SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

May 2016, Vol. 59 052109:1–052109:11
doi: 10.1007/s11432-015-5443-2

Cryptanalysis of an asymmetric cipher protocol using a matrix decomposition problem

Jinhui LIU[1,2], Huanguo ZHANG[1,2]*, Jianwei JIA[1,2], Houzhen WANG[1,2], Shaowu MAO[1,2] & Wanqing WU[1,2]

[1]*Computer School of Wuhan University, Wuhan* 430072*, China;*
[2]*Key Laboratory of Aerospace Information security and trusted computing Ministry of Education, Wuhan* 430072*, China*

Received August 27, 2015; accepted November 1, 2015; published online January 18, 2016

Abstract Advances in quantum computation threaten to break public key cryptosystems such as RSA, ECC, and ElGamal that are based on the difficulty of factorization or taking a discrete logarithm, although up to now, no quantum algorithms have been found to be able to solve certain mathematical problems on non-commutative algebraic structures. Against this background, Raulynaitis et al. have proposed a novel asymmetric cipher protocol using a matrix decomposition problem. Their proposed scheme is vulnerable to a linear algebra attack based on the probable occurrence of weak keys in the generation process. In this paper, we show that the asymmetric cipher of the non-commutative cryptography scheme is vulnerable to a linear algebra attack and that it only requires polynomial time to obtain the equivalent keys for some given public keys. We also propose an improvement to enhance the scheme of Raulynaitis et al.

Keywords cryptography, post-quantum computational cryptography, asymmetric cipher protocol, cryptanalysis, matrix decomposition

Citation Liu J H, Zhang H G, Jia J W, et al. Cryptanalysis of an asymmetric cipher protocol using a matrix decomposition problem. Sci China Inf Sci, 2016, 59(5): 052109, doi: 10.1007/s11432-015-5443-2

1 Introduction

Most public key cryptosystems used today rely on the assumed difficulty of either factorization or computing discrete logarithms. However, the trustworthiness of these assumptions has been eroded by improvements in factorization algorithms and by polynomial-time quantum algorithms that solve both problems. These are among the reasons that have motivated research into the development of a new family of cryptosystems that can resist quantum computer attacks and that are more efficient in terms of computation. In recent years, cryptographers have been making efforts in the area of post-quantum computational cryptography [1–6]. They have also begun to construct alternative post-quantum (i.e., quantum-resistant) public key cryptosystems from other mathematically intractable problems [7–13].

Before going into details, we would like to mention that nonabelian algebraic structures have already been used in a cryptographic context. For a general introduction to non-commutative cryptography,

* Corresponding author (email: liss@whu.edu.cn)

we refer to [6]. In this paper, we study cryptanalysis of an asymmetric cipher protocol using a matrix decomposition problem (MDP) proposed in [14, 15]. The scheme proposed by Raulynaitis et al. is vulnerable to linear algebra attack based on the probable occurrence of weak keys in the generation process. We show that the scheme is insecure against a linear algebra attack. Using the linear algebra attack, we attempt to analyze the scheme so that we can obtain the equivalent keys from an associated public key with significant probability in a reasonable time. We then analyze the basic rationale for the linear algebra attack. We also propose an improved scheme that remedies the weakness of Raulynaitis's scheme.

The rest of this paper is organized as follows. Section 2 reviews necessary background material. Section 3 gives an overview of the asymmetric cipher scheme based on the MDP proposed in [14, 15]. Section 4 proposes an attack method, and describes the corresponding algorithmic description and efficiency analysis. In Section 5, the modified scheme is proposed. The security analysis is proposed in Section 6. Finally, Section 7 provides some concluding remarks and discusses possible lines of future work.

2 Preliminaries

Throughout this paper, we use the following notation.

Let q be a power of a prime and $\mathbb{F}_q$ be a finite field. For an integer $k > 1$, $GL_k(\mathbb{F}_q)$ is the set of $k \times k$ invertible matrices with entries in $\mathbb{F}_q$, $M_k(\mathbb{F}_q)$ is a set of $k \times k$ matrices with entries in $\mathbb{F}_q$, $I_k \in GL_k(\mathbb{F}_q)$ is the identity matrix and 0_k is the $k \times k$ matrix with all-zero elements.

For a matrix A, A^{T} is the transpose of A. Let

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1k_1} \\ \vdots & \ddots & \vdots \\ a_{k_11} & \cdots & a_{k_1k_1} \end{pmatrix} \in M_{k_1}(\mathbb{F}_q), \quad B \in M_{k_2}(\mathbb{F}_q),$$

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1k_1}B \\ \vdots & \ddots & \vdots \\ a_{k_11}B & \cdots & a_{k_1k_1}B \end{pmatrix}_{k_1k_2 \times k_1k_2}, \quad \overrightarrow{A} = \begin{pmatrix} a_{11} & \cdots & a_{1k_1} & a_{21} & \cdots & \cdots & a_{k_1k_1} \end{pmatrix} \in \mathbb{F}_q^{1 \times k_1^2}.$$

Proposition 1. The Kronecker product "$\otimes$" has the following simple properties:

$$\begin{aligned} &(A \otimes B) \otimes C = A \otimes (B \otimes C), \quad A \otimes (B + C) = (A \otimes B) + (A \otimes C), \\ &(A \otimes B)^{\mathrm{T}} = A^{\mathrm{T}} \otimes B^{\mathrm{T}}, \quad (A \otimes B)(C \otimes D) = AC \otimes BD, \\ &(aA) \otimes B = a(A \otimes B) = A \otimes (aB) \text{ for } \forall a \in \mathbb{F}_q, \\ &(A \otimes B)^{-1} = A^{-1} \otimes B^{-1} \text{ for } A, B \in GL_n(\mathbb{F}_q). \end{aligned}$$

Proposition 2. Stretching of the rows of a matrix into one long row vector "$\overrightarrow{\cdot}$" has the following simple properties:

$$\begin{aligned} &\overrightarrow{\alpha A + \beta B} = \alpha \overrightarrow{A} + \beta \overrightarrow{B}, \quad (\overrightarrow{AX})^{\mathrm{T}} = (A \otimes I)(\overrightarrow{X})^{\mathrm{T}}, \\ &(\overrightarrow{XB})^{\mathrm{T}} = (I \otimes B^{\mathrm{T}})(\overrightarrow{X})^{\mathrm{T}}, \quad (\overrightarrow{ACB})^{\mathrm{T}} = (A \otimes B^{\mathrm{T}})(\overrightarrow{C})^{\mathrm{T}}. \end{aligned}$$

3 Description of the asymmetric cipher

In this section, we briefly review the asymmetric cipher protocol based on the MDP proposed by Raulynaitis et al.

First, the common setting of the public parameters of the proposed scheme is given by

$$\langle M_{L_1}, M_{L_2}, M_{R_1}, M_{R_2}, \mathcal{M}, \mathcal{P}, \mathcal{N} \rangle,$$

where the elements of $\mathcal{M}$ are n-dimensional square matrices with entries in the natural numbers $\mathcal{N} = \{0, 1, 2, \ldots\}$, and the subsets $\mathcal{M}_L \subseteq \mathcal{M}, \mathcal{M}_R \subseteq \mathcal{M}$ are generated by M_L and M_R, respectively. To avoid arithmetic with large integers, the elements of these matrices should be bounded, i.e., their values should not exceed some number $s \in \mathcal{N}$. Let $\mathcal{N} = \{0, 1, 2, \ldots, q\}$, $\mathbb{F}_q = \{0, 1, 2, \ldots, q-1\}$ be two groups, and since $\mathbb{F}_q = \mathcal{N}/\{q\}$, we can consider matrix operations over a finite field $\mathbb{F}_q$. We can also perform the same analysis over the residue class ring $\mathbb{Z}_q$.

$$M_{L_1} = \begin{pmatrix} L_1 & 0 \\ 0 & l_1 I \end{pmatrix}, \quad M_{L_2} = \begin{pmatrix} l_2 I & 0 \\ 0 & L_2 \end{pmatrix}, \quad M_{R_1} = \begin{pmatrix} R_1 & 0 \\ 0 & r_1 I \end{pmatrix}, \quad M_{R_2} = \begin{pmatrix} r_2 I & 0 \\ 0 & R_2 \end{pmatrix}, \tag{1}$$

and $M_{L_1} M_{L_2} = M_{L_2} M_{L_1}, M_{R_1} M_{R_2} = M_{R_2} M_{R_1}$. All block matrices L_1, L_2, R_1, R_2 and I over $\mathbb{F}_q$ are chosen at random with dimension $\frac{n}{2}$, and the scalars l_1, l_2, r_1, r_2 in $\mathbb{F}_q$ are also chosen randomly.

Let $\mathcal{P} = \{p_i()\}$ be the set of all polynomials over $\mathbb{F}_q$. Using randomly generated matrices L_1, L_2, R_1, R_2 of the form (1), the following polynomial matrices can be calculated:

$$\begin{cases} X = p_{X_1}(M_{L_1}) \cdot p_{X_2}(M_{L_2}), \\ Y = p_{Y_1}(M_{R_1}) \cdot p_{Y_2}(M_{R_2}), \\ U = p_{U_1}(M_{L_1}) \cdot p_{U_2}(M_{L_2}), \\ V = p_{V_1}(M_{R_1}) \cdot p_{V_2}(M_{R_2}), \end{cases} \tag{2}$$

where all the coefficients

$$a_1 = (a_{10}, a_{11}, \ldots, a_{1n}), \quad a_2 = (a_{20}, a_{21}, \ldots, a_{2n}), \quad b_1 = (b_{10}, b_{11}, \ldots, b_{1n}), \quad b_2 = (b_{20}, b_{21}, \ldots, b_{2n}),$$
$$c_1 = (c_{10}, c_{11}, \ldots, c_{1n}), \quad c_2 = (c_{20}, c_{21}, \ldots, c_{2n}), \quad d_1 = (d_{10}, d_{11}, \ldots, d_{1n}), \quad d_2 = (d_{20}, d_{21}, \ldots, d_{2n})$$

of the polynomials $p_{X_1}, p_{X_2}, p_{Y_1}, p_{Y_2}, p_{U_1}, p_{U_2}, p_{V_1}, p_{V_2}$ in $\mathcal{P}$ are generated at random over $\mathbb{F}_q$ and $\cdot$ represents a general multiplication operation.

The asymmetric cipher protocol based on the MDP can now be described as follows.

To encrypt and decrypt the message t, both Bob and Alice know how to form an n-dimensional encoded matrix T corresponding to t.

KeyGen: Bob chooses at random secret vectors $X \in \mathcal{M}_L, Y \in \mathcal{M}_R$ and a fully filled square n-dimensional matrix $Q \in \mathcal{M}$, and calculates $A = XQY$. The output is (A, Q) as the public key pair and (X, Y) as the private key pair.

Enc:

(1) The input is the public key pair $(M_{L_1}, M_{L_2}, M_{R_1}, M_{R_2}, A, Q, \mathcal{M}, \mathcal{P}, \mathbb{F}_q)$ and the message matrix T.

(2) Alice chooses randomly four polynomials in $\mathcal{P}$ and calculates matrices U, V using (2).

(3) Alice computes $\varepsilon = UAV$ and $\delta = UQV$ by using Bob's PuK (A, Q).

(4) Alice obtains the ciphertext C computed from the formula

$$C = \varepsilon \oplus T = UAV \oplus T,$$

where $\oplus$ is the bitwise sum modulo 2. Then Alice sends $D = (C, \delta)$ to Bob.

Dec: Bob calculates the encoded plaintext T using his private key (X, Y) from the equation

$$X\delta Y \oplus C = T.$$

4 The key recovery attack

This section attempts to attack the asymmetric cipher scheme. The attack makes use of the elementary tools mentioned above and is intended to show the structural vulnerabilities of the system. We know that an attacker $\mathcal{A}$ is observing the asymmetric cipher protocol, and he is able to get the information

$(M_{L_1}, M_{L_2}, M_{R_1}, M_{R_2}, A, Q, D, \delta)$. According to $K = X\delta Y = XUQVY = UXQYV = UAV$, he searches for a pair of matrices (X, Y) such that

$$\begin{cases} A = XQY, \\ XM_{L_1} = M_{L_1}X, \\ XM_{L_2} = M_{L_2}X, \\ YM_{R_1} = M_{R_1}Y, \\ YM_{R_2} = M_{R_2}Y. \end{cases} \tag{3}$$

Then the proposed scheme always has a weakness. It remains to analyze the asymmetric cipher, which can be done as follows.

Proposition 3. If an adversary can find matrices $\widetilde{X}, \widetilde{Y}$ satisfying the equations (3), then the asymmetric cipher protocol based on the MDP can be broken.

Proof. If an adversary can find matrices $\widetilde{X}, \widetilde{Y}$ satisfying the equations (3), then the asymmetric cipher protocol based on the MDP can be summarized as follows.

From $M_{L_1}M_{L_2} = M_{L_2}M_{L_1}, M_{R_1}M_{R_2} = M_{R_2}M_{R_1}$, $\widetilde{X}M_{L_1} = M_{L_1}\widetilde{X}, \widetilde{X}M_{L_2} = M_{L_2}\widetilde{X}, \widetilde{Y}M_{R_1} = M_{R_1}\widetilde{Y}, \widetilde{Y}M_{R_2} = M_{R_2}\widetilde{Y}$, and

$$\begin{cases} U = p_{U_1}(M_{L_1}) \cdot p_{U_2}(M_{L_2}), \\ V = p_{V_1}(M_{R_1}) \cdot p_{V_2}(M_{R_2}), \end{cases} \tag{4}$$

we have

$$\begin{aligned} U \cdot \widetilde{X} &= p_{U_1}(M_{L_1}) \cdot p_{U_2}(M_{L_2}) \cdot \widetilde{X} \\ &= p_{U_1}(M_{L_1}) \cdot \widetilde{X} \cdot p_{U_2}(M_{L_2}) \\ &= \widetilde{X} \cdot p_{U_1}(M_{L_1}) \cdot p_{U_2}(M_{L_2}) \\ &= \widetilde{X} \cdot U, \end{aligned} \tag{5}$$

and

$$\begin{aligned} V \cdot \widetilde{Y} &= p_{V_1}(M_{L_1}) \cdot p_{V_2}(M_{L_2}) \cdot \widetilde{Y} \\ &= p_{V_1}(M_{L_1}) \cdot \widetilde{Y} \cdot p_{V_2}(M_{L_2}) \\ &= \widetilde{Y} \cdot p_{V_1}(M_{L_1}) \cdot p_{V_2}(M_{L_2}) \\ &= \widetilde{Y} \cdot V, \end{aligned} \tag{6}$$

where $\cdot$ represents a general multiplication operation.

Since $A = \widetilde{X}Q\widetilde{Y}$, the attacker calculates the decryption algorithm:

$$\begin{aligned} \widetilde{X}\delta\widetilde{Y} \oplus C &= \widetilde{X}UQV\widetilde{Y} \oplus C \\ &= U\widetilde{X}Q\widetilde{Y}V \oplus C \\ &= UAV \oplus C \\ &= UAV \oplus UAV \oplus T \\ &= T. \end{aligned} \tag{7}$$

The attacker thereby obtains the message T.

4.1 Algorithmic description and efficiency analysis

If either $X = p_{X_1}(M_{L_1}) \cdot p_{X_2}(M_{L_2})$ or $Y = p_{Y_1}(M_{R_1}) \cdot p_{Y_2}(M_{R_2})$ is an invertible matrix (or if both are invertible), the matrix equations (3) are equivalent to linear equations. Suppose that the matrix X is

invertible; then solving the matrix equations (3) is equivalent to solving the following multivariate linear equations:

$$\begin{cases} X^{-1}A = QY, \\ X^{-1}M_{L_1} = M_{L_1}X^{-1}, \\ X^{-1}M_{L_2} = M_{L_2}X^{-1}, \\ YM_{R_1} = M_{R_1}Y, \\ YM_{R_2} = M_{R_2}Y. \end{cases} \tag{8}$$

Using Proposition 2, it is easy to obtain the following result:

$$WN^{\mathrm{T}} = 0, \tag{9}$$

where

$$W = \begin{pmatrix} I_n \otimes A^{\mathrm{T}} & -Q \otimes I_n \\ M_{L_1} \otimes I_n - I_n \otimes M_{L_1}^{\mathrm{T}} & 0 \\ M_{L_2} \otimes I_n - I_n \otimes M_{L_2}^{\mathrm{T}} & 0 \\ 0 & M_{R_1} \otimes I_n - I_n \otimes M_{R_1}^{\mathrm{T}} \\ 0 & M_{R_2} \otimes I_n - I_n \otimes M_{R_2}^{\mathrm{T}} \end{pmatrix}_{5n^2 \times 2n^2}, \quad N = \left(\overrightarrow{X^{-1}}, \overrightarrow{Y} \right)_{1 \times 2n^2},$$

$\overrightarrow{X^{-1}}, \overrightarrow{Y}$ are the stretches of the matrices X^{-1}, Y, $\det(X^{-1}) \neq 0, \det(Y) \neq 0$, $\otimes$ represents the Kronecker product, I_n is the $n \times n$ identity, and 0 is the matrix with all-zero elements.

The method for calculating a matrix pair $\widetilde{X^{-1}}, \widetilde{Y}$ of (3) is shown in Algorithm 1. Formally, the key recovery attack can be described by Algorithm 1. It takes as input matrices $(M_{L_1}, M_{L_2}, M_{R_1}, M_{R_2}, A, Q)$ and outputs equivalent keys $\widetilde{X^{-1}}, \widetilde{Y}$.

Algorithm 1: Recovering equivalent keys for a given public key

Input: Matrices $(M_{L_1}, M_{L_2}, M_{R_1}, M_{R_2}, A, Q)$.

Output: Equivalent keys $\widetilde{X^{-1}}, \widetilde{Y}$.

Step 1: Solve the homogeneous linear equations in the $2n^2$ entries of the unknown vector N: $WN^{\mathrm{T}} = 0$.

Step 2: Fix a basis for the solution space and transform the vectors $\widetilde{\overrightarrow{X^{-1}}}, \widetilde{\overrightarrow{Y}}$ to matrices $\widetilde{X^{-1}}, \widetilde{Y}$, respectively. Pick a random solution matrix $\widetilde{X^{-1}}$ until $\widetilde{X^{-1}}$ is invertible.

Step 3: Compute $\widetilde{X^{-1}}^{-1} = \widetilde{X}$.

Step 4: Return $\widetilde{X}, \widetilde{Y}$.

Combining the above discussions, let us make a performance evaluation on Algorithm 1. The classical techniques for matrix multiplication/inversion in $\mathbb{F}_q$ take about $\mathcal{O}(n^\omega \log^2 q)$ bit operations, since the best known algorithm for the product of two $n \times n$ matrices requires $\mathcal{O}(n^\omega)(\omega \approx 2.3755)$ $\mathbb{F}_q$ operations and each $\mathbb{F}_q$ operation needs $\mathcal{O}(\log^2 q)$ bit operations [14, 16, 17]. Suppose that the rank of a $5n^2 \times 2n^2$ coefficient matrix W is r. By employing the method of Gaussian elimination, we know that $0 < r \leqslant 2n^2$. If $r = 2n^2$, then the matrix W has full column rank, i.e., $N = 0$. We know that there is at least one solution to the equations (3), namely, the private keys X and Y, and thus $0 < r < 2n^2$. Then, it remains to analyze the complexity of Algorithm 1, which can be concluded from Table 1.

The number of free variables of the matrix equations (5) is $2n^2 - r$; then the total expected running time of step 2 is $5n^2(2n^2 - r)^{\omega-1} \leqslant 5 \cdot (2n)^{2\omega}$. To generate one pair of random elements $\widetilde{X^{-1}}, \widetilde{Y}$, we take a linear combination of a basis of the solution space. Thus, the probability of an invertible matrix $\widetilde{X^{-1}}$ is

$$\begin{aligned} \mathbf{P}(\mathrm{rank}(\widetilde{X^{-1}}) = n) &= \frac{q^{n(n-1)/2} \prod_{i=1}^{n} (q^i - 1)}{q^{n^2}} \\ &= \left(1 - \frac{1}{q}\right) \cdot \left(1 - \frac{1}{q^2}\right) \cdot \cdots \cdot \left(1 - \frac{1}{q^n}\right) \approx 1 - \frac{1}{q}. \end{aligned}$$

Table 1 Computation complexity of Algorithm 1

Computational content	Computational cost	Explanation
$W\overrightarrow{N}^{\mathrm{T}}=0$	$\mathcal{O}(5\cdot(2n)^{2\omega}\log^2 q)$	$5n^2$ equations in $2n^2$ variables
Invertible solutions $\widetilde{X^{-1}},\widetilde{Y}$	$\mathcal{O}(5n^2(2n^2-r)^{\omega-1}\log^2 q)$	A linear combination of solution spaces
$\widetilde{X^{-1}}^{-1}$	$\mathcal{O}(n^{\omega}\log^2 q)$	1 inversion

Table 2 Practical implementations of Algorithm 1

q	n	Design security	Size of public key (KB)	Computational complexity	Attack time (s)
2^{16}	5	2^{160}	0.195	$2^{19.03}$	0.094
2^{18}	5	2^{180}	0.195	$2^{19.37}$	0.156
2^{20}	5	2^{200}	0.195	$2^{19.68}$	0.167
2^{25}	5	2^{250}	0.195	$2^{20.32}$	0.750
2^{20}	6	2^{240}	0.281	$2^{20.93}$	0.344
2^{20}	10	2^{400}	0.781	$2^{24.43}$	4.297
2^{20}	20	2^{800}	3.125	$2^{29.18}$	213.672
2^{20}	40	2^{1600}	12.500	$2^{33.93}$	11742.500

Now, if we neglect small constant factors, the key recovery attack against an asymmetric cipher based on the MDP can be completed with a complexity of $\mathcal{O}(n^{2\omega}\log^2 q)$.

So

$$\widetilde{X}\delta\widetilde{Y}=\widetilde{X}UQV\widetilde{Y}=U\widetilde{X}Q\widetilde{Y}V=UAV,$$

and the message matrix $T=\widetilde{X}\delta\widetilde{Y}\oplus C$ is obtained.

If the matrix Y is invertible, then the analysis of the matrix equations (3) is the same as above. If neither $X=p_{X_1}(M_{L_1})\cdot p_{X_2}(M_{L_2})$ nor $Y=p_{Y_1}(M_{R_1})\cdot p_{Y_2}(M_{R_2})$ is an invertible matrix, but $l_1\cdot l_2=0$, $r_1\cdot r_2=0$, and L_1,L_2,R_1,R_2 are invertible matrices, then the scheme of Raulynaitis et al. can also be insecure when using block matrices. If neither $X=p_{X_1}(M_{L_1})\cdot p_{X_2}(M_{L_2})$ nor $Y=p_{Y_1}(M_{R_1})\cdot p_{Y_2}(M_{R_2})$ is an invertible matrix and $l_1\cdot l_2\neq 0, r_1\cdot r_2\neq 0$, the key recovery attack fails.

4.2 Practical implementations of Algorithm 1

We carry out our experiments on a 2.6 GHz Intel processor PC using the Magma implementation with different parameters for Algorithm 1 and then collect the results of our analysis in Table 2.

4.3 A toy example

In order to illustrate the steps in our cryptanalysis over $\mathbb{M}_n(\mathbb{F}_q)$, we give the following toy example.

Let $q=2^{16}$ and $n=4$. The public keys and known information δ are

$$M_{L_1}=\begin{pmatrix}24771 & 63557 & 0 & 0\\ 33158 & 24770 & 0 & 0\\ 0 & 0 & 678 & 0\\ 0 & 0 & 0 & 678\end{pmatrix},\quad M_{L_2}=\begin{pmatrix}634 & 0 & 0 & 0\\ 0 & 634 & 0 & 0\\ 0 & 0 & 34624 & 5294\\ 0 & 0 & 9152 & 33873\end{pmatrix},$$

$$M_{R_1}=\begin{pmatrix}50781 & 2658 & 0 & 0\\ 42551 & 16669 & 0 & 0\\ 0 & 0 & 49513 & 0\\ 0 & 0 & 0 & 49513\end{pmatrix},\quad M_{R_2}=\begin{pmatrix}25206 & 0 & 0 & 0\\ 0 & 25206 & 0 & 0\\ 0 & 0 & 28496 & 57820\\ 0 & 0 & 53610 & 2838\end{pmatrix},$$

$$A=\begin{pmatrix} 27132 & 33378 & 36182 & 47677 \\ 32769 & 52971 & 33326 & 19680 \\ 2633 & 25484 & 32334 & 59028 \\ 58290 & 41126 & 25688 & 32692 \end{pmatrix}, \delta=\begin{pmatrix} 28666 & 8567 & 5440 & 7730 \\ 60889 & 6797 & 8156 & 6069 \\ 53021 & 24021 & 54734 & 59413 \\ 35562 & 47467 & 38873 & 44760 \end{pmatrix}.$$

The private keys are

$$X=\begin{pmatrix} 2208 & 55121 & 0 & 0 \\ 18944 & 4951 & 0 & 0 \\ 0 & 0 & 57196 & 17143 \\ 0 & 0 & 21153 & 24741 \end{pmatrix}, \quad Y=\begin{pmatrix} 61720 & 4399 & 0 & 0 \\ 56552 & 1991 & 0 & 0 \\ 0 & 0 & 37550 & 50372 \\ 0 & 0 & 37260 & 62813 \end{pmatrix}.$$

Using Algorithm 1, we calculate a basis of the solution space $W\overrightarrow{N}^{\mathrm{T}}=0$, where $W=\begin{pmatrix} W_1 & W_2 \end{pmatrix}$, with

$$W_1=\left(\begin{array}{cccccccccccccccc}
27132 & 32769 & 2633 & 58290 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
33378 & 52971 & 25484 & 41126 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
36182 & 33326 & 32334 & 25688 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
47677 & 19680 & 59028 & 32692 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 27132 & 32769 & 2633 & 58290 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 33378 & 52971 & 25484 & 41126 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 36182 & 33326 & 32334 & 25688 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 47677 & 19680 & 59028 & 32692 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 27132 & 32769 & 2633 & 58290 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 33378 & 52971 & 25484 & 41126 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 36182 & 33326 & 32334 & 25688 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 47677 & 19680 & 59028 & 32692 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 27132 & 32769 & 2633 & 58290 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 33378 & 52971 & 25484 & 41126 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 36182 & 33326 & 32334 & 25688 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 47677 & 19680 & 59028 & 32692 \\
0 & 33158 & 0 & 0 & 63557 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
63557 & 1 & 0 & 0 & 0 & 63557 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 25189 & 0 & 0 & 0 & 63557 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 25189 & 0 & 0 & 0 & 63557 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
33158 & 0 & 0 & 0 & 1 & 33158 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 33158 & 0 & 0 & 63557 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 33158 & 0 & 0 & 0 & 25188 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 33158 & 0 & 0 & 0 & 25188 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 25189 & 33158 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 63557 & 25188 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 25189 & 33158 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 63557 & 25188 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 34106 & 9152 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 5294 & 34347 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 34106 & 9152 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 5294 & 34347 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 34106 & 0 & 0 & 0 & 5294 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 34106 & 0 & 0 & 0 & 5294 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9152 & 0 & 0 & 5294 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5294 & 785 & 0 & 0 & 0 & 5294 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9152 & 0 & 0 & 0 & 34347 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9152 & 0 & 0 & 0 & 34347 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9152 & 0 & 0 & 0 & 785 & 9152 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9152 & 0 & 0 & 5294 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right),$$

$$
W_2=\begin{pmatrix}
8471 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2546 & 0 & 0 & 0 & 51431 & 0 & 0 \\
0 & 8471 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2546 & 0 & 0 & 0 & 51431 & 0 & 0 \\
0 & 0 & 8471 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2546 & 0 & 0 & 0 & 51431 & 0 \\
0 & 0 & 0 & 8471 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2546 & 0 & 0 & 0 & 51431 \\
0 & 0 & 0 & 0 & 9367 & 0 & 0 & 0 & 57522 & 0 & 0 & 0 & 63253 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 9367 & 0 & 0 & 0 & 57522 & 0 & 0 & 0 & 63253 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 9367 & 0 & 0 & 0 & 57522 & 0 & 0 & 0 & 63253 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 9367 & 0 & 0 & 0 & 57522 & 0 & 0 & 0 & 63253 \\
4213 & 0 & 0 & 0 & 45660 & 0 & 0 & 0 & 32888 & 0 & 0 & 0 & 3834 & 0 & 0 & 0 \\
0 & 4213 & 0 & 0 & 0 & 45660 & 0 & 0 & 0 & 32888 & 0 & 0 & 0 & 3834 & 0 & 0 \\
0 & 0 & 4213 & 0 & 0 & 0 & 45660 & 0 & 0 & 0 & 32888 & 0 & 0 & 0 & 3834 & 0 \\
0 & 0 & 0 & 4213 & 0 & 0 & 0 & 45660 & 0 & 0 & 0 & 32888 & 0 & 0 & 0 & 3834 \\
19802 & 0 & 0 & 0 & 168 & 0 & 0 & 0 & 36514 & 0 & 0 & 0 & 62992 & 0 & 0 & 0 \\
0 & 19802 & 0 & 0 & 0 & 168 & 0 & 0 & 0 & 36514 & 0 & 0 & 0 & 62992 & 0 & 0 \\
0 & 0 & 19802 & 0 & 0 & 0 & 168 & 0 & 0 & 0 & 36514 & 0 & 0 & 0 & 62992 & 0 \\
0 & 0 & 0 & 19802 & 0 & 0 & 0 & 168 & 0 & 0 & 0 & 36514 & 0 & 0 & 0 & 62992 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 42551 & 0 & 0 & 2658 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2658 & 34624 & 0 & 0 & 0 & 2658 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1844 & 0 & 0 & 0 & 2658 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1844 & 0 & 0 & 0 & 2658 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
42551 & 0 & 0 & 0 & 34624 & 42551 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 42551 & 0 & 0 & 2658 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 42551 & 0 & 0 & 0 & 32884 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 42551 & 0 & 0 & 0 & 32884 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1844 & 42551 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2658 & 32884 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1844 & 42551 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2658 & 32884 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 3366 & 53610 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 57820 & 26976 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 3366 & 53610 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 57820 & 26976 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3366 & 0 & 0 & 0 & 57820 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3366 & 0 & 0 & 0 & 57820 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 53610 & 0 & 0 & 57820 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 57820 & 25670 & 0 & 0 & 0 & 57820 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 53610 & 0 & 0 & 0 & 26976 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 53610 & 0 & 0 & 0 & 26976 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 53610 & 0 & 0 & 0 & 25670 & 53610 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 53610 & 0 & 0 & 57820 & 0
\end{pmatrix}.
$$

Then the basis ε is (47351 44245 0 0 36549 61217 0 0 0 0 14426 50125 0 0 28101 36412 59301 59889 0 0 10345 11128 0 0 0 0 53842 39711 0 0 15847 738). We transform the vector ε to matrices

$$
\widetilde{X^{-1}}=\begin{pmatrix} 47351 & 44245 & 0 & 0 \\ 36549 & 61217 & 0 & 0 \\ 0 & 0 & 14426 & 50125 \\ 0 & 0 & 28101 & 36412 \end{pmatrix}, \quad \widetilde{Y}=\begin{pmatrix} 59301 & 59889 & 0 & 0 \\ 10345 & 11128 & 0 & 0 \\ 0 & 0 & 53842 & 39711 \\ 0 & 0 & 15847 & 738 \end{pmatrix}.
$$

Then

$$
\widetilde{X}=\widetilde{X^{-1}}^{-1}=\begin{pmatrix} 59489 & 1 & 0 & 0 \\ 23235 & 39688 & 0 & 0 \\ 0 & 0 & 11265 & 30081 \\ 0 & 0 & 28437 & 6325 \end{pmatrix}.
$$

Thus, we obtain

$$UAV = \widetilde{X}\delta\widetilde{Y} = \begin{pmatrix} 39880 & 52090 & 6171 & 18609 \\ 64382 & 10999 & 52603 & 60868 \\ 9239 & 55355 & 46685 & 23938 \\ 41207 & 19526 & 58081 & 59578 \end{pmatrix}.$$

That is to say, we obtain the message matrix $T = \widetilde{X}\delta\widetilde{Y} \oplus C$.

5 Improvement of the scheme of Raulynaitis et al.

When neither X nor Y is an invertible matrix and $l_1 \cdot l_2 \neq 0, r_1 \cdot r_2 \neq 0$, the key recovery attack fails, as mentioned in Section 4. We therefore propose an improved scheme that can protect against key recovery attack.

The common setting of the public parameters of the improved scheme is given by $\langle M_{L_1}, M_{L_2}, M_{R_1}, M_{R_2}, M_n(\mathbb{F}_q), \mathcal{P}, \mathbb{F}_q\rangle$ and $\mathcal{M}_L \subseteq M_n(\mathbb{F}_q), \mathcal{M}_R \subseteq M_n(\mathbb{F}_q)$ are generated by non-invertible matrices M_L, M_R, respectively.

$$M_{L_1} = \begin{pmatrix} L_1 & 0 \\ 0 & l_1 I \end{pmatrix}, \quad M_{L_2} = \begin{pmatrix} l_2 I & 0 \\ 0 & L_2 \end{pmatrix}, \quad M_{R_1} = \begin{pmatrix} R_1 & 0 \\ 0 & r_1 I \end{pmatrix}, \quad M_{R_2} = \begin{pmatrix} r_2 I & 0 \\ 0 & R_2 \end{pmatrix}, \tag{10}$$

all block matrices L_1, L_2, R_1, R_2 over $\mathbb{F}_q$ are chosen to be non-invertible matrices with dimension $\frac{n}{2}$, and $l_1, l_2, r_1, r_2 \in \mathbb{F}_q/\{0\}$ are chosen randomly. We choose four $(\frac{n}{2}-1)\times\frac{n}{2}$ matrices $\widetilde{L}_1, \widetilde{L}_2, \widetilde{R}_1, \widetilde{R}_2$ at random, then compute a random linear combination of $\frac{n}{2}-1$ rows to generate the $\frac{n}{2}$ rows of the matrices L_1, L_2, R_1, R_2, respectively.

Using the generated non-invertible matrices L_1, L_2, R_1, R_2 of the form (10), we calculate the following polynomial matrices:

$$\begin{cases} X = p_{X_1}(M_{L_1}) \cdot p_{X_2}(M_{L_2}), \\ Y = p_{Y_1}(M_{R_1}) \cdot p_{Y_2}(M_{R_2}), \\ U = p_{U_1}(M_{L_1}) \cdot p_{U_2}(M_{L_2}), \\ V = p_{V_1}(M_{R_1}) \cdot p_{V_2}(M_{R_2}), \end{cases} \tag{11}$$

where both X and Y are non-invertible matrices, $\mathcal{P} = \{p_i()\}$ is the set of all polynomials over $\mathbb{F}_q$ and

$$a_1 = (a_{10}, a_{11}, \ldots, a_{1n}), \quad a_2 = (a_{20}, a_{21}, \ldots, a_{2n}), \quad b_1 = (b_{10}, b_{11}, \ldots, b_{1n}), \quad b_2 = (b_{20}, b_{21}, \ldots, b_{2n}),$$
$$c_1 = (c_{10}, c_{11}, \ldots, c_{1n}), \quad c_2 = (c_{20}, c_{21}, \ldots, c_{2n}), \quad d_1 = (d_{10}, d_{11}, \ldots, d_{1n}), \quad d_2 = (d_{20}, d_{21}, \ldots, d_{2n})$$

are the coefficients of the polynomials $p_{X_1}, p_{X_2}, p_{Y_1}, p_{Y_2}, p_{U_1}, p_{U_2}, p_{V_1}, p_{V_2} \in \mathcal{P}$.

The improved asymmetric cipher protocol based on the MDP can be described as follows.

First, both Alice and Bob know how to form an n-dimensional encoded matrix T corresponding to t.

KeyGen: Bob chooses at random singular matrices $X \in \mathcal{M}_L, Y \in \mathcal{M}_R$ and an $n \times n$ matrix $Q \in \mathcal{M}$, and calculates $A = XQY$. The output is (A, Q) as the public key pair and (X, Y) as the private key pair.

Enc:

(1) The input is the public key pair $(M_{L_1}, M_{L_2}, M_{R_1}, M_{R_2}, A, Q)$ and the message matrix T.

(2) Alice chooses randomly four polynomials in $\mathcal{P}$ and calculates the matrices U, V using (11).

(3) Alice computes $\varepsilon = UAV$ and $\delta = UQV$ by using Bob's PuK (A, Q).

(4) Alice obtains the ciphertext C computed from the formula

$$C = \varepsilon \oplus T = UAV \oplus T,$$

where $\oplus$ is the bitwise sum modulo 2. Then Alice sends $D = (C, \delta)$ to Bob.

Dec: Bob calculates the encoded plaintext T using his private key (X, Y) from the equation

$$X\delta Y \oplus C = T.$$

6 Cryptanalysis of our proposed scheme

Our improved scheme uses the asymmetric cipher protocol based on the MDP proposed by Raulynaitis et al., which relies on the solution of a multivariate polynomial system of equations.

6.1 Direct attack

In the improved scheme, it is known that even the solution of a multivariate quadratic polynomial system of equations over any field is an NP-complete problem. $X = p_{X_1}(M_{L_1}) \cdot p_{X_2}(M_{L_2})$ and $Y = p_{Y_1}(M_{R_1}) \cdot p_{Y_2}(M_{R_2})$ can be rewritten in the following way:

$$X = \sum_{i,j} a_{1i} a_{2j} M_{L_1}^i M_{L_2}^j, \quad Y = \sum_{k,l} b_{1k} b_{2l} M_{R_1}^k M_{R_2}^l.$$

So

$$A = \sum_{i,j,k,l} a_{1i} a_{2j} b_{1k} b_{2l} M_{L_1}^i M_{L_2}^j M_{R_1}^k M_{R_2}^l.$$

Hence, the security of the proposed improved scheme is based on the solution of a system of multivariate equations of fourth order, which is a hard problem.

6.2 Linear algebra attack

Since $l_1, l_2, r_1, r_2 \in \mathbb{F}_q/\{0\}$, both $X = p_{X_1}(M_{L_1}) \cdot p_{X_2}(M_{L_2})$ and $Y = p_{Y_1}(M_{R_1}) \cdot p_{Y_2}(M_{R_2})$ are non-invertible matrices, and therefore a linear algebra attack will fail.

7 Conclusion

We have presented a cryptanalysis of the scheme of Raulynaitis et al. by showing that their scheme is vulnerable to a linear algebra attack. We have shown that an asymmetric cipher based on the MDP is insecure in the sense that an attacker who is able to solve the linear equations with high efficiency can break the asymmetric cipher scheme except in the case where X, Y are not invertible matrices and $l_1 \cdot l_2 \neq 0, r_1 \cdot r_2 \neq 0$. We have proposed an improved scheme and have discussed the enhanced security features that provide good protection against the aforementioned attack. The question whether there exist groups on which an asymmetric cipher scheme based on the MDP is secure remains open. When designing an asymmetric cipher based on the MDP on other groups, the considerations of this paper must be taken into account. Another open question concerns whether it is possible to use several nonabelian algebraic structures to construct a public key cryptosystem with the potential to resist attacks from known quantum algorithms.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61303212, 61170080, 61202386), State Key Program of National Natural Science of China (Grant Nos. 61332019, U1135004), National Key Basic Research Program of China (Grant No. 2014CB340600), Major Research Plan of the National Natural Science Foundation of China (Grant No. 91018008), and Hubei Natural Science Foundation of China (Grant Nos. 2011CDB453, 2014CFB440).

Conflict of interest The authors declare that they have no conflict of interest.

References

1 Cao Z. New Directions of Modern Cryptography. Boca Raton: CRC Press, 2012. 10–255
2 Peikert C. Lattice cryptography for the internet. In: Mosca M, ed. Post-Quantum Cryptography. Waterloo: Springer, 2014. 197–219
3 Shi J J, Shi R H, Guo Y, et al. Batch proxy quantum blind signature scheme. Sci China Inf Sci, 2013, 56: 052115
4 Song F. A note on quantum security for post-quantum cryptography. In: Mosca M, ed. Post-Quantum Cryptography. Waterloo: Springer, 2014. 246–265

5 Tsaban B. Polynomial-time solutions of computational problems in noncommutative-algebraic cryptography. J Cryptol, 2013, 28: 601–622
6 Zhang H G, Liu J H, Jia J W, et al. A survey on applications of matrix decomposition in cryptography. J Cryptol Res, 2014, 1: 341–357
7 Mao S W, Zhang H G, Wu W Q, et al. A resistant quantum key exchange protocol and its corresponding encryption scheme. China Commun, 2014, 11: 131–141
8 Wang H Z, Zhang H G, Wang Z Y, et al. Extended multivariate public key cryptosystems with secure encryption function. Sci China Inf Sci, 2011, 54: 1161–1171
9 Ling S, Phan D H, Stehlé D, et al. Hardness of k-LWE and applications in traitor tracing. In: Proceedings of Advances in Cryptology-CRYPTO. Berlin: Springer, 2014. 315–334
10 Regev O. On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing. New York: ACM Press, 2005. 84–93
11 Braun J, Buchmann J, Mullan C, et al. Long term confidentiality: a survey. Design Code Cryptogr, 2014, 71: 459–478
12 Wang S B, Zhu Y, Ma D, et al. Lattice-based key exchange on small integer solution problem. Sci China Inf Sci, 2014, 57: 112111
13 Albrecht M R, Faugere J C, Fitzpatrick R, et al. Practical cryptanalysis of a public-key encryption scheme based on new multivariate quadratic assumptions. In: Proceedings of Public Key Cryptography-PKC. Berlin: Springer, 2014. 446–464
14 Raulynaitis A, Sakalauskas E, Japertas S. Security analysis of asymmetric cipher protocol based on matrix decomposition problem. Informatica, 2010, 21: 215–228
15 Raulynaitis A, Japertas S. Asymmetric cipher protocol using decomposition problem. In: Proceedings of Information Research and Applications, Varna, 2008. 107–111
16 Gashkov S B, Sergeev I S. Complexity of computation in finite fields. J Math Sci, 2013, 191: 661–685
17 Gu L, Zheng S. Conjugacy systems based on nonabelian factorization problems and their applications in cryptography. J Appl Math, 2014, 52: 1–9