MST3 Cryptosystem Based on a Generalized Suzuki 2-Groups

Gennady Khalimov[1] [0000-0002-2054-9186], Yevgen Kotukh[2] [0000-0003-4997-620X],
Svitlana Khalimova[3] [0000-0001-7224-589X]

[1]Kharkiv National University of Radioelectronics, Kharkiv, Ukraine
hennadii.khalimov@nure.ua
[2]University of Customs and Finance, Dnipro, Ukraine
yevgenkotukh@gmail.com
[3]Kharkiv National University of Radioelectronics, Kharkiv, Ukraine
svitlana.*khalimova*@*nure*.ua

Abstract. The article describes a new implementation of MST3 cryptosystems based on the generalized Suzuki 2 - groups. The main difference in the presented implementation is the presence of many-stage recovery of parts of the message from the encrypted text. The presented implementation of a cryptosystem has lower costs of key data. The complexity of the cryptanalysis and the size of the message for encryption depend of the power a generalized Suzuki 2 - groups.

Keywords: MST cryptosystem, logarithmic signature, random cover, generalized Suzuki 2 - groups

1 Introduction

Current is the development of efficient cryptographic cryptosystems that can with stand quantum attacks. It is believed that the advent of quantum computers and the presence of quantum algorithms for factoring integers and discrete logarithms will lead to hacking of known cryptosystems with a public key.

The idea of constructing public-key cryptosystems on the basis of an intractable word problem was proposed by Wagner and Magyarik in [1]. The basis is the use of permutation groups. Since the 2000s, several dozen cryptosystems in group constructions have been proposed [2÷5].

The development of the idea of Wagner and Magyarik is the proposal of Magliveras [6]. Magliveras proposed a symmetric cryptosystem based on a special type of factorization of finite groups named logarithmic signatures for finite permutation groups. The idea of using a logarithmic signature was investigated by Lempken et al. and developed in the construction for random covers in [7].

In this scheme, the public key consists of a tame logarithmic signature as well as some random numbers, and the secret key is design of random cover and sandwich transformation of the cover [8].

The intractability assumptions of this scheme are group factorization problem on nonabelian groups.

Magliveras cryptosystem based on the Suzuki group is known as MST3. Further improvements to this scheme were made by Svaba and van Trung in [9]. They introduced a secret cover of a random cover. A digital signature scheme based on MST3 cryptosystems was proposed and explored in [Hong].

Using two Suzuki parametric groups to build the MST3 cryptosystem leads to security and complexity of the assessment, proportional to the square of the measurement of the final field.

A further increase in security is possible by expanding the group. The MST3 cryptosystem based on the three-parameter group of automorphisms of the functional field of the Hermite curve [14] and the small Ri group has security and complexity estimates proportional to the cube of the dimension of the finite field [15].

In this paper will be presentation MST3 cryptosystems based on the multi-parameter generalized Suzuki 2 - groups.

2 The generalized Suzuki 2 - groups

The construct a generalizations of Suzuki 2-groups was proposed by Hakai in [16] at research conjugacy classes and characters for a family of groups satisfying Bannai condition. Construction of groups.

Let F_q , $q = 2^n$ is the finite field, and θ is an automorphism of F. Define for a positive integer l and $a_1, a_2, ..., a_l \in F$ the following matrix

$$S(a_1, a_2, ..., a_l) = \begin{pmatrix} 1 & & & & & \\ a_1 & 1 & & & & \\ a_2 & a_1\theta & 1 & & & \\ a_3 & a_2\theta & a_1\theta^2 & 1 & & \\ ... & ... & ... & ... & ... & \\ a_l & a_{l-1}\theta & a_{l-2}\theta^2 & ... & a_1\theta^{l-1} & 1 \end{pmatrix} \in M_{l+1}(F)$$

and

$$A_l(n,\theta) = \left\{ S(a_1, a_2, ..., a_l) \mid a_i \in F_q \right\}.$$

The each element of $A_l(n,\theta)$ can be expressed uniquely and it follows that $|A_l(n,\theta)| = 2^{nl}$ and $A_l(n,\theta)$ define a group of order 2^{nl} . If $l = 2$, this group is isomorphic to a Suzuki 2-group $A(n,\theta)$.

Group operation is defined as a product of two matrices

$$S(a_1, a_2, ..., a_l)S(b_1, b_2, ..., b_l) = S(a_1 + b_1, a_2 + (a_1\theta)b_1 + b_2, a_3 + (a_2\theta)b_1 + (a_1\theta^2)b_2 + b_3,$$
$$..., a_l + (a_{l-1}\theta)b_1 + ... + (a_1\theta^{l-1})b_{l-1} + b_l).$$

Identity element is unit diagonal matrix $S(0_1, 0, ..., 0)$ .The inverse element is determined by the inverse of the matrix. The direct calculations can to show that

$$S(a_1, a_2, a_3, ..., a_l)^{-1} = S(a_1, a_2 + a_1\theta a_1, a_3 + a_2\theta a_1 + a_1\theta^2(a_2 + a_1\theta a_1), ..., a_l + a_{l-1}\theta a_1 + ...).$$

Put $G = A_l(n,\theta)$. The group G is nonabelian group and has nontrivial center

$$Z(G) = \left\{ S(0,0,...,c) \middle| c \in F_q \right\}.$$

Since the center $Z\left(G\right)$ is elementary abelian of order *q*, it can be identified with the additive group of the field F_q . Assume that θ is the Frobenius automorphism of $F, \theta : x \to x^2$. All the involutions of G are in the center $Z\left(G\right)$. Define

$$G_i = \left\{S(0,...,0,a_i,a_{i+1},...,a_l)\right\} .$$

The elements $g \in G_i$ have order 2^{l-i} .

Let $G/G_i \triangleq A_{i-1}(n,\theta)$. Thus G_i is a normal subgroup of $A_l(n,\theta)$. Simple show that group G_i for $i \geq (l+1)/2$ is abelian. This holds by direct calculations.

Let $\lambda \in F^{\times}$, λ is a generator of $F^{\times}$ and is fixed and define mapping $\xi_\lambda : A_l(n,\theta) \to A_l(n,\theta)$ by

$$\xi_\lambda\left(S(a_1,a_2,...,a_l)\right) = S(\lambda_1 a_1, \lambda_2 a_2,...,\lambda_l a_l)$$

where

$$\lambda_1 = \lambda,\ \ \lambda_2 = \lambda\lambda^2,\ \lambda_3 = \lambda\left(\lambda^2\right)^2,\ ...,\ \lambda_i = \lambda^{2^{i-1}+1} .$$

Then mapping ξ_λ is an automorphism of $A_l(n,\theta)$ and if $(n,i)=1$, then ξ_λ permutes $G_i / G_{i+1} - G_{i+1}$ transitively. For the fixed finite field, the group $A_l(n,\theta)$ order is greater than classical Suzuki 2 - group. A larger group order gives an advantage to cryptosystem secrecy, and definition over small fields gives an advantage for the implementation in general.

3 MST cryptosystems

The basic idea of MST cryptosystems is to surjective mapping input message into an element of group using a conversion key. Formalization of computations is given by the following definition [12]. *Definition 1* (cover (logarithmic signature) mappings). Let $\alpha = \left[A_1,...,A_s\right]$ be a cover (logarithmic signature) of type $(r_1,r_2,...,r_s)$ for G with $A_i = \left[a_{i,1},a_{i,2},...,a_{i,r_i}\right]$, where $m = \prod_{i=1}^{s} r_i$. Let $m_1 = 1$ and $m_i = \prod_{j=1}^{i-1} r_j$ for $i = 2,...,s$. Let τ denote the canonical bijection

$$\tau : \mathbb{Z}_{r_1} \times \mathbb{Z}_{r_2} \times ... \times \mathbb{Z}_{r_s} \to \mathbb{Z}_m,$$

$$\tau\left(j_1, j_2,..., j_s\right) = \sum_{i=1}^{s} j_i \cdot m_i .$$

Then the surjective (bijection) mapping $\alpha' : \mathbb{Z}_m \to G$ induced by is

$$\alpha'\left(x\right) = a_{1j_1} \cdot a_{2j_2} \cdots a_{sj_s}$$

where $\left(j_1, j_2,..., j_s\right) = \tau^{-1}\left(x\right)$.

More generally, if $\alpha = \left[A_1,...,A_s\right]$ is a logarithmic signature (cover) for, then each element $g \in G$ ∈ can be expressed uniquely (at least one way) as a product of the form

$$g = a_1 \cdot a_2 \cdots a_s ,$$

for $a_i \in A_i$ [7].

Let G is ultimate nonabelian group with nontrivial center $\mathbf{Z}$, such that G does not decompose over $\mathbf{Z}$. Suppose that $\mathbf{Z}$ is quite large, such that the search is over $\mathbf{Z}$ is computational impracticable.

The cryptographic hypothesis, which is the basis for the cryptosystem, is that if $\alpha = [\mathrm{A}_1, \mathrm{A}_2, ..., \mathrm{A}_s] := (a_{i,j})$ – accidental cover for a "large" matrices S at G, then search for the layout $g = a_{1j_1} a_{2j_2} \cdots a_{sj_s}$ for any element $g \in G$ relatively α is, in general, not a solvable problem. There are several encryption algorithms for MST cryptosystems. One of the latest versions of MST3 presented in [13] has the following implementation.The main steps of the encryption algorithm. *Public and private key calculation step:*

- generating a tame logarithmic signatures $\beta = [B_1, B_2, ..., B_s] := (\mathrm{b}_{ij})$ the class $(r_1, r_2, ..., r_s)$ for $\mathbb{Z}$;
- generating a random cover $\alpha = [\mathrm{A}_1, \mathrm{A}_2, ..., \mathrm{A}_s] := (a_{i,j})$ the same class as i β for some subset J from G such that $\mathrm{A}_1, ..., \mathrm{A}_s \subseteq G \setminus \mathbf{Z}$;
- generating a set of elements $t_0, t_1 ..., t_s \in G \setminus \mathbf{Z}$;
- definition of the homomorphism to calculate$f : G \to \mathbb{Z}$;
- calculating $\gamma := (h_{ij}) = (t_{i-1}^{-1} f(a_{ij}) b_{ij} t_i)$ for $i = 1, ..., s$, $j = 1, ..., r_i$.

Will get public key – $(\alpha = (a_{ij}), \gamma = (h_{ij}), f)$ and private key – $(\beta = (\mathrm{b}_{ij}), (\mathrm{t}_0, ..., \mathrm{t}_s))$

Encryption step. Set a random number $R \in Z_{|\square|}$. Let the message to be encrypted $x \in \mathbf{Z}_{|Z|}$. Calculate

$$y_1 = \alpha'(R) \cdot x ,$$

$$y_2 = \gamma(R) = t_0^{-1} f(\alpha'(R)) b'(R) t_s .$$

Transmit $y = (y_1, y_2)$.

Decryption step. For decryption we have the cipher text $y = (y_1, y_2)$, private key $(\beta = (\mathrm{b}_{ij}), (\mathrm{t}_0, ..., \mathrm{t}_s))$ and the function of the homomorphism $f : G \to \square$.

Let's calculate $\beta(R) = y_2 t_s^{-1} f(y_1)^{-1} t_0$. Checking is determined by the fact that

$$y_2 = \gamma(R) = t_0^{-1} f(a_{1j_1}(R)) b_{1j_1}(R) t_1 \cdots t_{s-1}^{-1} f(a_{sj_s}(R)) b_{sj_s}(R) t_s$$

$$= b_{1j_1}(R) t_0^{-1} f(a_{1j_1}(R)) t_1 \cdots b_{sj_s}(R) t_{s-1}^{-1} f(a_{sj_s}(R)) t_s = \beta(R) t_0^{-1} f(\alpha(R)) t_s = \beta(R) t_0^{-1} f(y_1) t_s$$

Recover R with $\beta(R)$ using β^{-1}, because β is simple.

Calculate $x = \alpha'(R)^{-1} \cdot y_1$.

Complexity analysis.The main costs of implementation in the MST3 cryptosystem are determined by the volume a logarithmic signature over the finite field of the group representation.

The long-term key is defined by a logarithmic signature array β and vectors $t_0, t_1 ..., t_s \in G \setminus \mathbf{Z}$.

Let type a logarithmic signatures over the finite field F_q, $q = 2^n$ is $(r_1, ..., r_s)$ and $\prod_{i=1}^{s} r_i = 2^n$.

Suppose that the values r_i are approximately equal $r_i = 2^{n/s}$, then the size V of the logarithmic signature will have an estimate $s2^{n/s}$. For $q = 2^{512}$ and $s = 2^5, 2^6, 2^7, 2^8$ we obtain, respectively, $V = 2^{21}, 2^{14}, 2^{11}, 2^{10}$ of the 512 bit strings.

The large size of the logarithmic signature is a drawback to the practical implementation of the MST3 cryptosystem.

On the other hand, the large size of the logarithmic signature determines the potentially very large entropy of the long-term key for cryptographic transformations.

Security Analysis. Message x masked by a logarithmic signature on the arrays $\alpha = (a_{ij})$, $\gamma = (h_{ij})$ which is calculated for a random number $\boldsymbol{R}$.

The function of the homomorphism $f: G \to \mathbb{Z}$ moves the group element to the center of the group. The random values $\boldsymbol{R}$ are actually a session keys. Calculation $\beta(R) = y_2 t_s^{-1} f(y_1)^{-1} t_0$ and the subsequent recovery of R is possible due to the commutativity of the center. Commutative calculations in the center reduce the secrecy of the MST3 cryptosystem based on Suzuki 2-group to evaluate $\mathrm{O}(q^2)$.

4 MST3 cryptosystems based on the generalized Suzuki 2 - groups

We apply the above construction to build the MST3 cryptosystem based on the generalized Suzuki group.

Very large generalized Suzuki groups can be constructed over a finite field of fixed dimension. This allows to achieve a compromise between practical implementation and the cryptosystem secrecy. Description of the Scheme. Let's consider basic encryption steps in MST3 cryptosystem based on the generalized Suzuki group.

Key Generation:

Input: a large group $A_l(n,\theta) = \left\{ S(a_1, a_2, ..., a_l) \mid a_i \in F_q \right\}$, $q = 2^n$ with the center Z

Output: a public key $[f, (\alpha_k, \gamma_k)]$ with corresponding private key $\left[\beta_k, \left(t_{0(k)}, ..., t_{s(k)}\right)\right]$, $k = \overline{1, l/2}$.

Choose a tame logarithmic signatures $\beta_k = \left[B_{1(k)}, ..., B_{s(k)}\right] = \left(b_{ij}\right)_k = S\left(0, .., 0, b_{ij(l/2+k)}, 0, ..., 0\right)$ of type $\left(r_{1(k)}, ..., r_{s(k)}\right)$, $i = \overline{1, s}$, $j = \overline{1, r_{i(k)}}$, $b_{ij(l/2+k)} \in F_q$, $k = \overline{1, l/2}$. The tame logarithmic signature is defined as a bijective and factorizable map of $\beta_k(R)$. Select a random cover

$$\alpha_k = \left[A_{1(k)}, ..., A_{s(k)}\right] = \left(a_{ij}\right)_k = S\left(0, ..., 0, a_{ij(k)\cdot}^{(1)}, 0, ..., 0, a_{ij(l/2+k)}^{(2)}, 0, ..., 0\right)$$

of the same type as β, where $a_{ij} \in A_l(n,\theta)$, $a_{ij(k)}^{(1)}, a_{ij(k)}^{(2)} \in F_q \setminus \{0\}$, $i = \overline{1, s}$, $j = \overline{1, r_{i(k)}}$, $k = \overline{1, l/2}$.

Choose $t_{0(k)}, t_{1(k)}, ..., t_{s(k)} \in A_l(n,\theta) \setminus Z$, $t_{i(k)} = S(t_{i1(k)}, ..., t_{il(\mathrm{k})})$, $t_{ij(\mathrm{k})} \in F^{\times}$, $i = \overline{0,s}$, $j = \overline{1,l}$, $k = \overline{1,l/2}$. Let's $t_{\mathrm{s}(v)} = t_{0(v+1)}$, $v = \overline{1,l/2-1}$.

Construct a homomorphism f defined by $f\left(S(a_1,...,a_l)\right) = S(0,...,0,a'_{l/2+1} = a_1,...,a'_l = a_{l/2})$.

Compute $\gamma_k = \left[h_{1(k)},...,h_{s(k)}\right] = \left(h_{ij}\right)_k = t^{-1}_{(i-1)(k)} f\left(\left(a_{ij}\right)_k\right)\left(b_{ij}\right)_k t_{i(k)}$, $i = \overline{1,s}$, $j = \overline{1,r_i}$, $k = \overline{1,l/2}$, where $f\left(\left(a_{ij}\right)_k\right)\left(b_{ij}\right)_k = S\left(0,...,0,\left(a^{(1)}_{ij}\right)_{l/2+k} + \left(b_{ij}\right)_{l/2+k},0,...,0\right)$.

Output public key $\left[f,(\alpha_k,\gamma_k)\right]$, and private key $\left[\beta_k,\left(t_{0(k)},...,t_{s(k)}\right)\right]$, $k = \overline{1,l/2}$.

Encryption:

Input: a message $x \in G_{l/2+1}$, and the public key $\left[f,(\alpha_k,\gamma_k)\right]$, $k = \overline{1,l/2}$.

Output: a ciphertext $\left(y_1,y_2\right)$ of the message x.

Choose a random $R = (R_1,R_2,...,R_{l/2})$, $R_1, R_2, \ldots, R_{l/2_l} \in \mathbb{Z}_{|F_q|}$.

Compute:

$$y_1 = \alpha'(R)\cdot x = \alpha_1'\left(R_1\right)\cdot\alpha_2'\left(R_2\right)\cdot\alpha_3'\left(R_3\right)\cdots\alpha_{l/2}'\left(R_{l/2}\right)\cdot x$$
$$= S\left(a^{(1)}_1\left(R_1\right), a^{(1)}_2\left(R_2\right)+*,...,a^{(1)}_{l/2}\left(R_{l/2}\right)+*, a^{(2)}_{l/2+1}\left(R_1\right)+x_{l/2+1}+*,...,a^{(2)}_l\left(R_{l/2}\right)+x_l+*\right).$$

The components of $(*)$ in the formula are determined by cross-calculations in the group operation of the product.

Compute:

$$y_2 = \gamma'(R) = \gamma_1'\left(R_1\right)\cdot\gamma_2'\left(R_2\right)\cdots\gamma_{l/2}'\left(R_{l/2}\right)$$
$$= S\left(*,*,...,*,a^{(1)}_{l/2+1}\left(R_1\right)+\beta_{l/2+1}\left(R_1\right)+*,...,a^{(1)}_l\left(R_{l/2}\right)+\beta_l\left(R_{l/2}\right)+*\right).$$

Here, the $(*)$ components are determined by cross-calculations in the group operation of the product of $t_{0(k)},...,t_{s(k)}$, $k = \overline{1,l/2}$.

Output $\left(y_1,y_2\right)$.

Decryption:

Input: a ciphertext $\left(y_1,y_2\right)$ and private key $\left[\beta_k,\left(t_{0(k)},...,t_{s(k)}\right)\right]$, $k = \overline{1,l/2}$.

Output: the message $x \in G_{l/2+1}$ corresponding to ciphertext $\left(y_1,y_2\right)$.

To decrypt a message x, we need to restore random numbers $R = (R_1,R_2,...,R_{l/2})$

The parameter $a^{(1)}_1\left(R_1\right)$ is known from the y_1 as the first parameter and it is included in the $l/2+1$ component of y_2, because $a^{(1)}_{l/2+1}\left(R_1\right) = a^{(1)}_1\left(R_1\right)$. Compute:

$$D^{(1)}(R_1,R_2,...,R_{l/2}) = t_{0(1)}\cdot y_2 t^{-1}_{s(l/2)}$$
$$= S\left(0,...,0,a^{(1)}_{l/2+1}\left(R_1\right)+\beta_{l/2+1}\left(R_1\right),...,a^{(1)}_l\left(R_{l/2}\right)+\beta_l\left(R_{l/2}\right)\right).$$

$$D^*(R) = D^{(1)}(R_1,R_2,...,R_{l/2})\,f\left(y_1\right)$$
$$= S\left(0,...,0,\beta_{l/2+1}\left(R_1\right),a^{(1)}_{l/2+2}\left(R_2\right)+\beta_{l/2+2}\left(R_2\right)+*,...\right).$$

Restore R_1 with $\beta_{l/2+1}\left(R_1\right)$ using $\beta_{l/2+1}\left(R_1\right)^{-1}$, because β is simple.

For further calculations, it is necessary to remove the components of the arrays $\alpha_1'\left(R_1\right)$ and $\gamma_1'\left(R_1\right)$ from ciphertext $\left(y_1, y_2\right)$.

Compute:

$$\begin{aligned} y_1^{(1)} &= \alpha_1'\left(R_1\right)^{-1} \cdot y_1 = \alpha_2'\left(R_2\right) \cdot \alpha_3'\left(R_3\right) \cdots \alpha_{l/2}'\left(R_{l/2}\right) \cdot x \\ &= S\left(0, a_2^{(1)}\left(R_2\right), a_3^{(1)}\left(R_3\right) + *, \ldots, a_{l/2}^{(1)}\left(R_{l/2}\right) + *,\right. \\ &\left. x_{l/2+1} + *, a_{l/2+2}^{(2)}\left(R_2\right) + x_{l/2+2} + *, \ldots, a_l^{(2)}\left(R_{l/2}\right) + x_l + *\right) \end{aligned}$$

$$\begin{aligned} y_2^{(1)} &= \gamma_1'\left(R_1\right)^{-1} y_2 = \gamma_2'\left(R_2\right) \cdots \gamma_{l/2}'\left(R_{l/2}\right) \\ &= S\left(*, *, \ldots, *, a_{l/2+2}^{(1)}\left(R_2\right) + \beta_{l/2+2}\left(R_2\right) + *, \ldots, a_l^{(1)}\left(R_{l/2}\right) + \beta_l\left(R_{l/2}\right) + *\right). \end{aligned}$$

Repeat the calculations

$$\begin{aligned} D^{(2)}(R_2, \ldots, R_{l/2}) &= t_{0(2)} \cdot y_2 t_{s(l/2)}^{-1} \\ &= S\left(0, \ldots, 0, a_{l/2+2}^{(1)}\left(R_2\right) + \beta_{l/2+2}\left(R_2\right), \ldots, a_l^{(1)}\left(R_{l/2}\right) + \beta_l\left(R_{l/2}\right)\right) \end{aligned}$$

$$\begin{aligned} D^*(R) &= D^{(2)}(R_2, \ldots, R_{l/2}) f\left(y_1^{(1)}\right) \\ &= S\left(0, \ldots, 0, \beta_{l/2+2}\left(R_2\right), a_{l/2+3}^{(1)}\left(R_3\right) + \beta_{l/2+3}\left(R_3\right) + *, \ldots\right). \end{aligned}$$

Restore R_2 with $\beta_{l/2+2}\left(R_2\right)$ using $\beta_{l/2+2}\left(R_2\right)^{-1}$.

Repeating iteratively calculating after $l/2$ steps, we obtain the recovery of $R = (R_I, R_2, \ldots, R_{l/2})$ and the message x from y_1.

Example. We will show the correctness of the obtained expressions in the following simple example. Fix the generalized Suzuki group $G = A_4(n, \theta)$ over the finite field F_q, $q = 2^{10}$. Assume that θ is the Frobenius automorphism of $F_q, \theta : \alpha \to \alpha^2$.

Let’s define $A_4(n, \theta) = \left\{ S(a_1, a_2, a_3, a_4) \mid a_i \in F_q \right\}$.

Group operation is defined as a product of two matrices

$$\begin{aligned} &S(a_1, a_2, a_3, a_4) S(b_1, b_2, b_3, b_4) = \\ &S(a_1 + b_1, a_2 + a_1{}^2 b_1 + b_2, a_3 + a_2{}^2 b_1 + a_1{}^4 b_2 + b_3, a_4 + a_3{}^2 b_1 + a_2{}^4 b_2 + a_1{}^8 b_3 + b_4). \end{aligned}$$

The inverse element is determined as

$$\begin{aligned} &S(a_1, a_2, a_3, a_4)^{-1} = S(a_1, a_2 + a_1{}^2 a_1, a_3 + a_2{}^2 a_1 + a_1{}^4 (a_2 + a_1{}^2 a_1), \\ &a_4 + a_3{}^2 a_1 + a_2{}^4 (a_2 + a_1{}^2 a_1) + a_1{}^8 (a_3 + a_2{}^2 a_1 + a_1{}^4 (a_2 + a_1{}^2 a_1))) = \\ &S(a_1, a_2 + a_1{}^3, a_3 + a_2{}^2 a_1 + a_1{}^4 a'_2, a_4 + a_3{}^2 a_1 + a_2{}^4 a'_2 + a_1{}^8 a'_3) \end{aligned}$$

where $a'_2 = a_2 + a_1{}^3$, $a'_3 = a_3 + a_2{}^2 a_1 + a_1{}^4 a'_2$.

Let`s construct a tame logarithmic signatures $\beta_k = \left[B_{1(k)}, \ldots, B_{s(k)}\right] = \left(b_{ij}\right)_k = S\left(0, .., 0, b_{ij(l/2+k)}, 0, \ldots, 0\right)$ of type $\left(r_{1(k)}, \ldots, r_{s(k)}\right)$, $i = \overline{1, s}$, $j = \overline{1, r_{i(k)}}$, $b_{ij(l/2+k)} \in F_q$, $k = \overline{1, l/2}$.

We have $l = 4$, $k = \overline{1, 2}$. Let`s define two arrays as follows

$$\beta_1 = \left[B_{1(1)},...,B_{s(1)}\right] = \left(b_{ij}\right)_1 = S\left(0,0,b_{ij(3)},0\right),$$
$$\beta_2 = \left[B_{1(2)},...,B_{s(2)}\right] = \left(b_{ij}\right)_2 = S\left(0,0,0,b_{ij(4)}\right)$$

and $b_{ij(3)}, b_{ij(4)} \in F_q$.

Logarithmic signatures β_1 and β_2 in a group representations define $b_{ij(3)}$ and $b_{ij(4)}$ coordinates. Types $\left(r_{1(k)},...,r_{s(k)}\right)$ and logarithmic signatures β_1 and β_2 are chosen independently. Let`s logarithmic signatures β_1 and β_2 have a same type $\left(r_{1(k)},...,r_{s(k)}\right) = \left(r_1,...,r_s\right)$.

As an example, $\left(r_1,...,r_s\right) = \left(2^2,2^2,2^3,2^3\right)$. Arrays $b_{ij(3)}$, $b_{ij(4)}$ consist of four subarrays with a number of rows equal to r_i . You can select any fragmentation of arrays with the condition $\prod_{i=1}^{s} r_i = q$. In our case we have $\prod_{i=1}^{s} r_i = 2^{10}$. Each row b_{ij} it`s an element of the field F_q .

The construction of arrays of logarithmic signatures is presented in [17].

First stage is to generate a tame logarithmic signature with the dimension of corresponding selected type $\left(r_{1(k)},...,r_{s(k)}\right)$ and finite field F_q .

In our case $\left(r_{1(k)},...,r_{s(k)}\right) = \left(2^2,2^2,2^3,2^3\right)$, $q = 2^{10}$ and $\beta_k = \left[B_{1(k)},B_{2(k)},B_{3(k)},B_{4(k)}\right]$. Since β_1 and β_2 have a same type $\left(r_{1(k)},...,r_{s(k)}\right) = \left(r_1,...,r_s\right)$ we will get the same β_k , $k = \overline{1,2}$ on the first stage.

Let's set a tame logarithmic signature with entries *B*i.

β(1)=	*B*1	00	00	000	000
		10	00	000	000
		01	00	000	000
		11	00	000	000
	*B*2	00	00	000	000
		00	10	000	000
		00	01	000	000
		00	11	000	000
	*B*3	00	00	000	000
		00	00	100	000
		00	00	010	000
		00	00	110	000
		00	00	001	000
		00	00	101	000
		00	00	011	000
		00	00	111	000
	*B*4	00	00	000	000
		00	00	000	100
		00	00	000	010
		00	00	000	110
		00	00	000	001
		00	00	000	101
		00	00	000	011
		00	00	000	111

Let $R = 673$. We obtain the following basis factorization for a given type $\left(r_{1(k)},...,r_{s(k)}\right) = \left(2^2, 2^2, 2^3, 2^3\right)$ in the form of $R = \left(R_1, R_2, R_3, R_4\right) = \left(1, 0, 2, 5\right)$, where $R_1 + R_2 2^2 + R_3 2^4 + R_4 2^7 = 673$.

Let`s calculate the vector of the logarithmic signature

$$\beta(R) = B_1\left(R_1\right) + B_2\left(R_2\right) + B_3\left(R_3\right) + B_{4(k)}\left(R_4\right) =$$
$$1000000000 + 0000000000 + 0000010000 +$$
$$0000000101 = 1000010101$$

To calculate $\beta(R)^{-1}$ it is enough to select groups of bits in the vector according to the type

$$\beta(R)^{-1} = 10|00|010|101 = \left(R_1, R_2, R_3, R_4\right) = \left(1, 0, 2, 5\right)$$

and recover R.

Bits positions in a logarithmic signature vector $\beta(R)$ are uniquely associated with subarrays $B_{i(k)}$ and bit values at these positions with the row number of the subarray $B_{i(k)}$. To increase the security of arrays β_k various cryptographic transformations can be used. For example, simple ones like adding noise vectors, permutations of strings in subarrays B_i, merge of arrays B_i, their permutation, matrix transformations.

In our example, we use noising of β(1) and merge of arrays B_i. This allows to construct two different logarithmic signatures β_1 and β_2. Perform random noising of β(1) in accordance with the above mentioned rule

β(2)=	B_1	00	00	000	000
		10	00	000	000
		01	00	000	000
		11	00	000	000
	B_2	**11**	00	000	000
		10	10	000	000
		10	01	000	000
		01	11	000	000
	B_3	**10**	**10**	000	000
		01	**11**	100	000
		01	**00**	010	000
		00	**10**	110	000
		11	**01**	001	000
		10	**01**	101	000
		01	**11**	011	000
		00	**10**	111	000
	B_4	**01**	**00**	**110**	000
		00	**11**	**010**	100
		11	**00**	**011**	010
		10	**01**	**000**	110
		01	**10**	**101**	001
		01	**10**	**010**	101
		00	**11**	**100**	011
		11	**01**	**001**	111

Here the noise bits are in bold. For $R=(1,0,2,5)$ we get the logarithmic signature vector

$$\beta(R)=B_1(R_1)+B_2(R_2)+B_3(R_3)+B_4(R_4)=$$
$$1000000000+1100000000+0100010000+0110010101=0110000101$$

Let`s define a highest group of bits in the vector for the calculation of $\beta(R)^{-1}$ in accordance with type $\beta(R)^{-1}=0110\,000|101\rightarrow(*,*,*,5)$ and by the combination of 101 we will recover R_4. Select the sixth row from the array B_4 and deduct it from the original vector

$$\beta(R)^{-1}=0110000101+0100010000=0010|010|101\rightarrow(*,*,2,5).$$

Repeat this procedure until the last subarray of the logarithmic signature and restore $(R_1,R_2,R_3,R_4)=(1,0,2,5)$. Apply the merge procedure for the subarrays B_1,B_2,B_3,B_4 within a rule of $C_1=(B_1,B_4)$, $C_2=B_2$, $C_3=B_3$. The first logarithmic signature β_1 has the type $(r_{1(1)},...,r_{s(1)})=(2^5,2^2,2^3)$ and imagine these permutations on the next map $\mu_1=\begin{pmatrix}1&2&3&4\\4&1&2&3\end{pmatrix}$. In the field representation β_1 has the following form

β_1=	$B_{1(1)}$	{25,103,380,332,666,173,820,814,385,525,195, 261,787,364,151,830,81,908,528,765,612,47, 588,442,258,705,330,846,751,397,989,136}
	$B_{2(1)}$	{77,154,10,957}
	$B_{3(1)}$	{154,232,309,620,849,654,578,404}

The numerical values of arrays determine the degree indicators of the generating element of the field $\gamma=\alpha^i$, whose binary representation are strings β_1. Representation of $\underline{0}$ defines single element α^0 of the field F_q. Similarly, we construct the second logarithmic signature β_2. We represent these permutations by a mapping of the form $\mu_2=\begin{pmatrix}1&2&3&4\\1&4&2&3\end{pmatrix}$. We got a new type $(r_{1(2)},...,r_{s(2)})=(2^2,2^5,2^3)$ of a logarithmic signature β_2. β_2 has the following form in the field representation

β_2=	$B_{1(2)}$	{00,0,1,77}
	$B_{2(2)}$	{258,705,330,846,751,397,989,136,577,45, 1009,592,704,921,941,187,680,667,643,84, 295,869,978,903,233,214,468,548,852,465, 333,475}
	$B_{3(2)}$	{154,232,309,620,849,654,578,404}

Arrays of logarithmic signatures β_1 and β_2 in the group representation, defines the coordinates $b_{ij(3)}$ and $b_{ij(4)}$, respectively

$$\beta_1=\left[B_{1(1)},...,B_{s(1)}\right]=\left(b_{ij}\right)_1=S\left(0,0,b_{ij(3)},0\right),$$
$$\beta_2=\left[B_{1(2)},...,B_{s(2)}\right]=\left(b_{ij}\right)_2=S\left(0,0,0,b_{ij(4)}\right).$$

Construct random covers α_k, for the same type as β_1 и β_2

$$\alpha_1 = \left[A_{1(1)}, ..., A_{s(1)} \right] = \left(a_{ij} \right)_1 = S\left(a^{(1)}_{ij(1)\cdot}, 0, a^{(1)}_{ij(3)}, 0 \right)$$

$$\alpha_2 = \left[A_{1(2)}, ..., A_{s(2)} \right] = \left(a_{ij} \right)_2 = S\left(0, a^{(2)}_{ij(2)\cdot}, 0, a^{(2)}_{ij(4)} \right),$$

where $a_{ij} \in A_l(n,\theta)$, $a^{(1)}_{ij(k)}, a^{(2)}_{ij(k+2)} \in F_q \setminus \{0\}$, $i = \overline{1,s}$, $j = \overline{1,r_{i(k)}}$, $k = \overline{1,2}$.

Each cover α_k defined by only two arrays $\left(a^{(1)}_{ij(k)}, a^{(2)}_{ij(k+2)} \right)$ with non-zero entries.

Let`s generate random covers α_1, α_2.

α_1=	$A_{1(1)}$	{(15,40),(3,400),(215,82),(990,633), (1017,597),(212,67),(788,14),(101,35), (876,505),(12,15),(21,44),(72,590), (4,319),(161,431),(41,30),(181,1008), (87,938),(427,713),(112,37),(611,147), (8,42),(188,652),(98,744),(366,96), (251,388),(349,726),(170,833),(110,897), (429,616),(331,647),(298,801),(221,529)}
	$A_{2(1)}$	{(717,101),(21,95),(977,1344),(170,195)}
	$A_{3(1)}$	{(454,865),(335,550),(881,677),(458,704), (644,233),(689,439),(329,934),(188,457)}

and

α_2=	$A_{1(2)}$	{(111,1010),(51,349),(756,953),(337,527)}
	$A_{2(2)}$	{(210,243),(309,833),(221,133),(889,126), (535,129),(909,565),(728,441),(572,375), (15,637),(71,228),(215,9),(108,553), (473,240),(348,570),(369,731),(213,416), (648,799),(451,606),(713,587),(909,436), (520,314),(19,52),(564,871),(11,531), (449,277),(392,688),(265,1002),(78,93), (223,924),(558,748),(372,975),(513,185)}
	$A_{3(2)}$	{(204,882),(69,633),(779,354),(988,754), (2277,553),(419,894),(291,632),(551,15)}

Choose random $t_{0(k)}, t_{1(k)}, ..., t_{s(k)} \in A_l(n,\theta) \setminus Z$, $s = 3$, $l = 4$, $k = \overline{1,2}$ and $t_{3(1)} = t_{0(2)}$.

Let for the first logarithmic signature we have

$t_{0(1)}$=(117,960,531,471)	$t^{-1}_{0(1)}$=(117,541,917,223)
$t_{1(1)}$=(332,801,14,522)	$t^{-1}_{1(1)}$=(332,158,398,295)
$t_{2(1)}$=(158,432,254,173)	$t^{-1}_{2(1)}$=(158,19,206,894)
$t_{3(1)}$=(1003,389,195,56)	$t^{-1}_{0(1)}$=(1003,329,199,962)

and for the second logarithmic signature β_2

$t_{0(2)}$=(1003,389,195,56)	$t^{-1}_{0(2)}$=(1003,329,199,962)
$t_{1(2)}$=(448,674,345,179)	$t^{-1}_{1(2)}$=(448,689,655,108)
$t_{2(2)}$=(221,33,309,992)	$t^{-1}_{2(2)}$=(221,15,1021,692)
$t_{3(2)}$=(649,712,760,702)	$t^{-1}_{3(2)}$=(649,23,656,14)

The next step is to calculate the arrays γ_1 и γ_2. By the condition of the example, we obtain

$$\gamma_1 = \left[h_{1(1)},...,h_{3(1)}\right] = \left(h_{ij}\right)_1 = t_{(i-1)(1)}^{-1} f\left(\left(a_{ij}\right)_1\right)\left(b_{ij}\right)_1 t_{i(1)}$$

$$\gamma_2 = \left[h_{1(2)},...,h_{3(2)}\right] = \left(h_{ij}\right)_2 = t_{(i-1)(2)}^{-1} f\left(\left(a_{ij}\right)_2\right)\left(b_{ij}\right)_2 t_{i(2)}$$

Construct a homomorphism f defined by $f\left(S(a_1,a_2,a_3,a_4)\right) = S(0,0,a_1,a_2)$.

For example, let $R_1 = \left(R_{1(1)}, R_{2(1)}, R_{3(1)}\right) = \left(1,1,5\right) = 673$ and

$$\gamma_1\left(673\right) = h_{1(1)}\left(1\right)h_{2(1)}\left(1\right)h_{3(1)}\left(5\right) = S\left(516,578,850,374\right).$$

Let $R_2 = \left(R_{1(2)}, R_{2(2)}, R_{3(2)}\right) = \left(2,6,2\right) = 354$. Compute γ_2

$$\gamma_2\left(354\right) = h_{1(2)}\left(2\right)h_{2(2)}\left(6\right)h_{3(2)}\left(2\right) = S\left(487,227,651,318\right).$$

Encryption

Input: a message $x \in G_3$, $x = S\left(0,0,x_3,x_4\right)$, and the public key $\left[f,(\alpha_k,\gamma_k)\right]$, $k = \overline{1,2}$.

Output: a ciphertext $\left(y_1,y_2\right)$ of the message x.

Let $x = S\left(0,0,299,824\right)$.

Choose a random $R = (R_1,R_2)$, $R_1, R_2, \in \square_{|F_q|}$.

Let $R_1 = 673$ и , $R_2 = 354$.

Compute

$$y_1 = \alpha'\left(R\right)\cdot x = \alpha_1'\left(R_1\right)\cdot\alpha_2'\left(R_2\right)\cdot x = S\left(139,814,393,699\right)$$

$$y_2 = \gamma'\left(R\right) = \gamma_1'\left(R_1\right)\cdot\gamma_2'\left(R_2\right) = S\left(30,766,734,871\right).$$

Output $y_1 = \left(139,814,393,699\right)$, $y_2 = \left(30,766,734,871\right)$.

Decryption

Input: a ciphertext $\left(y_1,y_2\right)$ and private key $\left[\beta_k,\left(t_{0(k)},...,t_{s(k)}\right)\right]$, $k = \overline{1,2}$.

Output: the message $x \in G_3$ corresponding to ciphertext $\left(y_1,y_2\right)$.

To decrypt a message x, we need to restore random numbers $R = (R_1,R_2)$.

Compute

$$D^{(1)}(R_1,R_2) = t_{0(1)}y_2t_{s(2)}^{-1} = t_{0(1)}S\left(30,766,734,871\right)t_{s(2)}^{-1} = S\left(0,0,459,233\right).$$

$$D^*(R) = D^{(1)}(R_1,R_2,...,R_{l/2})f\left(y_1\right) = S\left(0,...,0,\beta_{l/2+1}\left(R_1\right),a_{l/2+2}^{(1)}\left(R_2\right)+\beta_{l/2+2}\left(R_2\right)+*,...\right)$$

$$S\left(0,0,459,233\right)S\left(0,0,139,814\right) = S\left(0,0,235,950\right).$$

We get $\beta_1\left(R_1\right) = \alpha^{235} = (0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0)$.

Recovery of R_1 was done earlier $R = \left(R_1,R_2,R_3\right) = \left(1,1,5\right)$.

For further calculations, it is necessary to remove the components of the arrays $\alpha_1'\left(R_1\right)$ and $\gamma_1'\left(R_1\right)$ from ciphertext $\left(y_1,y_2\right)$.

Compute:

$$y_2^{(1)} = \gamma_1'\left(R_1\right)^{-1} y_2 = S\left(516,578,850,374\right)^{-1} S\left(30,766,734,871\right) =$$
$$S\left(516,97,108,579\right) S\left(30,766,734,871\right) =$$
$$S\left(487,227,651,318\right)$$

Repeat the calculations

$$D^{(2)}(R_2) = t_{0(2)} y_2^{(1)} t_{s(2)}^{-1} = t_{0(2)} S\left(487,227,651,318\right) t_{s(l/2)}^{-1} = S\left(0,0,0,233\right)$$
$$y_1^{(1)} = \alpha_1'\left(R_1\right)^{-1} \cdot y_1 = S\left(139,27,959,977\right)^{-1} S\left(139,814,393,699\right) =$$
$$S\left(139,787,0,148\right) S\left(139,814,393,699\right) = S\left(0,693,299,784\right)$$
$$D^{*}(R) = D^{(2)}(R_2) f\left(y_1^{(1)}\right) = S\left(0,0,0,233\right) f\left(S\left(0,693,299,784\right)\right) =$$
$$S\left(0,0,0,233\right) S\left(0,0,0,693\right) = S\left(0,0.0,444\right)$$

Restore R_2 with $\beta_2\left(R_2\right) = \alpha^{444} = \left(1\,1\,1\,1\,1\,1\,0\;0\,1\,1\right)$.

Perform inverse calculations $\beta_2\left(R_2\right)^{-1}$. Select bit groups in vector $\beta\left(R\right)$ according to type $\left(r_{1(2)},...,r_{s(k2}\right) = \left(2^2,2^2,2^3,2^3\right)$. We use the same calculations as in the example for $\beta_1\left(R_1\right)^{-1}$, and we get

11|11|110|**011**	R_2=(*,*,*,6)
00|11|100|011	row from β(2)
11|00|**010**|000	R_2= (*,*,2,6)
01|00|010|000	row from β(2)
10|**00**|000|000	R_2= (*,0,2,6)
11|00|000|000	row from β(2)
01|00|000|000	R_2= (2,0,2,6)

$$\beta_2\left(R'\right)^{-1} = 11|00|010|011 = \left(R_1',R_2',R_3',R_4'\right) = \left(2,0,2,6\right).$$

The resulting vector along with the concatenation of array entries $B_{2(2)}, B_{4(2)}$ due to their merging, we write in the following bit representation

$$\beta_2\left(R'\right)^{-1} := \mu_2\left\{\left(2,0,2,6\right)\right\} \rightarrow \left(2,6\|0,2\right) = \left(01,01100,010\right).$$

The transition from bit to numeric gives the desired value $R_2 = \left(R_{1(2)},R_{2(2)},R_{3(2)}\right) = \left(2,6,2\right)$. Receive a message

$$x = \alpha'\left(R\right)^{-1} y_1 = \alpha_2'\left(R_2\right)^{-1} \cdot \alpha_1'\left(R_1\right)^{-1} \cdot y_1$$
$$= S\left(0,693,0,418\right) S\left(139,787,0,148\right) S\left(139,814,393,699\right) = S\left(0,0,299,824\right).$$

Output: the message $x = \left(0,0,299,824\right)$.

Security Analysis. An attack on the cryptosystem is possible by solving the $\beta(R_1,R_2,...,R_{l/2}) = t_{0(1)} \cdot y_2 t_{s(l/2)}^{-1} f\left(y_1\right)$ equation by selecting $t_{0(k)}$ and $t_{s(k)}$ vectors. There are $l/2$ such vectors.

The success of an attack is determined by the selection of $t_{0(k)}$ and $t_{s(k)}$ vectors and has complexity proportional to the $l/2$ power of the group $\left|A_l(n,\theta)\right|$ due to the non-commutativity of the generalized Suzuki group. Brute force attack is possible by selection $R_1, R_2, ..., R_{l/2l} \in \square_{|F_q|}$. The complexity is determined by the value $l/2$, finite field power F_q in proportion to $l/2|F_q|$. Complexity analysis. Fix the generalized Suzuki group $G = A_l(n,\theta)$ and the logarithmic signature for the type $(r_1, ..., r_s)$ over the finite field F_q, $q = 2^n$. Suppose that the values r_i are approximately equal $r_i = 2^{n/s}$.

Early show that the size of the logarithmic signature has the estimate $V = ls|A_l(n,\theta)|^{1/ls}$. For $q = 2^{64}$, $|A_l(n,\theta)| = 2^{512}$, $l = 8$ and $s = 2^3, 2^4, 2^5$ we obtain, respectively, $V = 2^{14}, 2^{11}, 2^{10}$ of the 64 bit strings.

5 Conclusions

The proposed design of the MST3 cryptosystem based on the generalized Suzuki group provides potentially higher privacy and optimizes the cost of key data. The difference from the well-known MST3 construction is the iterative key recovery from calculations in the large normal subgroup of the generalized Suzuki group.

References

1. N.R. Wagner and M.R. Magyarik, “A public-key cryptosystem based on the word problem”, Proc. Advances in Cryptology – CRYPTO 1984, LNCS 196, Springer-Verlag (1985), pp. 19–36.
2. 2K.H. Ko, S.J. Lee, J.H .Cheon, J.W .Han, J. Kang, and C. Park, “New public-key cryptosystem using braid groups”, in Advances in cryptology—CRYPTO 2000 ,vol.1880of Lecture Notes in Computer Science , pp. 166–183, Springer, Berlin, Germany, 2000.
3. B. Eick and D. Kahrobaei, “Polycyclic groups: a new platform for cryptology”, http://arxiv.org/abs/math/0411077.
4. V. Shpilrain and A. Ushakov, “Thompsons group and public key cryptography”, in Applied Cryptography and Network Security, vol. 3531 of Lecture Notes in Computer Science, pp. 151–164, 2005.
5. 5D. Kahrobaei, C. Koupparis, and V. Shpilrain, “Public key exchange using matrices over group rings”, Groups, Complexity, and Cryptology ,vol.5,no.1,pp.97–115,2013.
6. S.S. Magliveras, “A cryptosystem from logarithmic signatures of finite groups”, in Proceedings of the 29th Midwest Symposium on Circuits and Systems , pp. 972–975, Elsevier Publishing, Amsterdam, The Netherlands, 1986.
7. W. Lempken, S.S. Magliveras, Tran van Trung and W. Wei, “A public key cryptosystem based on non-abelian finite groups”, J. of Cryptology, 22 (2009), 62–74.
8. S.S.Magliveras, D.R.Stinson, and T.vanTrung, “New approaches to designing public key cryptosystems using one-way functions and trapdoors in finite groups”, Journal of Cryptology , vol. 15, no. 4, pp. 285–297, 2002.
9. S.S. Magliveras, P. Svaba, T. Van Trung, and P. Zajac, “On the security of a realization of cryptosystem MST3”, Tatra Mountains Mathematical Publications ,vol.41,pp.65–78,2008.
10. H.Stichtenoth, “Über die Automorphismengruppe eines algebraischen Funktionenkörpers von Primzahlcharakteristik” I, II, Arch. Math. 24, pp.524–544 and pp.615–631, 1973.

11. A. Garcia, H. Stichtenoth, C.-P.Xing, “On Subfields of the Hermitian Function Field”, Kluwer Academic Publishers, Compositio Mathematica 120: pp.137–170, 2000.
12. W. Lempken and T. van Trung, “On minimal logarithmic signatures of finite groups”, Experimental Mathematics,vol.14, no. 3, pp. 257–269, 2005.
13. H.Hong, J.Li, L.Wang, Y. Yang, X.Niu “A Digital Signature Scheme Based on MST3 Cryptosystems” Hindawi Publishing Corporation, Mathematical Problems in Engineering ,vol 2014, 11 pages, http://dx.doi.org/10.1155/2014/630421
14. G.Khalimov, Y. Kotukh, S.Khalimova “MST3 cryptosystem based on the automorphism group of the hermitian function field”, 2019 2019 IEEE International Scientific-Practical Conference: Problems of Infocommunications Science and Technology, PIC S and T 2019 – Proceedings
15. G. Khalimov, T.T.Simon, A.M.Adrees, E.Kotukh “MST3 cryptosystem based on a small Ree groups” 3rd IEEE international conference advanced information and communication technologies-2019 “Next-generation networking for the Internet of Things: 5g, sdn, nfv and cloud computing” 2~6 july, 2019 ,Lviv, Ukraine
16. A.Hanaki “A CONDITION ON LENGTHS OF CONJUGACY CLASSES AND CHARACTER DEGREES” Osaka J. Math. 33 pp.207-216, 1996
17. P. Svaba, “Covers and logarithmic signatures of finite groups in cryptography”, Dissertation, https://bit.ly/2Ws2D24