Similar Keys of Multivariate Quadratic Public Key Cryptosystems

Yuh-Hua Hu[1], Lih-Chung Wang[2], Chun-Yen Chou[3], and Feipei Lai[4]

[1] Department of Computer Science and Information Engineering,
National Taiwan University, Taipei 106, Taiwan
d92015@csie.ntu.edu.tw

[2] Department of Applied Mathematics,
National Donghwa University,
Hualien 974, Taiwan
lcwang@mail.ndhu.edu.tw

[3] Department of Mathematical Education,
National Hualien University of Education,
Hualien 970, Taiwan
choucy@mail.nhlue.edu.tw

[4] Departments of Electrical Engineering & of Computer Science
and Information Engineering,
National Taiwan University, Taipei 106, Taiwan
flai@ntu.edu.tw

Abstract. Most multivariate schemes have potentially much higher performance than other public key cryptosystems [15] [4] [1] [2]. Wolf and Preneel [16] show multivariate quadratic public key schemes have many equivalent keys and provide some transformations to identify the keys. In this paper, we propose the idea of similar keys of MQ-based public key cryptosystems(PKCs) and provide a method to reduce the size of private key in MQ-based PKCs to 50% $\sim$ 70% of its original size. And our method is generic for most MQ-based PKCs except for UOV-like and STS-like schemes. Moreover, our method remains the equivalent security and efficiency with original MQ-based PKCs.

Keywords: MQ, multivariate, public key cryptosystem, digital signature, similar key.

1 Introduction

Public key cryptography is involving the use of two separate keys, and the use of two keys has profound consequences in the areas of non-repudiation, confidentiality, and authentication. For example, on-line transactions need the digital signature schemes to verify the validness, the e-mail security application like PGP[18] needs the public key cryptosystem to protect the session key, and the heart of the authentication service X.509[18] is public key certificate. Finding a efficient, secure and easy to implement PKC is helpful to the network security

Y.G. Desmedt et al. (Eds.): CANS 2005, LNCS 3810, pp. 211–222, 2005.

application. Most MQ-based PKCs are faster than other PKCs in key generation/signing or decrypting/verifying or encrypting [15] [4] [1] [2]. Hence, they may be applied in more occasions. However, the key size of MQ-based PKCs is their drawback.

Number-theoretical PKCs have relatively small private key size, for example RSA-1024 bits, ECC-163bits [7] [5]. MQ-based PKCs have a large size of private key, such as C^*[8], HFE[11], QUARTZ[12], SFLASH^{v3}[2], TRMS[15], TTS[1] and UOV[6]. The reason is that most MQ-based PKCs need to store the affine transformations, consisting of an invertible matrix and constant offset, and the coefficients of polynomials in φ_2. The coefficients of the affine transformations are the major parts of the private key.

Changing the affine transformation is an intuitive way to reduce the size of private key. Wang et al. [15] used the extension field instead of the ground field and Hu et al. [4] used the elementary row operations to reduce the size of private key, and both of them speeded up the signing or decrypting time. Though there is still no attack to these specific affine transformations, they did not prove that the specific affine transformation has the same security with arbitrary invertible matrix.

Wolf and Preneel[16] showed some systematic schemes to analyze the equivalent keys. And they provide the concept and the normal forms to reduce the private key. In this paper we introduce the idea of similar keys of MQ-based PKCs, and give a model for most MQ-based PKCs that can reduce the size of the private key to 50% $\sim$ 70% of original size except for UOV-like and STS-like [17] schemes, and we sketch that the new model has the same security as the original model.

In Section 2, we describe the model of MQ-based public key scheme. In Section 3, we define the similar key of MQ-based PKCs. In Section 4, we give our model to reduce the keys and the performance. In Section 5, we discuss and analyze our model. And our conclusion is in Section 6.

2 MQ-Based PKCs

For a typical MQ-based PKC, they operate on a base field $\mathbb{K}$. And its public key is composed of three maps, $\varphi_3 \circ \varphi_2 \circ \varphi_1$, and its private key is the triple $(\varphi_1^{-1}, \varphi_2, \varphi_3^{-1})$. φ_1 and φ_3 are affine transformations in $\mathbb{K}^n$ and $\mathbb{K}^m$ respectively and φ_1^{-1} and φ_3^{-1} are their inverse transformations. The φ_2 is a quadratic transformation and the structure of φ_2 in each MQ-based PKC is different (HFE, SFLASH^{v3}, C^*, QUARTZ, TTS, TRMS, UOV). We illustrate the idea of similar keys with TRMS. The following example is revised in the workshop of PKC2005 [13].

2.1 Structure of TRMS

There are a variety of schemes of TRMS which are all based on tractable rational maps. Tractable rational maps on $\mathbb{K}^n$ are invertible affine transformations or, after a rearrangement of indices if necessary, functions of the following form $\varphi : \mathbb{K}^n \to \mathbb{K}^n$,

$$\begin{cases} y_1 &= r_1(x_1) \\ y_2 &= r_2(x_2)\dfrac{p_2(x_1)}{q_2(x_1)} + \dfrac{f_2(x_1)}{g_2(x_1)} \\ &\vdots \\ y_k &= r_k(x_k)\dfrac{p_k(x_1, x_2, \ldots, x_{k-1})}{q_k(x_1, x_2, \ldots, x_{k-1})} + \dfrac{f_k(x_1, x_2, \ldots, x_{k-1})}{g_k(x_1, x_2, \ldots, x_{k-1})} \\ &\vdots \\ y_n &= r_n(x_n)\dfrac{p_n(x_1, x_2, \ldots, x_{n-1})}{q_n(x_1, x_2, \ldots, x_{n-1})} + \dfrac{f_n(x_1, x_2, \ldots, x_{n-1})}{g_n(x_1, x_2, \ldots, x_{n-1})} \end{cases}$$

where for $i = 2, 3, \ldots, n$, p_i, q_i, f_i, g_i are polynomials, and for $i = 1, 2, \ldots, n$, r_i is a permutation polynomial on $\mathbb{K}$. That is, r_i is a polynomial function which is also a bijection from $\mathbb{K}$ onto itself.

Let $\mathbb{K} = GF(2^8)$. We will construct 3 maps $\varphi_1 : \mathbb{K}^{28} \to \mathbb{K}^{28}$, $\varphi_2 : \mathbb{K}^{28} \to \mathbb{K}^{20}$, $\varphi_3 : \mathbb{K}^{20} \to \mathbb{K}^{20}$ where φ_1, φ_3 are invertible affine transformations, $\varphi_2 = \pi \circ \widetilde{\varphi_2} \circ i$ with π a projection, i an imbedding, and $\widetilde{\varphi_2}$ identified as a tractable rational map over some extension field over $\mathbb{K}$.

Public Key. The public key is the result of the composition map $\varphi_3 \circ \varphi_2 \circ \varphi_1$.

Private Key. The private key is the triple $(\varphi_1^{-1}, \varphi_2, \varphi_3^{-1})$.

Signing. To sign a message M, first we compute its hash $\mathbf{z} = H(M) \in \mathbb{K}^{20}$ by a publicly agreed hash function. Then do $\mathbf{y} = \varphi_3^{-1}(\mathbf{z})$. Then choose 8 nonzero random numbers $r_1, r_2, \ldots, r_8$. Then get $\mathbf{x}$ by identifying it with $(\widetilde{\varphi_2}^{-1} \circ i)(r_1, r_2, \ldots, r_8, \mathbf{y})$ which is computed by a sequence of substitutions. Then get the signature $\mathbf{w} = \varphi_1^{-1}(\mathbf{x})$.

Verifying. To verify a signature $\mathbf{w}$, simply check if $V(\mathbf{w}) = (\varphi_3 \circ \varphi_2 \circ \varphi_1)(\mathbf{w}) = (\varphi_3 \circ \pi \circ \widetilde{\varphi_2} \circ i)(\mathbf{x}) = (\varphi_3 \circ \pi)(r_1, r_2, \ldots, r_8, \mathbf{y}) = \varphi_3(\mathbf{y}) = \mathbf{z} = H(M)$.

2.2 Details of φ_1 and φ_3

Let φ_1, φ_3 be invertible affine maps on $\mathbb{K}^{28}$ and $\mathbb{K}^{20}$ respectively such that $\varphi_1 = T_1 \circ L_1 \circ D_1 \circ U_1$ and $\varphi_3 = T_3 \circ L_3 \circ D_3 \circ U_3$ where

1. T_1 is a translation on $\mathbb{K}^{28}$ and T_3 is a translation on $\mathbb{K}^{20}$. T_3 is used to cancel the constant terms in the public key. Therefore T_3 is not chosen but determined.
2. In general, L_1 is a 28×28 lower triangular matrix over $\mathbb{K}$ and L_3 is a 20×20 lower triangular matrix over $\mathbb{K}$ such that both with diagonal entries equal to 1.
3. D_1 is a 28×28 diagonal matrix over $\mathbb{K}$ and D_3 is a 20×20 diagonal matrix over $\mathbb{K}$.
4. In general, U_1 is a 28×28 upper triangular matrix over $\mathbb{K}$ and U_3 is a 20×20 upper triangular matrix over $\mathbb{K}$ such that both with diagonal entries equal to 1.

The scheme in [15] is a special form of φ_1 and φ_3.

2.3 Details of φ_2

Let $\mathbb{L}, \mathbb{L}', \mathbb{L}''$ be the finite extension fields of $\mathbb{K}$ such that $\mathbb{K} \subset \mathbb{L}'' \subset \mathbb{L}' \subset \mathbb{L}$ and $[\mathbb{L}'' : \mathbb{K}] = 2,\ [\mathbb{L}' : \mathbb{L}''] = 3,\ [\mathbb{L} : \mathbb{L}'] = 3$. Therefore we can identify an element in $\mathbb{K}^2$ as an element in $\mathbb{L}' = GF(2^{16}) \subset \mathbb{L}' \subset \mathbb{L}$, an element in $\mathbb{K}^6$ as an element in $\mathbb{L}' = GF(2^{48}) \subset \mathbb{L}$, and an element in $\mathbb{K}^{18}$ as an element in $\mathbb{L} = GF(2^{144})$.

Decompose $(x_1, x_2, \ldots, x_{28}) \in \mathbb{K}^{28}$ into five groups: $X_1 = (x_1, x_2, \ldots, x_8)$, $X_2 = (x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14})$, $X_3 = (x_{15}, x_{16})$, $X_4 = (x_{17}, x_{18}, x_{19})$ and $X_5 = (x_{20}, x_{21}, \ldots, x_{28})$. Identify X_1 with $(0, \ldots, 0, x_1, x_2, \ldots, x_8) \in \mathbb{L}$. Identify $X_2 \in \mathbb{K}^6$ as an element in $\mathbb{L}' \subset \mathbb{L}$. Identify $X_3 \in \mathbb{K}^2$ as an element in $\mathbb{L}'' \subset \mathbb{L}' \subset \mathbb{L}$ and $X_4 \in \mathbb{K}^3$ with $(0, x_{17}, 0, x_{18}, 0, x_{19}) \in \mathbb{L}'' \subset \mathbb{L}$. Identify $X_5 \in \mathbb{K}^9$ with $(0, x_{20}, 0, x_{21}, \ldots, 0, x_{28})$ as an element in $\mathbb{L}$. Hence we have a natural imbedding $i : \mathbb{K}^{28} \hookrightarrow \mathbb{L}^5$ by $i(x_1, x_2, \ldots, x_{28}) = (X_1, X_2, X_3, X_4, X_5)$. Similarly, decompose $(y_9, y_{10}, \ldots, y_{32}) \in \mathbb{K}^{20}$ into four groups: $Y_2 = (y_9, y_{10}, y_{11}, y_{12}, y_{13}, y_{14})$, $Y_3 = (y_{15}, y_{16})$, $Y_4 = (y_{17}, y_{18}, y_{19})$ and $Y_5 = (y_{20}, y_{21}, \ldots, y_{28})$ and identify them as elements in $\mathbb{L}$. For any $r_i \in \mathbb{K}$, $i = 1, 2, \ldots, 8$, identify $R_1 = (r_1, r_2, \ldots, r_8) \in \mathbb{K}^8$ with $(0, \ldots, 0, r_1, r_2, \ldots, r_8) \in \mathbb{L}$. Then we also have

$$i(r_1, r_2, \ldots, r_8, y_9, y_{10}, \ldots, y_{28}) = (R_1, Y_2, Y_3, Y_4, Y_5) \in \mathbb{L}^5.$$

Furthermore, since $\mathbb{K}^{20}$ is a subspace of $\mathbb{L}^5 = \mathbb{K}^{90}$, we have the projection $\pi : \mathbb{L}^5 \to \mathbb{K}^{20}$ such that $(\pi \circ i)(r_1, r_2, \ldots, r_8, y_9, y_{10}, \ldots, y_{28}) = (y_9, y_{10}, \ldots, y_{28})$

Let $\widetilde{\varphi_2} : \mathbb{L}^5 \to \mathbb{L}^5$ be a tractable rational map of the following form.

$$\begin{cases} R_1 = X_1 \\ Y_2 = X_2\ p_2(X_1)\ +\ f_2(X_1) \\ Y_3 = r_3(X_3)\ +\ f_3(X_1, X_2) \\ Y_4 = X_4\ p_4(X_1, X_2, X_3)\ +\ f_4(X_1, X_2, X_3) \\ Y_5 = X_5\ p_5(X_1, X_2, X_3, X_4)\ +\ f_5(X_1, X_2, X_3, X_4) \end{cases}$$

such that $\varphi_2 = \pi \circ \widetilde{\varphi_2} \circ i$, and we have the following in φ_2:

1. $R_1 = X_1$ induces $(r_1, r_2, \ldots, r_8) = (x_1, x_2, \ldots, x_8)$.
2. $Y_2 = X_2\ p_2(X_1)\ +\ f_2(X_1)$ induces

$$\begin{pmatrix} y_9 \\ y_{10} \\ \vdots \\ y_{14} \end{pmatrix} = \begin{pmatrix} x_9 \\ x_{10} \\ \vdots \\ x_{14} \end{pmatrix} *_6 \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_6 \end{pmatrix} + \begin{pmatrix} c_1 x_1 x_2 \\ c_2 x_2 x_3 \\ \vdots \\ c_6 x_6 x_7 \end{pmatrix} + \begin{pmatrix} c_7 x_3 \\ c_8 x_4 \\ \vdots \\ c_{12} x_8 \end{pmatrix}$$

   where c_i's are constant parameters of user's choice and $\mathbf{u} *_n \mathbf{v}$ denotes first identifying $\mathbf{u}, \mathbf{v} \in \mathbb{K}^n$ in the extension field with degree n then carrying out the multiplication there. For details see Appendix.
3. $Y_3 = r_3(X_3)\ +\ f_3(X_1, X_2)$ induces

$$\begin{pmatrix} y_{15} \\ y_{16} \end{pmatrix} = \begin{pmatrix} x_{15} \\ x_{16} \end{pmatrix}^2 + \begin{pmatrix} c_{13} x_1 x_2 + c_{14} x_3 x_4 + \cdots + c_{19} x_{13} x_{14} \\ c_{20} x_{14} x_1 + c_{21} x_2 x_3 + \cdots + c_{26} x_{12} x_{13} \end{pmatrix} + \begin{pmatrix} c_{27} x_1 \\ c_{28} x_2 \end{pmatrix}$$

where $\begin{pmatrix} x_{15} \\ x_{16} \end{pmatrix}^2 = \begin{pmatrix} x_{15} \\ x_{16} \end{pmatrix} *_2 \begin{pmatrix} x_{15} \\ x_{16} \end{pmatrix}$ and c_i's are constant parameters of user's choice.

4. $Y_4 = X_4\ p_4(X_1, X_2, X_3)\ +\ f_4(X_1, X_2, X_3)$ induces
$$\begin{pmatrix} y_{17} \\ y_{18} \\ y_{19} \end{pmatrix} = \begin{pmatrix} x_{17} \\ x_{18} \\ x_{19} \end{pmatrix} *_3 \begin{pmatrix} x_8 \\ x_9 + x_{11} + x_{12} \\ x_{13} + x_{15} + x_{16} \end{pmatrix} + \begin{pmatrix} c_{29}x_4x_{16} \\ c_{30}x_5x_{10} \\ c_{31}x_{15}x_{16} \end{pmatrix} + \begin{pmatrix} c_{32}x_9 \\ c_{33}x_{10} \\ c_{34}x_{11} \end{pmatrix}$$
where c_i's are constant parameters of user's choice.
5. $Y_5 = X_5\ p_5(X_1, X_2, X_3, X_4)\ +\ f_5(X_1, X_2, X_3, X_4)$ induces
$$\begin{pmatrix} y_{20} \\ y_{21} \\ \vdots \\ y_{28} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} x_{19} \\ x_{18} \\ x_{17} \end{pmatrix} & \begin{pmatrix} x_{16} \\ x_{15} \\ x_{14} \end{pmatrix} & \begin{pmatrix} x_{13} \\ x_{12} \\ x_{11} \end{pmatrix} \\ \begin{pmatrix} x_{10} \\ x_9 \\ x_8 \end{pmatrix} & \begin{pmatrix} x_7 \\ x_6 \\ x_5 \end{pmatrix} & \begin{pmatrix} x_4 \\ x_3 \\ x_2 \end{pmatrix} \\ \begin{pmatrix} x_1 \\ x_{19} \\ x_{18} \end{pmatrix} & \begin{pmatrix} x_{17} \\ x_{16} \\ x_{15} \end{pmatrix} & \begin{pmatrix} x_{14} \\ x_{13} \\ x_{12} \end{pmatrix} \end{pmatrix} *_3 \begin{pmatrix} \begin{pmatrix} x_{20} \\ x_{21} \\ x_{22} \end{pmatrix} \\ \begin{pmatrix} x_{23} \\ x_{24} \\ x_{25} \end{pmatrix} \\ \begin{pmatrix} x_{26} \\ x_{27} \\ x_{28} \end{pmatrix} \end{pmatrix} + \begin{pmatrix} c_{35}x_{18}x_{19} \\ c_{36}x_{17}x_{13} \\ c_{37}x_{16}x_{14} \\ c_{38}x_{12}x_{13} \\ c_{39}x_{15}x_{14} \\ c_{40}x_{19}x_{12} \\ c_{41}x_{18}x_{10} \\ c_{42}x_{12}x_6 \\ c_{43}x_{13}x_5 \end{pmatrix} + \begin{pmatrix} c_{44}x_1 \\ c_{45}x_2 \\ \vdots \\ c_{52}x_9 \end{pmatrix}$$
where c_i's are constant parameters of user's choice.

The reason why the formulas in the above assignments represents a permutation polynomial r_3 and polynomials $p_2, f_2, f_3, p_4, f_4, p_5, f_5$ is as follows.

1. We identify $X_3 = (x_{15}, x_{16})$ as an element in $\mathbb{L}'' = GF(2^{16})$ which is of characteristic 2. For any finite field of characteristic 2, $X \mapsto X^2$ is an automorphism. Hence let $r_3(X) = X^2$, then r_3 is an automorphism on $\mathbb{L}''$, hence a permutation polynomial. And $\begin{pmatrix} x_{15} \\ x_{16} \end{pmatrix} \mapsto \begin{pmatrix} x_{15} \\ x_{16} \end{pmatrix}^2$ surely represents r_3.
2. For polynomials p_2, f_2, f_3, p_4, f_4, p_5, f_5, simply notice that on a finite field, any map is a polynomial map. See [14] for details. For example, we show the case of p_2 for illustration. Consider a map $\mathcal{P}$ on $\mathbb{L}$ as follows
$$\mathcal{P}(X_1) = \begin{cases} \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} & \text{if } X_1 = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{pmatrix}, \\ \overrightarrow{0} & \text{otherwise.} \end{cases}$$
Simply let p_2 be the polynomial representation for $\mathcal{P}$.

2.4 The Key Size of TRMS

As shown above, $\varphi_1 = T_1 \circ L_1 \circ D_1 \circ U_1$, $\varphi_3 = T_3 \circ L_3 \circ D_3 \circ U_3$, and there are 52 parameters $c_1, c_2, \ldots, c_{52}$ for the private key user to choose in φ_2. Therefore the size for private key is $[28 + (0 + 1 + \cdots + 27) + 28 + (27 + 26 + \cdots + 0)] + [20 + (0 + 1 + \cdots + 19) + 20 + (19 + 18 + \cdots + 0)] + 52 = 1284$ Bytes.

Also, since the public key is 20 general quadratic polynomials in 28 variables without constant terms, its size is $20 \cdot (\frac{28 \cdot 29}{2} + 28) = 8680$ bytes. In general, there are two ways to generate the public keys.

3 Similar Keys

First, we define the term "Similar Keys" and discuss two transformations of TRMS for similar keys.

Definition 1. *Two public keys of MQ-based PKCs are similar if they are identical after a bijective linear transformation by the public key information in polynomial time. Here the polynomial time should be less than the time to attack the original MQ-based PKC.*

We define some terms for discussing later. Let two public keys of TRMS be $V_P = (\varphi_{P3} \circ \varphi_{P2} \circ \varphi_{P1}) = (p_1, p_2, \cdots, p_m)$ and $V_Q = (\varphi_{Q3} \circ \varphi_{Q2} \circ \varphi_{Q1}) = (q_1, q_2, \cdots, q_m)$,where φ_{P3} and φ_{Q3} are invertible affine transformations over $\mathbb{K}^m$, φ_{P1} and φ_{Q1} are invertible affine transformations over $\mathbb{K}^n$, φ_{P2} and φ_{Q2} are projections $\mathbb{K}^n \rightarrow \mathbb{K}^m$, and $p_1, p_2, \cdots, p_m, q_1, q_2, \cdots, q_m$ are quadratic polynomials in n variables without constant terms.

3.1 Invertible Linear Transformation of φ_3

If $q_1, q_2, \cdots, q_m$ could be expressed as linear combinations of V_P. And $p_1, p_2, \cdots, p_m$ could be expressed as linear combinations of V_Q. Then V_P and V_Q are similar. More precisely, $V_P = L \circ V_Q$ and L is an invertible linear transformation.

Lemma 2. *If $\varphi_{P1} = \varphi_{Q1}$ and $\varphi_{P2} = \varphi_{Q2}$. Then V_P and V_Q are similar keys.*

Proof. $V_P = \varphi_{P3} \circ \varphi_{P2} \circ \varphi_{P1}$ and $V_Q = \varphi_{Q3} \circ \varphi_{Q2} \circ \varphi_{Q1}$. Since φ_{P3} and φ_{Q3} are invertible transformations, there exists φ_{P3}^{-1}, the inverse of φ_{P3}, and φ_{Q3}^{-1}, the inverse of φ_{Q3}. Hence $\varphi_{P3}^{-1} \circ \varphi_{P3} = \varphi_{Q3}^{-1} \circ \varphi_{Q3} = I_m$.

$$\begin{aligned}
\varphi_{P3} &= \varphi_{P3} \circ I_m \\
&= \varphi_{P3} \circ \varphi_{P3}^{-1} \circ \varphi_{P3} \\
&= \varphi_{P3} \circ \varphi_{Q3}^{-1} \circ \varphi_{Q3} \\
V_P &= \varphi_{P3} \circ \varphi_{P2} \circ \varphi_{P1} \\
&= \varphi_{P3} \circ \varphi_{P3}^{-1} \circ \varphi_{P3} \circ \varphi_{P2} \circ \varphi_{P1} \\
&= \varphi_{P3} \circ \varphi_{Q3}^{-1} \circ \varphi_{Q3} \circ \varphi_{Q2} \circ \varphi_{Q1} \\
&= \varphi_{P3} \circ \varphi_{Q3}^{-1} \circ V_Q
\end{aligned}$$

Then we get the equations $V_P = L \circ V_Q$, where $L = \varphi_{P3} \circ \varphi_{Q3}^{-1}$ is an invertible linear transformation.

Since p_i is a linear combination of V_Q, then we get the equation $l_{i,1}q_1 + l_{i,2}q_2 + \cdots + l_{i,m}q_m + p_i = 0$, for $i \in (1, 2, \cdots, m)$ and $l_{i,j}$ is the element of L in i-row and j-column. It is easy to solve $(l_{i,1}, l_{i,2}, \cdots, l_{i,m})$. We could get L by solving $\frac{n\cdot(n+3)}{2}$ equation in (m+1) variables in time complexity $O(m^2n^2)$.

3.2 Substitution of φ_1

Let R be a random permutation of X, $X = (x_1, x_2, \cdots, x_n)$. If $p_1(X) = q_1(R(X))$, $p_2(X) = q_2(R(X)), \cdots, p_m(X) = q_m(R(X))$. Then V_P and V_Q are similar.

Lemma 3. *If $\varphi_{P2} = \varphi_{Q2}$, $\varphi_{P3} = \varphi_{Q3}$ and $\varphi_{P1} = \varphi_{Q1} \circ A$, where A is a permutation matrix. Then V_P and V_Q are similar keys.*

Proof. As our definition,

$$V_P = \varphi_{P3} \circ \varphi_{P2} \circ \varphi_{P1} = \varphi_{Q3} \circ \varphi_{Q2} \circ \varphi_{Q1} \circ A = V_Q \circ A.$$

If A could be computed by V_P and V_Q, then V_P and V_Q are similar. Let

$$X_{i,v} = (x_1, x_2, \cdots, x_n) \text{ ,where } x_j = v \text{ if } j = i, \text{ and } x_j = 0 \text{ if } j \neq i.$$

For example $X_{1,1} = (1, 0, 0, \cdots, 0), X_{2,3} = (0, 3, 0, \cdots, 0)$. Then we evaluate $V_P(X_{1,1}), V_P(X_{2,1}), \cdots, V_P(X_{n,1})$ and $V_Q(X_{1,1}), V_Q(X_{2,1}), \cdots, V_Q(X_{n,1})$. Since $V_P = V_Q \circ A$ and A is a permutation matrix, we have $V_P(X_{i,1}) = V_Q(A(X_{i,1})) = V_Q(X_{j,1})$, where $j \in (1, 2, \cdots, n)$.

If the values of $V_P(X_{i,1})$ are all different, we could find the mapping of $A(X_{i,1}) = X_{j,1}$ for all $i \in (1, 2, \cdots, n)$.

If $V_P(X_{i,1})$ and $V_P(X_{j,1})$ are equivalent, we evaluate $V_P(X_{i,2}), V_P(X_{j,2})$ and $V_Q(X_{i,2}), V_Q(X_{j,2})$ to find the mapping of $A(X_{i,2}) = X_{k,2}$. If $V_P(X_{i,2})$ and $V_P(X_{j,2})$ are still equivalent, we change $X_{i,2}$ to $X_{i,3}$ and continue to get the permutation matrix A.

The probability of $V_P(X_{i,v}) = V_P(X_{j,v})$ is $\frac{1}{|K|^{m-n}} = \frac{1}{256^{(28-20)}} = 2^{-64}$. The probability of $V_P(X_{i,1}) = V_P(X_{j,1})$ and $V_P(X_{i,2}) = V_P(X_{j,2})$ is 2^{-128}. It is so small that we could ignore this happens. So A could be computed in the time complexity $O(m)$.

4 Our Scheme and Performance

We use Lemma 2 to transform the model of TRMS or other MQ-based PKCs. If we fix φ_2 and φ_1, we will get the similar key no matter φ_3 we choose. But we still need φ_3 to mask the equations of $\varphi_2 \circ \varphi_1$.

4.1 Our Scheme

When we generate the key pair of TRMS as Section 2, we add a new affine transformation φ_4 such that $V'_P = \varphi_{P4} \circ \varphi_{P3} \circ \varphi_{P2} \circ \varphi_{P1} = (p'_1, p'_2, \cdots, p'_m)$ has a

$$
\begin{array}{rcccccccccc}
p_i'(X) = & a_{11}x_1^2 & + & a_{12}x_1x_2 & + \cdots + & a_{1i}x_1x_i & + \cdots + & a_{1n}x_1x_n \\
& & + & a_{22}x_2^2 & + \cdots + & a_{2i}x_2x_i & + \cdots + & a_{2n}x_2x_n \\
& & & & & \vdots & & \\
& & & & & & + & a_{nn}p_np_n \\
+ & 0x_1 & + & 0x_2 & + \cdots + & x_i & + \cdots + & 0x_n
\end{array}
$$

Fig. 1. One example of our new model

special form like Fig.1. If the coefficient of x_i in p_i' is zero, we generate φ_1 again. There are many choices of the above form of p_i'. We just give one example to illustrate.

From Lemma 2, any choice of φ_3 has the unique φ_4 to form the specific polynomial set. And $\varphi_4 \circ \varphi_3$ is also an affine transformation. The new model is still one of TRMS (or the original MQ-based PKC). Actually we do not have to generate and save φ_3 and φ_4, as $\varphi_4 \circ \varphi_3$ is unique. We generate $\varphi_2 \circ \varphi_1$ first, and then use the Gaussian elimination to find the polynomials in Fig.1.

Public Key. The public key is the polynomials of V_P', and some terms in public key are always zero so that we do not need to store these values and to compute when encrypting or verifying.

Private Key. The private key for the new model of TRMS is only φ_2 and φ_1^{-1}.

Signing and Verifying. When signing and verifying, we do the same steps of [15], except that when signing, we first read the private key and find $\varphi_4 \circ \varphi_3$. This overhead is the key expansion. This overhead is first introduced in MQ-based PKCs, but it is quite general in symmetric key cryptosystems, like AES, DES. The signing time of the new model equals the original model, and for some zero monomials, the verifying time should be a little faster than the original model.

4.2 Performance

Key Size. The key size of private key in our model is only 50% $\sim$ 70% of the one in the original model. We apply our model to other MQ-based PKCs. The result is in Table 1.

Table 1. Private key reduction ratio of MQ-based PKCs

Scheme Name	Original model (Bytes)	Our model (Bytes)	Reduction ratio
TRMS(20,28)[1]	396	276	69.7%
TRMS(20,28)[2]	1284	864	67.3%
TTS(20,28)	1399	979	70.0%
SFLASH^{v2}	2450	1225	50.0%
SQARTZ	3914	2575	65.8%

[1] computed with the detail in [15]

[2] computed with the detail in Section 2

Execution Time. We wrote code to test the execution time of our model and the original model of TRMS in [15]. The result is in Table 2. The environment is that CPU: P4 2.4GHz, RAM: 1024MB, OS: Linux + gcc 3.3, and parameters: gcc -O3 -march=pentium4 -fomit-frame-pointer.

Table 2. Execution time of our model and the original model of TRMS

Operation	Original model	Our model
Generating Key (ms)	1.3	0.9
Setting Key (ms)	x	0.1
Signing (ns)	7	7
Verifying (ns)	20	20

5 Discussion

5.1 Remark on Security

As we mentioned in Section 4, the public key of our new scheme is $\varphi_{P4}\circ\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}$ and $\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}$ is the original model. φ_4 is the unique transformation that is computable by the information of $\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}$. Here we motivate that the security of the new model and the original model are equivalent.

We assume that there is a method A that can make a fake signature of our new model of TRMS (or other MQ-based signature). Hence we can get a fake signature of the original model.

Since φ_4 can be computed from $\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}$ in time complexity $O(m^2n^2)$ and A could make a fake signature with $\varphi_{P4}\circ\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}$, for any $H\in\mathbb{K}^m$. Then we get a signature S, that satisfies $H=\varphi_{P4}\circ\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}(S)$.

If we want to make a fake signature with $\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}$, we get the hash value $H_{original}$ first. Then we evaluate the $\varphi_4(H_{original})$. Then we apply A to compute $S_{original}$ such that $\varphi_4(H_{original})=\varphi_{P4}\circ\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}(S_{original})$. And φ_{P4} is an invertible affine transformation. Then we get $S_{original}$ is the signature of $H_{original}$.

$$\begin{aligned} H_{original} &= \varphi_{P4}^{-1}\circ\varphi_{P4}(H_{original})\\ &= \varphi_{P4}^{-1}\circ\varphi_{P4}\circ\varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}(S_{original})\\ &= \varphi_{P3}\circ\varphi_{P2}\circ\varphi_{P1}(S_{original}). \end{aligned}$$

The other direction is easy to understand as the new model of TRMS is one of the original model. If a method A' can attack the original model, hence A' can attack the new model.

5.2 Key Generation

The key generation time is faster than the original model is reasonable. The difference between these models is φ_3. The original model needs to generate φ_3 and the inverse of φ_3. φ_3 needs a lot of random numbers and the inverse of φ_3 needs the Gaussian elimination. Though our model needs φ_3 and φ_4, $\varphi_4\circ\varphi_3$

is from the Gaussian elimination of $\varphi_2 \circ \varphi_1$. The Gaussian elimination takes additive and multiplicative operations in the finite field. These operations take less time than the random number generations.

5.3 Polynomial Forms

In our experiment, the probability to generate key successfully is $\frac{1000}{1094\sim1108} \approx 0.9$. Some reviewer surprised at this result. As φ_1, φ_3, φ_4 are all invertible, it must be possible to map the linear terms to the identity matrix. The reason is φ_2 is quadratic. Then the linear terms in public key are not only from the linear terms in φ_2 and φ_1 but also the quadratic terms in φ_2 and the linear terms and constant terms in φ_1.

Some reviewer suggests to restrict both φ_1 and φ_3 to linear transformations and then we always generate key successfully. This is an interesting idea for HFE-like or UOV-like schemes. However, it is not applicable to the current version of TRMS as there are only 11 linear terms in φ_2.

$$\begin{bmatrix} p_1(X) \\ p_2(X) \\ \vdots \\ p_m(X) \end{bmatrix} = \begin{bmatrix} a_{111} & \cdots & a_{11n} & a_{122} & \cdots & a_{12n} & \cdots & a_{1nn} & b_{11} & \cdots & b_{1n} \\ a_{211} & \cdots & a_{21n} & a_{222} & \cdots & a_{22n} & \cdots & a_{2nn} & b_{21} & \cdots & b_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m11} & \cdots & a_{m1n} & a_{m22} & \cdots & a_{m2n} & \cdots & a_{mnn} & b_{m1} & \cdots & b_{mn} \end{bmatrix} \begin{bmatrix} x_1^2 \\ \vdots \\ x_1x_n \\ x_2^2 \\ \vdots \\ x_2x_n \\ \vdots \\ p_np_n \\ x_1 \\ \vdots \\ x_n \end{bmatrix}$$

Fig. 2. The matrix form of the public key

The polynomial form in Fig.1 is an example. In order to always generate key successfully, we can change the polynomial form. If $(p_1, p_2, \cdots, p_m)$ is the public key. The public key can be represented in the matrix form like Fig.2. We perform elementary row operations to find the rank of the coefficient matrix and make the first non-zero term in each row to 1. The final coefficient matrix is the new public key. As the polynomial form in Fig.1, the new public key is unique if the original public key is similar. When saving the public key, we first save the index of the first non-zero term in each row and then save the subsequent terms.

6 Conclusion

Wolf and Preneel [16] showed multivariate quadratic public key schemes have many equivalent keys. In this paper, we introduce the idea of similar keys of

MQ-based PKCs and utilize the idea for the new model of some MQ-based PKCs. And this new model could reduce the size of private key to 50% $\sim$ 70%. Moreover, our model remains the equivalent security and has a little advantage of public key in size and verifying time.

We introduce two transformations to find the similar key in affine transformation, and we only apply Lemma 2 to reduce the size of the private key. The methods to compute the number of the similar keys of the MQ-based PKCs are not only these two transformations we provide. We will survey the others in the future. And there may be other methods to reduce the key size with Lemma 3 or new transformations.

In this paper, we concentrate on the two affine transformations for similar keys. There is another way to find the similar keys of a particular MQ-based PKC, like HFE, TTS. That is to utilize the kernel information, φ_2, to find the similar keys and reduce the private key space.

Finally, our new model is general since most MQ-based PKCs are composed of $(\varphi_1,\varphi_2,\varphi_3)$. There are two kinds of exceptions, STS schemes and UOV-like schemes. STS schemes have a little reduction than others as there are many coefficients in φ_2. UOV-like schemes are composed of (φ_1,φ_2). But all still could enjoy the advantage of the size of public key and encrypting time/verifying time.

Acknowledgements

The authors would like to thank Dr. Christopher Wolf and Dr. Bart Preneel for the fruitful discussions.

References

1. Jiun-Ming Chen and Bo-Yin Yang, *A More Secure and Efficacious TTS Scheme*, ICISC 2003, LNCS v. 2971, pp. 320-338, full version at http://eprint.iacr.org/2003/160.
2. N. Courtois, L. Goubin, and J. Patarin, *SFLASH*^{v3}*, a Fast Asymmetric Signature Scheme*, eprint 2003/211, available at http://eprint.iacr.org/2003/211.
3. M. Garey and D. Johnson, *Computers and Intractability, A Guide to the Theory of NP-completeness*, 1979, p. 251.
4. Yuh-Hua Hu, Lih-Chung Wang, Jiun-Ming Chen, Feipei Lai, and Chun-Yen Chou, *An implementation of public key cryptosystem TTM with linear time complexity for decryption*, Proceedings. IEEE International Symposium on Information Theory 2003, pp. 17.
5. Neal Koblitz, *Elliptic curve cryptosystems*, Mathematics of Computation 48 (1987), pp.203-209.
6. A. Kipnis, J. Patarin, and L. Goubin, *Unbalanced Oil and Vinegar Sigature Schemes*, CRYPTO 1999, LNCS v. 1592, pp. 206-222.
7. V.S. Miller, *Use of elliptic curves in cryptography*, CRYPTO 1985, LNCS v. 218, pp. 417-426.
8. Tsutomu Matsumoto and Hideki Imai, *Public Quadratic Polynomial-Tuples for Efficient Signature-Verification and Message-Encryption*, EUROCRYPT 1988, LNCS v. 330, pp. 419-453.

9. *New European Schemes for Signatures, Integrity, and Encryption*, project homepage at http://www.cryptonessie.org.
10. *Performance of Optimized Implementations of the NESSIE primitives, version 2.0* http://www.cryptonessie.org.
11. J. Patarin, *Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP) Two New Families of Asymmetric Algorithms*, EUROCRYPT 1996, LNCS v. 1070, pp. 33-48.
12. J. Patarin, N. Courtois, and L. Goubin, *QUARTZ, 128-Bit Long Digital Signatures*, CT-RSA 2001, LNCS v. 2020, pp. 282-297. Updated version available at http://www.cryptonessie.org.
13. http://www.am.ndhu.edu.tw/ lcwang/lcwang.htm.
14. Lih-Chung Wang and Fei-Hwang Chang, *Tractable Rational Map Cryptosystem*, eprint 2004/046, available at http://eprint.iacr.org/2004/046.
15. Lih-Chung Wang, Yuh-Hua Hu, Bo-Yin Yang, Feipei Lai, Chun-Yen Chou and Bo-Yin Yang, *Tractable Rational Map Signature*, PKC 2005, LNCS v. 3386 pp. 244-257.
16. Christopher Wolf and Bart Preneel, *Large Superfluous Keys in Multivariate Quadratic Asymmetric Systems*, PKC 2005, LNCS v. 3386 pp. 275-287.
17. Christopher Wolf and Bart Preneel, *Taxonomy of Public Key Schemes based on the problem of Multivariate Quadratic equations*, eprint 2005/077, available at http://eprint.iacr.org/2005/077.
18. William Stallings, *CRYPTOGRAPHY AND NEWWORK SECURITY Principles and Practice*, Second Edition, pp. 356.
19. Bo-Yin Yang, Jiun-Ming Chen, and Yen-Hung Chen, *TTS: High-Speed Signatures on a Low-End Smart Card*, CHES 2004, LNCS v. 3156, pp. 371-385.