9. *Bernstein D. J., Chou T., and Schwabe P.* McBits: Fast constant-time code-based cryptography. LNCS, 2013, vol. 8086, pp. 250–272.

10. *Barreto A. and Misoczki R.* A New One-Time Signature Scheme from Syndrome Decoding. IACR Cryptology ePrint Archive, 2010.

11. *Nojima R., Imai H., Kobara K., et al.* Semantic security for the McEliece cryptosystem without random oracles. Designs, Codes, Cryptogr., 2008, vol. 49, pp. 289–305.

12. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. `https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf`.

UDC 003.26 DOI 10.17223/2226308X/14/25

AN IMPROVEMENT OF CRYPTOGRAPHIC SCHEMES BASED ON THE CONJUGACY SEARCH PROBLEM[1]

V. A. Roman'kov

The key exchange protocol is a method of securely sharing cryptographic keys over a public channel. It is considered as important part of cryptographic mechanism to protect secure communications between two parties. The Diffie — Hellman protocol, based on the discrete logarithm problem, which is generally difficult to solve, is the most well-known key exchange protocol. One of the possible generalizations of the discrete logarithm problem to arbitrary noncommutative groups is the so-called conjugacy search problem: given two elements g, h of a group G and the information that $g^x = h$ for some $x \in G$, find at least one particular element x like that. Here g^x stands for $x^{-1}gx$. This problem is in the core of several known public key exchange protocols, most notably the one due to Anshel et al. and the other due to Ko et al. In recent years, effective algebraic cryptanalysis methods have been developed that have shown the vulnerability of protocols of this type. The main purpose of this short note is to describe a new tool to improve protocols based on the conjugacy search problem. This tool has been introduced by the author in some recent papers. It is based on a new mathematical concept of a marginal set.

Keywords: *cryptography, key exchange protocol, conjugacy search problem, marginal set, algorithm.*

1. Introduction

The first detailed proposal for a key exchange protocol, due to Diffie and Hellman [1], was based on the discrete logarithm problem for a finite field. This protocol is one of the earliest practical examples of public key exchange implemented within the field of cryptography. It was followed by few alternative proposals for key exchange protocols, all based on commutative algebraic structures.

Noncommutative cryptography is the area of cryptology where the cryptographic primitives, methods, and systems are based on algebraic structures like semigroups, groups and rings which are noncommutative. One of the earliest applications of a noncommutative algebraic structure for cryptographic purposes was the use of braid groups to develop the Commutator key exchange protocol by Anshel, Anshel and Goldfeld (AAG) [2] and the noncommutative key exchange protocol on braids by Ko et al. [3]. Later, several other noncommutative structures like nilpotent and polycyclic groups, and matrix groups have been identified as potential candidates for cryptographic applications.

[1] The research was supported by a grant from the Russian Science Foundation (project no. 19-71-10017).

In [4], the author introduced the method of *linear decomposition* applicable in algebraic cryptanalysis. In [5], this method was further developed by the author and A. G. Myasnikov, see also [6]. In [7], this method was supplemented by the *nonlinear decomposition* method. These applications are called *linear* and *nonlinear decomposition attacks* respectively. They are deterministic, provable and polynomial-time. These methods were widely applied in cryptanalysis of dozens of protocols of algebraic cryptography, see monograph [8] and references therein. The linear decomposition attack can be applied to protocols based on matrix groups over arbitrary (finite or infinite) fields. The nonlinear decomposition attack is applicable to protocols based on groups that are not necessary matrix, or do not use matrix representations. The main distinguishing feature of these methods is that they reveal secret exchanged keys from open data without calculating the secret parameters used in the algorithm. Thus, we show that in this case, contrary to the common opinion, the typical computational security assumptions are not very relevant to the security of the schemes, i.e., one can break the schemes without solving the algorithmic problems on which the assumptions are based.

In [9] (see also [10]), B. Tsaban et al. introduced a method for obtaining provable polynomial-time solutions of problems in noncommutative algebraic cryptography called the *linear span-method*, or simply the *span-method*. This method is probabilistic. This method is a fundamental base for algebraic span cryptanalysis, a general approach for provable polynomial-time solutions of computational problems in groups of matrices over finite fields, and thus in all groups with efficient matrix representations over finite fields. This approach is widely applicable, in particular, it is applicable to the protocols mentioned above.

The main aim of this note is to describe the idea of using the concept of marginal sets to enhance the protocols based on the conjugacy search problem. In [11], the author presented an improved version of the AAG protocol based on this idea, see also [12] with some versions of AAG and Ko et al. protocols. In [13], the author proposed a new more strong version of the Diffie — Hellman non-commutative key exchange protocol of Ko et al. These new versions are resistant against attacks by methods of linear algebra. They are based on new hard algorithmic problems using a notion of a marginal set. In particular, they are resistant against attacks by the methods of Tsaban, and against the authors methods of the linear and nonlinear decompositions.

Notations: $\mathbb{N}$ — the set of nonnegative integers, $\mathbb{S}_n$ — symmetric group of degree n, $g^h = hgh^{-1}$ — conjugate, $\mathbb{F}_q$ — field of order q, $\mathrm{M}(n, \mathbb{F}_q)$ is the algebra of $n \times n$ matrices over $\mathbb{F}_q$.

2. The marginal sets

The introducing concept of marginal set formally generalizes the well-known concept of the marginal subgroup, but it is worth noting that this generalization is very different from the original concepts.

The marginal subgroup is determined by the word, but the marginal subset is determined by the word and its chosen value. The set of all marginal subsets is not closed under algebra- and group-theoretic operations. It can be very wild.

For brevity, we give definitions only for the case of algebra.

Let F be a free associative algebra with unity on a countably infinite set $\{x_1, x_2, \ldots\}$ and let $w = w(x_1, \ldots, x_k) \in F$. If $g_1, \ldots, g_k$ are elements of the algebra M, we define the *value* of the word w at $\bar{g} = (g_1, \ldots, g_k)$ to be $w(\bar{g}) = w(g_1, \ldots, g_k)$.

A subset $N \subseteq M$ is said to be *w-marginal* in M if

$$w(g_1, \ldots, g_k) = w(u_1 g_1, \ldots, u_k g_k)$$

for all $g_i \in G$, $u_i \in N$, $1 \leqslant i \leqslant n$. Obviously, all w-marginal subsets constitutes the maximal marginal subset $w^*(M)$, which is a submonoid in M^k.

We introduce a new concept that significantly extends the marginality property.

Definition 1. For $k \in \mathbb{N}$, let $w = w(x_1, \ldots, x_k)$ be an algebra word, M be an algebra and $\bar{g} = (g_1, \ldots, g_k)$ be a tuple of elements of M. We say that a tuple $\bar{c} = (c_1, \ldots, c_k) \in M^k$ is a *marginal tuple* determined by w and $\bar{g}$ if

$$w(c_1g_1, \ldots, c_kg_k) = w(g_1, \ldots, g_k).$$

We will write $\bar{c} \perp w(\bar{g})$ in this case. A set $\bar{C} \subseteq M^k$ is said to be *marginal* with respect to w and $\bar{g}$ if $\bar{c} \perp w(\bar{g})$ for every tuple $\bar{c} \in \bar{C}$. We write $\bar{C} \perp w(\bar{g})$ in this case.

Now we give a very simple and efficient algorithm for constructing a marginal set $\bar{C} \perp w(g_1, \ldots, g_k)$. This method does not depend on the structure of M.

Let $w(g_1, \ldots, g_k) \in M$ be any value of $w(x_1, \ldots, x_k)$. Note that some elements g_i, g_j can be equal to each other, that is, $g_i = g_j$. Consider an equation of the form

$$w(z_1g_1, \ldots, z_kg_k) = w(g_1, \ldots, g_k) \tag{1}$$

such that there is z_i that can be expressed in the form

$$z_i = z_i(z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_k, g_1, \ldots, g_k). \tag{2}$$

Then for any substitution $z_j = f_j$, $f_j \in M$, $j = 1, \ldots, i-1, i+1, \ldots, k$, we get a new marginal tuple

$$(f_1, \ldots, f_{i-1}, z_i(f_1, \ldots, f_{i-1}, f_{i+1}, \ldots, f_k, g_1, \ldots, g_k), f_{i+1}, \ldots, f_k) \in M^k \tag{3}$$

with respect to w and $\bar{g}$.

To hide the word w in (1) and elements $f_1, \ldots, f_k, g_1, \ldots, g_k$, (2) can be rewritten by expressing all the constituent elements through parameters and the generating elements $m_1, \ldots, m_s$ of the algebra M. The formula (2) can be changed as follows. Let us introduce into consideration the set of parameters $y_1, \ldots, y_q$ with arbitrary values in M. Let $z_j = z_j(y_1, \ldots, y_q, m_1, \ldots, m_s)$ be an arbitrary presentation for $j = 1, \ldots, i-1, i+1, \ldots, k$. Then $z_i = z_i'(y_1, \ldots, y_q, m_1, \ldots, m_k)$ be the rewritten presentation (2) of z_i. These parametric presentations can be published. This form of representation does not make it easy to recover the word w in (1).

Every solution of (1) can be included in a marginal set $\bar{C}$, $\bar{C} \perp w(\bar{g})$. We also can multiply a marginal tuple $\bar{c} = (c_1, \ldots, c_k)$ to any tuple $\bar{u} = (u_1, \ldots, u_k) \in w^*(M)^k$, and get new marginal tuple $\bar{c}\bar{u} = (u_1c_1, \ldots, u_kc_k)$.

3. An improved version of the conjugacy search problem

Recall the classical definition.

Definition 2. *Conjugacy Search Problem* (CSP). For a group G, we are asked to find an element x from $u, v \in G$ satisfying $v = u^x \in G$.

The version suggested below uses any private expression of the element g in the form of a word. Such view allows the use of a marginal set for given expression, defined below. It also becomes possible to apply multipliers that are not changed by the used transformation (conjugation). These methods protect the protocol from the attacks by methods of linear algebra. They change the underlying problem to a much more complex one. Let's move on to a description of the proposed changes. They are partially presented in [11–13].

Assumptions. Let $\mathbb{F}$ be an arbitrary field (in particular $\mathbb{F}_q$). Let $G \leqslant \mathrm{M}(n, \mathbb{F})$ be a matrix group and B be a finitely generated subgroup of G. Fix an element $g \in M = \mathrm{Alg}(G)$ (the algebra generated by G in $\mathrm{M}(n, \mathbb{F})$). We assume that all the data above is public. We set $\mathrm{Fix}(B) = \{o \in G : o^b = o \text{ for all } b \in B\}$.

Algorithm. Data selection and transmission.

Firstly we describe Alice's action:

- Alice chooses a tuple $\bar{g} = (g_1, \ldots, g_k) \in M^k$ and a ring word $u = u(x_1, \ldots, x_k)$ such that $g = u(g_1, \ldots, g_k)$. This data is private.
- Alice takes arbitrary private elements $g_{k+1}, \ldots, g_m \in M$ (these elements are called *virtual*) to obtain $\tilde{g} = (g_1, \ldots, g_k, g_{k+1}, \ldots, g_m) \in M^m$. She also chooses a private tuple of elements $\bar{h} = (h_1, \ldots, h_k) \in \mathrm{Fix}(B)^k$ and adds this tuple by random private elements $h_{k+1}, \ldots, h_m$ of M to get $\tilde{h} = (h_1, \ldots, h_k, h_{k+1}, \ldots, h_m) \in M^m$. Alice gets $\tilde{gh} = \tilde{g}\tilde{h} = (g_1 h_1, \ldots, g_k h_k, g_{k+1} h_{k+1}, \ldots, g_m h_m) \in M^m$. Then she picks up a private random permutation $\pi \in \mathbb{S}_m$ and publishes the tuple

$$\tilde{gh}_\pi = (g_{\pi(1)} h_{\pi(1)}, \ldots, g_{\pi(m)} h_{\pi(m)}).$$

- Alice constructs a marginal set $C \subseteq M^k$, $C \perp u(g_1, \ldots, g_k)$, adds each $\bar{c} = (c_1, \ldots, c_k) \in$ $\in C$ by arbitrary elements $c_{k+1}, \ldots, c_m$ to get $\tilde{c} = (c_1, \ldots, c_k, c_{k+1}, \ldots, c_m)$ and publishes $C_\pi = \{\tilde{c}_\pi = (c_{\pi(1)}, \ldots, c_{\pi(m)}) : \bar{c} \in C\}$.

Bob's action is similar. Now we restrict ourselves by considering the improved version of the conjugacy search problem, not some specific protocol.

Algorithm. Data processing:

- Bob chooses a random element $b \in B$.
- Bob chooses a random tuple $c_\pi \in C_\pi$ and calculates $c_\pi(\tilde{gh})_\pi$. Then he computes

$$(c_\pi(\tilde{gh})_\pi)^b = ((c_{\pi(1)} g_{\pi(1)} h_{\pi(1)})^b, \ldots, (c_{\pi(m)} g_{\pi(p)} h_{\pi(p)})^b)$$

and sends the result to Alice.

Algorithm. The key generation. Alice's action:

- Alice uses π^{-1} to remove virtual elements and get from $(c_\pi(\tilde{gh})_\pi)^b$ the tuple

$$(\bar{c}\bar{g})^b \bar{h}.$$

- She multiplies the result to $\bar{h}^{-1} = (h_1^{-1}, \ldots, h_k^{-1})$ and gets $\bar{c}\bar{g}^b$.
- Alice computes

$$u(\bar{c}\bar{g}^b) = u(\bar{c}\bar{g})^b = u(\bar{g})^b = g^b.$$

In many protocols Alice obtains the shared key as

$$K = (g^b)^a = g^{ab},$$

where $a \in G$ is her private element commuting with b.

Cryptanalysis. One cannot directly apply known method to calculate b. Indeed, for this one need in a pair of the form r, r^b $(r \in M)$, but instead one has $r, (cr)^b$ $(c \in M)$.

Instead, one can try to find the word $u'(x_1, \ldots, x_k)$ (one can be change k), indexes $i_1, \ldots, i_k$ and elements $h_i \in \mathrm{Fix}(B)$ $(i = 1, \ldots, k)$ so that

$$u'(g_{i_1} h_{i_1} h'_1, \ldots, g_{i_k} h_{i_k} h_k) = g.$$

But even if successful, this does not guarantee that the following equality holds:

$$u'((g_{i_1} h_{i_1} h'_1)^b, \ldots, (g_{i_k} h_{i_k} h_k)^b) = g^b,$$

because the marginality of C depends of the word $u(x_1, \ldots, x_k)$ and in general is not true for another word that presents g.

REFERENCES

1. *Diffie W. and Hellman M. I.* New directions in cryptography. IEEE Trans. Inform. Theory, 1976, vol. 22, pp. 644–654.
2. *Anshel I., Anshel M., and Goldfeld D.* An algebraic method for public-key cryptography. Math. Res. Lett., 1999, vol. 6, no. 3, pp. 287–291.
3. *Ko K. H., Lee S. J., Cheon J. H., et al.* New public-key cryptosystem using braid groups. LNCS, 2000, vol. 1880, pp. 166–183.
4. *Roman'kov V. A.* Algebraicheskaya kriptografiya [Algebraic Cryptography]. Omsk, Omsk State University Publ., 2013, 136 p. (in Russian)
5. *Myasnikov A. G. and Roman'kov V. A.* A linear decomposition attack. Groups, Complex., Cryptol., 2015, vol. 7, no. 1, pp. 81–94.
6. *Roman'kov V. A.* Kriptoanalis nekotorih shem ispolzujushih avtomorfizmi [Cryptanalysis of some schemes applying automorphisms]. Prikladnaya Discretnaya Matematika, 2013, no. 3, pp. 35–51. (in Russian)
7. *Roman'kov V. A.* A nonlinear decomposition attack. Groups, Complex., Cryptol., 2016, vol. 8, no. 2, pp. 197–207.
8. *Roman'kov V. A.* Essays in Algebra and Cryptology: Algebraic Cryptanalysis. Omsk, Omsk State University Publ., 2018. 207 p.
9. *Tsaban B.* Polynomial-time solutions of computational problems in noncommutative-algebraic cryptography. J. Cryptol., 2015, vol. 28, no. 3, pp. 601–622.
10. *Ben-Zvi A., Kalka A., and Tsaban B.* Cryptanalysis via algebraic span. LNCS, 2018, vol. 10991, pp. 255–274.
11. *Roman'kov V. A.* An improved version of the AAG cryptographic protocol. Groups, Complex., Cryptol., 2019, vol. 11, no. 1, pp. 35–42.
12. *Roman'kov V, A.* Algebraic cryptanalysis and new security enhancement. Moscow J. Combinat. Number Theory, 2020, vol. 9, no. 2, pp. 123–146.
13. *Roman'kov V. A.* An improvement of the Diffie-Hellman noncommutative protocol. Designs, Codes, Cryptogr., to appear.