmic signature for S_k of length l_{S_k}.

[1] In the sequel, when dealing explicitly with elements in S_k, we follow the convention that gh denotes the permutation constructed applying h first and then g.

The new blocks can be constructed in the following way:
first take

$$\alpha_1 = \{\mathrm{id}, (1,k), (2,k), \dots, (p_1 - 1, k)\},$$

then construct α_2, of length p_2 so that the permutations in it allow us to map each of the numbers $i \in \{p_1, 2p_1, \dots, (p_2 - 1)p_1\}$ to k directly, and any other number j with $p_1 < j < p_1 p_2$ into the set $\{1, \dots, p_1 - 1\}$. Namely, we define

$$\begin{aligned}\alpha_2 = \Big[\ &\mathrm{id}, \\ &(k, p_1) \cdot \prod_{i=1}^{p_1-1} (i, p_1 + i), \\ &(k, 2p_1) \cdot \prod_{i=1}^{p_1-1} (i, 2p_1 + i), \\ &\vdots \\ &\big(k, (p_2 - 1)p_1\big) \cdot \prod_{i=1}^{p_1-1} \big(i, (p_2 - 1)p_1 + i\big)\Big].\end{aligned}$$

Hence, the "partial logarithmic signature" $[\alpha_1, \alpha_2]$ enables us to construct permutations $g_i \in S_k$ with $g_i(i) = k$ for $1 \le i \le p_1 p_2 - 1$. Defining $\alpha_3, \dots, \alpha_{r-1}$ analogously, we finally construct the block α_r so that each element in

$$\{p_1 p_2 \cdots p_{r-1}, 2p_1 p_2 \cdots p_{r-1}, \dots, (p_r - 1)p_1 \cdots p_{r-1}\}$$

can be mapped to k directly, and any other number j with $p_1 \cdots p_{r-1} < j < k$ can be mapped into the set $\{1, \dots, p_1 p_2 \cdots p_{r-1} - 1\}$ by a permutation in α_r:

$$\begin{aligned}\alpha_r = \Big[\ &\mathrm{id}, \\ &(k, p_1 p_2 \cdots p_{r-1}) \cdot \prod_{i=1}^{p_1 p_2 \cdots p_{r-1} - 1} (i, p_1 p_2 \cdots p_{r-1} + i), \\ &(k, 2p_1 p_2 \cdots p_{r-1}) \cdot \prod_{i=1}^{p_1 p_2 \cdots p_{r-1} - 1} (i, 2p_1 p_2 \cdots p_{r-1} + i), \\ &\vdots \\ &\big(k, (p_r - 1)p_1 p_2 \cdots p_{r-1}\big) \cdot \prod_{i=1}^{p_1 p_2 \cdots p_{r-1} - 1} \big(i, (p_r - 1)p_1 p_2 \cdots p_{r-1} + i\big)\Big].\end{aligned}$$

It is now easy to check that $\alpha^k = [\alpha_1^{k-1}, \dots, \alpha_m^{k-1}, \alpha_1, \dots, \alpha_r]$ is a logarithmic signature for S_k of length l_{S_k}. □

Having in mind efficiency, it would be desirable that the two public logarithmic signatures α, β in MST_1 could be chosen in such a way that $\ell(\alpha)$ and $\ell(\beta)$ meet the bound in Remark 3.1. Unfortunately, Proposition 3.2 resp. its proof does not give a hint of how one could construct a logarithmic signature that is both wild and of minimal possible length.

But even without having a concrete suggestion for choosing the underlying group G and the public key $(\alpha, \beta) \in \Lambda \times \Lambda$, the bound from Remark 3.1 yields some information about potential instances of MST_1: with the suggested group size $|G| = 48!$ we would have $\ell(\alpha) + \ell(\beta) \geq 2 \cdot 679$. Representing each element in the logarithmic signatures with $\log_2(48!)$ bit, this would result in a public key of more than 2^{18} bit.—In particular, this is significantly larger than a typical public key size of $\leq 2^{12}$ bit (as, for instance, in the RSA system). Unfortunately, even leaving aside the question of efficiency, deriving secure keys for MST_1 still seems to be quite involved—the next section demonstrates that the restrictions on logarithmic signatures proposed in [13] are not sufficient for obtaining acceptable cryptographic security.

3.2 Hardness of constructing secure keys

We now have a lower bound on the size of the public key (α, β), but so far still ignore what kind of logarithmic signatures we are looking for. As "wild" stands for "computationally hard to invert" we cannot expect to find an algorithmic way of checking whether a given signature is wild. However, some theoretic results should support the suggestion of using a certain type of signatures as wild ones.

In [13] the authors allow the cryptographic assumption that logarithmic signatures in $\mathcal{TNT}$ are "wild-like". Subsequently we show that using logarithmic signatures contained in $\mathcal{TNT}$ is not a sufficient condition for offering acceptable cryptographic security in MST_1. To avoid critical security flaws, additional conditions have to be imposed on the public key.

One situation which should indeed be avoided is the case in which some parts of the wild logarithmic signature $\alpha = [\alpha_1, \ldots, \alpha_s]$ constitute a tame logarithmic signature for some subgroup H of G with $[G\colon H]$ small (from a cryptographic point of view say, e.g., $[G\colon H] < 2^{80}$). Namely, suppose we can construct a tame logarithmic signature $\alpha' = [\alpha'_1, \ldots, \alpha'_s]$, for some $H \leq G$ such that $\alpha'_i \subseteq \alpha_i$ for $1 \leq i \leq s$. Then the attacker is able to factor each element $g \in H$, and if $[G\colon H]$ is too small the system cannot be considered as secure.

Example 3.3. To illustrate the problem just mentioned, we can use the following logarithmic signature

$\alpha = [\alpha_1, \alpha_2] \in \mathcal{TNT}(A_5)$ given in [12]:

$$\begin{array}{llllll}
\alpha_1 = [& \mathrm{id}, & (1,5,2), & (1,3)(2,4), & (2,3,4), & (2,5,4), \\
 & (1,3,5,2,4), & (2,3)(4,5), & (1,5,4,3,2), & (1,5,3,4,2), & (3,4,5), \\
 & (2,5,3), & (3,5,4), & (2,5)(3,4), & (2,3,5), & (1,3,2,4,5)], \\
\alpha_2 = [& (1,3)(4,5), & (1,4)(3,5), & \mathrm{id}, & (1,3,5)]. &
\end{array}$$

Setting $\alpha_1' = \big[\mathrm{id}, (3,4,5), (3,5,4)\big] \subset \alpha_1$ and $\alpha_2' = \alpha_2$ we obtain an exact transversal logarithmic signature $\alpha' := [\alpha_1', \alpha_2']$ for the subgroup

$$\big\langle (1,3,5), (3,5,4) \big\rangle \simeq A_4$$

of A_5. In other words, for 20 % of the elements in A_5 the factorization with respect to $\alpha \in \mathcal{TNT}(A_5)$ can be found through factoring with respect to the exact transversal logarithmic signature α'.

A similar situation arises if parts of the "wild-like" signature α can somehow be embedded into a tame one, α^*, and subsequently some kind of *partial inversion* of $\breve{\alpha}$ can be performed. Here we illustrate such a situation with a toy example based on a totally-non-transversal logarithmic signature of a symmetric group:

Example 3.4. Using a computer algebra system like MAGMA [2], for instance, the following logarithmic signature $\alpha = [\alpha_1, \alpha_2, \alpha_3]$ of the symmetric group S_4 can easily be verified to be totally-non-transversal:

$$\begin{aligned}
\alpha_1 &= [\ \mathrm{id},\ \ (1,4,3),\ \ (3,4)\ ], \\
\alpha_2 &= [\ \mathrm{id},\ \ (1,2,4)\ ], \\
\alpha_3 &= [\ \mathrm{id},\ \ (1,2),\ \ (1,3)(2,4),\ \ (1,3,2,4)\ ].
\end{aligned}$$

Next, consider the following logarithmic signature $\alpha^* = [\alpha_1^*, \alpha_2^*, \alpha_3^*] \in \mathcal{E}(S_4)$:

$$\begin{aligned}
\alpha_1^* &= [\ \mathrm{id},\ \ (1,4,3),\ \ (3,4),\ \ (2,4,3)\ ], \\
\alpha_2^* &= [\ \mathrm{id},\ \ (1,2,4),\ \ (1,4,2)\ ], \\
\alpha_3^* &= [\ \mathrm{id},\ \ (1,2)\ ].
\end{aligned}$$

Then we have $\alpha_1 \subset \alpha_1^*$ and $\alpha_2 \subset \alpha_2^*$. Moreover, the blocks α_3 and α_3^* have two elements in common. Hence, for $3 \cdot 2 \cdot 2 = 12$ elements of S_4, i.e., for 50 % of the group elements, we can find the factorization with respect to $\alpha \in \mathcal{TNT}(S_4)$ by factoring it with respect to $\alpha^* \in \mathcal{E}(S_4)$.

Our discussion gives evidence that for the "wild" logarithmic signature in a public key of MST_1, the requirement of being totally-non-transversal is not sufficient. Even if we know a logarithmic signature to be totally aperiodic, it is not clear whether it withstands the kinds of attack just described. Consequently, using an approach as described in [13, Section 4.2] for deriving public keys in MST_1 seems critical: the latter procedure checks whether a signature is totally-non-transversal, but does not take into account the feasibility of "partial inversion attacks" as indicated in the above examples.

Finally, we would like to remark that the problem of "partial inversion" is also of concern for the approach considered by C. Cusack in [3, Section 3.7]. There C. Cusack raises the question of whether permutably transversal logarithmic signatures of abelian groups could be used for deriving the trap door required in MST_1. Although he specifies no concrete procedure for a key generation algorithm based on this idea, it is worth pointing out that his discussion ignores the problem of partial inversion completely. However, for a realistic key generation procedure, resistance against such an attack is an essential issue.

4. The public key cryptosystem MST_2

The second system we will focus on is based on so-called $[s,r]$-meshes. We give a brief description of it, for further details see [3, 13].

Let G be a finite group and H a group such that there exists an epimorphism $f\colon G \to H$. This epimorphism constitutes the private key. Alice's public key will be the quadruple (G, H, α, β), where α is a randomly chosen $[s,r]$-mesh for G and $\beta = f(\alpha)$ is an $[s,r]$-mesh for H.

If Bob wants to send the message $h \in H$ to Alice, he chooses a random integer $R \in \mathbb{Z}_{r^s}$ and computes the elements $y_1 = \breve{\alpha}(R)$, $y_2 = \breve{\beta}(R)$ and $y_3 = hy_2$. Then he sends the ciphertext $y = (y_1, y_3)$ to Alice. She will be able to recover the message by computing $y_2 = f(y_1)$ and then using $h = y_3 y_2^{-1}$.

Note that the security of the system relies on the assumption that $[s,r]$-meshes are hard to invert. For the ElGamal scheme in suitable cyclic groups, which can be seen as a special case of MST_2, this assumption seems to be fair.[2] The idea there is that computing the discrete logarithm in $\mathbb{F}_{2^n}^*$ with respect to a primitive element $a \in \mathbb{F}_{2^n}^*$ can be regarded as factoring with respect to the mesh $\alpha = \big[[1, a^{2^0}], \dots, [1, a^{2^{n-1}}]\big]$. But note that if we formulate the ElGamal system as an instance of MST_2, then one part of the public key consists of the above mesh α and for representing the element $a^{2^{i-1}}$ in the ith block ($1 \le i \le n$) of α, in principle we have to take n bits into account. Having n blocks, already these elements alone imply a length of the public key of at least n^2 bits. Hence,

[2] Of course, in this case the groups G and H in MST_2 are not represented as permutation groups.

taking also the (for practical purposes important) question of key length into account, MST_2 loses some of its attractiveness.

Moreover, for the "general" MST_2 system we again face a situation in which hardly any information is provided of how to choose a suitable group or appropriate key lengths for a secure implementation.—It is quite obvious that not every $[s,r]$-mesh induces a hard factorization, for, e.g., from any (tame) logarithmic signature of appropriate size one can derive an $[s,r]$-mesh that induces the same factorization on the group elements. Thus, we remark the need of introducing the concepts of wild and tame meshes in a similar way as it has been done for logarithmic signatures.

As evidence for the security of MST_2 the authors relate it to the problem of computing the intersection of centralizers in a finite group, but they leave open the problem of finding appropriate groups for constructing secure instances. The most concrete suggestion for an implementation of MST_2 we are currently aware of (see [13]) proposes using non-abelian groups and taking $H = G$ and f the conjugation by an element $g \in G$.

Unfortunately, if G is represented as a permutation group, such a construction is quite critical: suppose G is a permutation group of degree n, and the epimorphism f is conjugation by a permutation $g \in G$. Then indeed much information about g is available through the public channel: if the block elements of the public meshes α, β are given as products of disjoint cycles, we have at hand a system of simultaneous conjugacy equations

$$\alpha_{i0}^g = \beta_{i0}, \dots, \alpha_{ir-1}^g = \beta_{ir-1} \qquad (1 \le i \le s),$$

which provides a lot of information about the permutation g.

Recall that for a product of disjoint cycles $\tau = c_1 \cdots c_t$ we have $\tau^g = c_1^g \cdots c_t^g$, and for each cycle $(i_1, \dots, i_l)$ we obtain $(i_1, \dots, i_l)^g = \big(g(i_1), \dots, g(i_l)\big)$.

Hence, a simple comparison of the permutations/cycles at corresponding positions of α and $\beta = \alpha^g$ can reveal a critical amount of information about the secret element g. The following toy example based on a $[3,4]$-mesh of the alternating group A_5 illustrates this problem:

EXAMPLE 4.1. Consider the following $[3,4]$-mesh $\alpha = [\alpha_1, \alpha_2, \alpha_3]$ for the alternating group A_5:

$$\begin{aligned}
\alpha_1 &= [\ (1,2,4,5,3), \quad (1,4,5), \quad (2,3)(4,5), \quad (1,3,4,5,2)\ ], \\
\alpha_2 &= [\ (1,4,5), \quad (1,3)(2,5), \quad (1,3,5,4,2), \quad (1,3,2,4,5)\ ], \\
\alpha_3 &= [\ (3,4,5), \quad (1,3,4), \quad (1,4,3), \quad \mathrm{id}\ ].
\end{aligned}$$

Conjugating α with $g = (1,4,2,3,5) \in A_5$ yields a $[3,4]$-mesh $\beta = [\beta_1, \beta_2, \beta_3]$:

$$\begin{aligned}
\beta_1 &= [\ (1,5,4,3,2), \quad (1,4,2), \quad (1,2)(3,5), \quad (1,3,4,5,2)\ ], \\
\beta_2 &= [\ (1,4,2), \quad (1,3)(4,5), \quad (1,2,3,4,5), \quad (1,4,5,3,2)\ ], \\
\beta_3 &= [\ (1,5,2), \quad (2,4,5), \quad (2,5,4), \quad \mathrm{id}\ ].
\end{aligned}$$

Comparing the first elements in α_3 and β_3 we recognize that the (secret) conjugating element g maps the set $\{3,4,5\}$ onto $\{1,5,2\}$. Similarly, from the second elements in α_3 and β_3 we conclude $g(\{1,3,4\}) = \{2,4,5\}$. So we must have $g(5) \in \{1,5,2\} \setminus \{2,4,5\}$, i.e., $g(5) = 1$. In the same way we conclude that $g(1) \in \{2,4,5\} \setminus \{1,5,2\}$, and hence $g(1) = 4$.

From the second elements in α_2 and β_2 we see that $g\big((1,3)\big) \in \big\{(1,3),(4,5)\big\}$ holds, and with $g(1) = 4$ we get $g\big((1,3)\big) = (4,5)$. Consequently, $g(3) = 5$.

Using the equations $g\big((1,3)(2,5)\big) = (1,3)(4,5)$ and $g\big((1,3)\big) = (4,5)$ we derive $g\big((2,5)\big) = (1,3)$, i.e., with $g(5) = 1$ we obtain $g(2) = 3$. Now we know the images of 1, 2, 3, and 5 under g already, and we must have $g(4) = 2$.—In other words, we have recovered g completely.

Deriving a practical instance of MST_2 that allows the use of a permutation representation of the ambient group G seems to be rather involved. In particular, any suggestion for an implementation of MST_2 should specify how to represent the occurring group elements, as this might be a crucial issue for the security of the resulting cryptosystem.

5. Conclusions

The above discussion illustrates various obstacles that arise when one attempts to derive practical instances of MST_1 and MST_2: it is unclear how to generate logarithmic signatures α that are computationally hard to invert for (almost) all group elements. The given examples illustrate that the suggested requirement $\alpha \in \mathcal{TNT}$ does not guarantee acceptable cryptographic security. Having also in mind practical key sizes, it is unclear how a realistic key generation procedure for MST_1 should look like.

For MST_2 the situation is similar: it is unclear how practical instances of the system can be derived. In particular, if a natural permutation group representation is used, then serious obstacles in the suggested variant of MST_2 arise.

Acknowledgements. The authors would like to thank Thomas Beth, Jens-Matthias Bohli, Consuelo Martínez, and Martin Rötteler for several helpful comments and discussions.

REFERENCES

[1] ANSHEL, I.—ANSHEL, M.—GOLDFELD, D.: *An algebraic method for public-key cryptography*, Math. Res. Lett. **6** (1999), 1–5.

[2] BOSMA, W.—CANNON, J.—PLAYOUST, C.: *The magma algebra system I: The user language*, J. Symbolic Comput. **24** (1997), 235–265.

[3] CUSACK, C. A.: *Group Factorizations in Cryptography*, PhD thesis, University of Nebraska, 2000.

[4] ElGAMAL, E.: *A public key cryptosystem and a signature scheme based on discrete logarithms*, IEEE Trans. Inform. Theory **31** (1985), 469–472.

[5] GARZON, M.—ZALCSTEIN, Y.: *The complexity of Grigorchuk groups with application to cryptography*, Theoret. Comput. Science **88** (1991), 83–98.

[6] GROŠEK, O.—MAGLIVERAS, S. S.—WEI, W.: *On the security of a public-key cryptosystem*, Proceedings of the Public-key Cryptography and Computational Number Theory Conference, Stefan Banach International Mathematics Center, Warsaw, Poland, Walter de Gruyter, Berlin, 2001, pp. 71–75.

[7] KO, K. H.—LEE, S. J.—CHEON, J. H.—HAN, J. W.—KANG, J.—PARK, C.: *New Public-Key Cryptosystem Using Braid Groups*, Advances in Cryptology. Proceedings of CRYPTO '00 (M. Bellare, ed.), Santa Barbara, California, USA, Lecture Notes in Comput. Sci., Vol. 1880, Springer-Verlag, Berlin, 2000, pp. 166–183.

[8] MAGLIVERAS, S. S.: *A cryptosystem from logarithmic signatures of finite groups*, in: Proceedings of the 29'th Midwest Symposium on Circuits and Systems, (Elsevier Amsterdam), 1986, pp. 972–975.

[9] MAGLIVERAS, S. S.—MEMON, N. D.: *Complexity tests of cryptosystem PGM*, Congr. Numer. **79** (1990), 61–68.

[10] MAGLIVERAS, S. S.—MEMON, N. D.: *Properties of cryptosystem PGM*, Advances in Cryptology, Proceedings of CRYPTO '89, (G. Brassard, ed.), Lecture Notes in Comput. Sci., Vol. 435, Springer-Verlag, Berlin, 1990, pp. 447–460.

[11] MAGLIVERAS, S. S.—MEMON, N. D.: *The linear complexity profile of the cryptosystem PGM*, Congr. Numer. **72** (1990), 51–60.

[12] MAGLIVERAS, S. S.—MEMON, N. D.: *Algebraic properties of cryptosystem PGM*, J. Cryptology **5** (1992), 167–183.

[13] MAGLIVERAS, S. S.—STINSON, D. R.—VAN TRUNG, T.: *New approaches to designing public key cryptosystems using one-way functions and trap-doors in finite groups*, Tech. Rep. CORR 2000-49, Centre for Applied Cryptographic Research, University of Waterloo, 2000, At the time of writing available electronically at the URL `http://www.cacr.math.uwaterloo.ca/techreports/2000/corr2000-49.ps`.

[14] NGUYEN, P.—STERN, J.: *Merckle-Hellman Revisited: A Cryptanalysis of the Qu-Vanstone Cryptosystem Based on Group Factorizations*, Advances in Cryptology. Proceedings of CRYPTO '97, (B. S. Kaliski, ed.), Lecture Notes in Comput. Sci., Vol. 1294, Springer-Verlag, Berlin, 1997, pp. 198–212.

[15] QU, M.—VANSTONE, S. A.: *Factorizations in the elementary Abelian p-group and their cryptographic significance*, J. Cryptology **7** (1994), 201–212.

[16] QU, M.—VANSTONE, S. A.: *The Knapsack problem in cryptography*, Finite Fields: Theory, Applications, and Algorithms **168** (1994), 291–308.

[17] SHOR, P.: *Polynomial time algorithms for prime factorization and discrete logarithms on quantum computer*, SIAM J. Comput. **26** (1997), 1484–1509.

[18] WAGNER, N. R.—MAGYARIK, M. R.: *A Public Key Cryptosystem Based on the Word Problem*, Advances in Cryptology, Proceedings of CRYPTO '84, (G. R. Blakley; D. Chaum, ed.), Lecture Notes Comput. Sci., Vol. 196, Springer-Verlag, Berlin, 1985, pp. 19–36.

[19] WEBB, W.: *Cryptography using complementing subsets of polynomials over finite fields*, in: Proceedings of the international conference on finite fields, coding theory, and advances

in communications and computing, August 7–10, 1991, (Marcel Dekker, Inc., New York), Lecture Notes Pure Appl. Math., Vol. 141, 1993, pp. 411–415.

Received December 20, 2001

María Isabel González Vasco
Departamento de Matemáticas
Universidad de Oviedo
c/Calvo Sotelo, s/n
33007 Oviedo
SPAIN
E-mail: mvasco@orion.ciencias.uniovi.es

Rainer Steinwandt
Institut für Algorithmen und Kognitive Systeme
Arbeitsgruppe Systemsicherheit
Prof. Dr. Th. Beth
Universität Karlsruhe
D–76128 Karlsruhe
GERMANY
E-mail: steinwan@ira.uka.de