PRACTICAL POLYNOMIAL TIME SOLUTIONS OF SEVERAL MAJOR PROBLEMS IN NONCOMMUTATIVE-ALGEBRAIC CRYPTOGRAPHY
(preliminary announcement)

BOAZ TSABAN

Abstract. We provide new provable polynomial time solutions of a number of problems in noncommutative-algebraic cryptography. In contrast to the linear centralizer method of [2], the new method is very simple: In order to solve linear equations on matrices restricted to matrix groups, solve them over the generated algebras. We name this approach the *algebraic span method*.

The resulting algorithms are have substantially better performance than those of [2]. These algorithms constitute cryptanalyses of the motivating protocols that cannot be foiled by changing the distributions used in the protocols, and are practical for most affordable parameter settings.

1. Introduction

Since this is a preliminary, brief announcement, we refer the reader to our earlier paper [2] for history and background. We abbreviate *Key Exchange Protocol* as *KEP*. We demonstrate the method by applying it to the Commutator KEP. It is then straightforward to apply it to all other KEPs that were cryptanalyzed in [2].

2. The Commutator KEP

We will use, throughout, the following basic notation.

Notation 1. For a noncommutative group G and group elements $g, x \in G$, $g^x = x^{-1}gx$, the conjugate of g by x.

Useful identities involving this notation, that are easy to verify, include $g^{xy} = (g^x)^y$, and $g^c = g$ for every *central* element $c \in G$, that is, such that $ch = hc$ for all $h \in G$.

The *Commutator KEP* [1] is described succinctly in Figure 1.[1] In some detail:

(1) A noncommutative group G and elements $a_1, \ldots, a_k, b_1, \ldots, b_k \in G$ are publicly given.[2]
(2) Alice and Bob choose free group words in the variables $x_1, \ldots, x_k$, $v(x_1, \ldots, x_k)$ and $w(x_1, \ldots, x_k)$, respectively.[3]

Key words and phrases. noncommutative-algebraic cryptography, group theory-based cryptography, braid-based cryptography, Commutator key exchange, Centralizer key exchange, Braid Diffie–Hellman key exchange, linear cryptanalysis, invertibility lemma, Schwartz–Zippel lemma, algebraic span method, algebraic cryptanalysis.

[1] In our diagrams, green letters indicate publicly known elements, and red ones indicate secret elements, known only to the secret holders. Results of computations involving elements of both colors may be either publicly known, or secret, depending on the context. The colors are not necessary to follow the diagrams.

[2] By adding elements, if needed, we assume that the number of a_i's is equal to the number of b_i's.

[3] A free group word in the variables $x_1, \ldots, x_k$ is a product of the form $x_{i_1}^{\epsilon_1} x_{i_2}^{\epsilon_2} \cdots x_{i_m}^{\epsilon_m}$, with $i_1, \ldots, i_m \in \{1, \ldots, k\}$ and $\epsilon_1, \ldots, \epsilon_m \in \{1, -1\}$, and with no subproduct of the form $x_i x_i^{-1}$ or $x_i^{-1} x_i$.

(3) Alice substitutes $a_1, \ldots, a_k$ for $x_1, \ldots, x_k$, to obtain a secret element $a = v(a_1, \ldots, a_k) \in G$. Similarly, Bob computes $b = w(b_1, \ldots, b_k) \in G$.
(4) Alice sends the conjugated elements ${b_1}^a, \ldots, {b_k}^a$ to Bob, and Bob sends ${a_1}^b, \ldots, {a_k}^b$ to Alice.
(5) The shared key is the *commutator* $a^{-1}b^{-1}ab$.

As conjugation is a group isomorphism, we have that

$$v({a_1}^b, \ldots, {a_k}^b) = v(a_1, \ldots, a_k)^b = a^b = b^{-1}ab.$$

Thus, Alice can compute the shared key $a^{-1}b^{-1}ab$ as $a^{-1}v({a_1}^b, \ldots, {a_k}^b)$, using her secret $a, v(x_1, \ldots, x_k)$ and the public elements ${a_1}^b, \ldots, {a_k}^b$. Similarly, Bob computes $a^{-1}b^{-1}ab$ as $w({b_1}^a, \ldots, {b_k}^a)^{-1}b$.

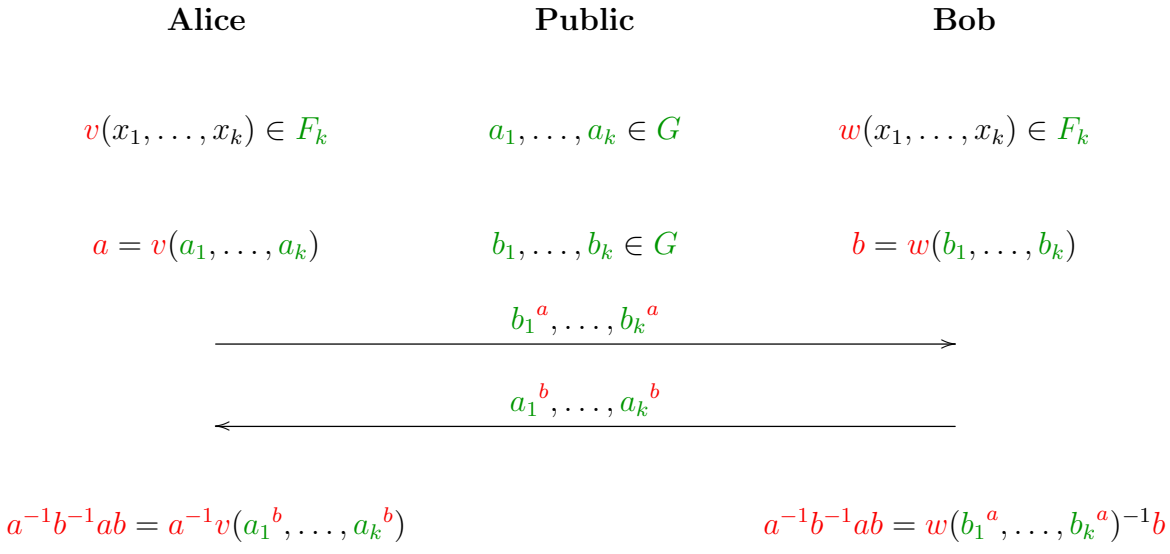

FIGURE 1. The Commutator KEP

In the passive adversary model, the security of the Commutator KEP is determined by the difficulty of the following problem. As usual, for a group G and elements $g_1, \ldots, g_k \in G$, $\langle g_1, \ldots, g_k \rangle$ denotes the subgroup of G generated by $g_1, \ldots, g_k$. Throughout, we assume that the given groups are represented in an efficient way.

Problem 2 (Commutator KEP Problem)**.** *Let G be a group. Let $a_1, \ldots, a_k, b_1, \ldots, b_k \in G$. Let $a \in \langle a_1, \ldots, a_k \rangle, b \in \langle b_1, \ldots, b_k \rangle$.*
Given $a_1, \ldots, a_k, b_1, \ldots, b_k, a_1^b, \ldots, a_k^b, b_1^a, \ldots, b_k^a$, compute $a^{-1}b^{-1}ab$.

The braid group $\mathbf{B}_N$ was proposed as a platform group for this KEP, but it was demonstrated in [2] that we may assume that the platform group is a matrix group over a finite field.

3. Algebraic spans

We solve the Commutator KEP Problem in matrix groups. Let $\mathbb{F}$ be a finite field. For a set $S \subseteq \mathrm{M}_n(\mathbb{F})$, let $\mathrm{Alg}(S)$ be the algebra generated by S, that is, the smallest Algebra $A \subseteq \mathrm{M}_n(\mathbb{F})$ that contains S as a subset. Every subalgebra of $\mathrm{M}_n(\mathbb{F})$ is also a vector space over $\mathbb{F}$. For a group $G \leq \mathrm{GL}_n(\mathbb{F})$, we have that $\mathrm{Alg}(G) = \mathrm{span}(G)$.

Proposition 3. *Let $G = \langle g_1, \ldots, g_k \rangle \leq \mathrm{GL}_n(\mathbb{F})$. A basis for the vector space $\mathrm{Alg}(G)$ can be computed in time $O(kn^6)$.*

Proof. Initialize $S = (e)$, the identity element of S. For each i from 1 to the length of S, do the following:

(1) Let s be the ith element of S.
(2) For $j = 1, \ldots, k$, if $sg_i \notin \operatorname{span} S$, then append sg_i at the end of S . If no element was appended to S in this step, terminate.

The resulting set $S \setminus \{e\}$ is a basis for $\operatorname{span} G$. □

We believe that there are much faster algorithms for Proposition 3. One possible route may be to move (using a variation of Meataxe, perhaps) to a direct sum of irreducible blocks, and then take the standard basis for each block. This requires further study.

Lemma 4. *Let $x, \tilde{x} \in \mathrm{GL}_n(\mathbb{F})$ and $G = \langle g_1, \ldots, g_k \rangle \leq \mathrm{GL}_n(\mathbb{F})$. If ${g_i}^x = {g_i}^{\tilde{x}}$ for all $i = 1, \ldots, k$, then $g^x = g^{\tilde{x}}$ for all $g \in \mathrm{Alg}(G)$.*

Proof. Conjugation is an automorphism of the matrix algebra. □

We are ready to present our solution of the Commutator KEP problem. There are two options for our algorithm. The first one divides between offline and online phases, and the second is all online.

Input: $a_1, \ldots, a_k, b_1, \ldots, b_k, a_1^b, \ldots, a_k^b, b_1^a, \ldots, b_k^a \in G$, where $a \in \langle a_1, \ldots, a_k \rangle, b \in \langle b_1, \ldots, b_k \rangle$ are unknown.

3.1. **Offline–Online version.**

(1) *Offline:* Generate bases for $\mathrm{Alg}(A)$ and $\mathrm{Alg}(B)$. Let d be the maximum of the sizes of these bases.
(2) *Online:*
   (a) Solve the following homogeneous system of linear equations in the unknown matrix $x \in \mathrm{Alg}(A)$:
$$\begin{aligned} b_1 \cdot x &= x \cdot {b_1}^a \\ &\vdots \\ b_k \cdot x &= x \cdot {b_k}^a, \end{aligned}$$
   a system of linear equations on the d coefficients determining x.
   (b) Fix a basis for the solution space, and pick random solutions x until x is invertible.
   (c) Solve the following homogeneous system of linear equations in the unknown matrix $y \in \mathrm{Alg}(B)$:
$$\begin{aligned} a_1 \cdot y &= y \cdot {a_1}^b \\ &\vdots \\ a_k \cdot y &= y \cdot {a_k}^b, \end{aligned}$$
   a system of linear equations on the d coefficients determining y.
   (d) Fix a basis for the solution space, and pick random solutions y until y is invertible.
   (e) *Output:* $x^{-1}y^{-1}xy$.

That the algorithm terminates follows from the Invertibility Lemma [2]. We show that the output is correct.

As $y \in \mathrm{Alg}(B)$, we have by Lemma 4 that $y^x = y^a$, and therefore

$$(y^{-1})^x = (y^x)^{-1} = (y^a)^{-1} = (y^{-1})^a.$$

It follows that

$$x^{-1}y^{-1}xy = (y^{-1})^x y = (y^{-1})^a y = a^{-1}y^{-1}ay = a^{-1}a^y.$$

As $a \in \mathrm{Alg}(A)$, we have by Lemma 4 that $a^y = a^b$, and thus

$$x^{-1}y^{-1}xy = a^{-1}a^b = a^{-1}b^{-1}ab.$$

3.2. **Online only version.**

(1) Generate a basis for $\mathrm{Alg}(A)$. Together with each element g of this basis, store also g^b.[4] Let $d = \dim \mathrm{Alg}(A)$.
(2) Solve the following homogeneous system of linear equations in the unknown matrix $x \in \mathrm{Alg}(A)$:

$$\begin{aligned} b_1 \cdot x &= x \cdot {b_1}^a \\ &\vdots \\ b_k \cdot x &= x \cdot {b_k}^a, \end{aligned}$$

a system of linear equations on the d coefficients determining x.
(3) Fix a basis for the solution space, and pick random solutions x until x is invertible.
(4) Compute x^b, using the representation of x as a linear combination of basis elements whose b-conjugates are known.
(5) *Output:* $x^{-1}x^b$.

Correctness:

$$x^{-1}x^b = x^{-1}b^{-1}xb = (b^x)^{-1}b = (b^a)^{-1}b = a^{-1}b^{-1}ab.$$

The complexity of the step with linear equations is kd^ω, which is at most $kn^{2\omega}$.

The approach also applies to some other schemes, including the Centralizer KEP, the Braid Diffie–Hellman KEP (and, more generally, the Double Coset KEP). It *does not* seem to apply to the Triple Decomposition KEP. Descriptions of all mentioned KEPs are available in [2].

4. Comments

The overall complexity of the first algorithm presented here, in field operations, is kn^6 offline and $kn^{2\omega}$ online. We have mentioned briefly why we expect that the offline complexity can be improved so that it becomes at most $kn^{2\omega}$. The complexity of the linear centralizer algorithm [2] for the same problem was $kn^{2\omega+2}$.

Consider the Commutator KEP in the braid group $\mathbf{B}_N$. To make it comparable to RSA, for example, in terms of space and time complexities, the braid index N should not be larger than 64. The reduction in [2] to matrix groups embeds the corresponding problem into $\mathrm{GL}_n(\mathbb{F})$ with $n = \binom{N}{2}$. The number of field operations thus becomes about $k2^{67}$, which is feasible. The complexity of the field operations can be kept affordable by Chinese remaindering (to see how, one has to consider the actual reduction, available in [2]). We believe that a careful analysis would render this approach applicable.

Acknowledgements. We thank Craig Gentry for intriguing discussions.

[4]Note that this can be done, using solely Bob's "public key".

References

[1] I. Anshel, M. Anshel, D. Goldfeld, *An algebraic method for public-key cryptography*, Mathematical Research Letters **6** (1999), 287–291.

[2] B. Tsaban, *Polynomial time solutions of computational problems in noncommutative-algebraic cryptography*, Journal of Cryptology, to appear. DOI: 10.1007/s00145-013-9170-9

Department of Mathematics, Bar-Ilan University, Ramat Gan 5290002, Israel, and Department of Mathematics, Weizmann Institute of Science, Rehovot 7610001, Israel

E-mail address: tsaban@math.biu.ac.il

URL: http://www.cs.biu.ac.il/~tsaban