Embedded Surface Attack on Multivariate Public Key Cryptosystems from Diophantine Equations

Jintai Ding[1,2], Ai Ren[2], and Chengdong Tao[3]

[1] Chongqing University
[2] University of Cincinnati
[3] South China University of Technology
{jintai.ding,chengdongtao2010}@gmail.com, renai@mail.uc.edu

Abstract. Let $X = (x_1, .., x_n)$ and $Y = (y_1, ..., y_m)$ be a pair of corresponding plaintext and ciphertext for a cryptosystem. We define an embedded surface of this cryptosystem as any polynomial equation:

$$E(X, Y) = E(x_1, .., x_n, y_1, ..., y_m) = 0,$$

which is satisfied by all such pairs. In this paper, we present a new attack on the multivariate public key cryptosystems from Diophantine equations developed by Gao and Heindl by using the embedded surfaces associated to this family of multivariate cryptosystems.

1 Introduction

The security of cryptosystems such as RSA, ECC, and Diffie-Hellman key exchange scheme, depends on assumptions about the hardness of certain number theoretic problems, such as the Integer Prime Factorization Problem or the Discrete Logarithm Problem. However, in 1994 Peter Shor [12] showed that quantum computers could break all public key cryptosystems that are based on these hard number theoretic problems. In recent years, significant efforts have been devoted to the search for alternative public key cryptosystems, which would remain secure in an era of quantum computers. Multivariate public key cryptosystems (MPKC) are one of the main families of cryptosystems that have the potential to resist quantum computer attacks.

The public key of a MPKC is a system of multivariate polynomials, mostly quadratic, over a finite field. This construction is based on the fact that solving a random multivariate polynomial system over a finite field is an NP-complete problem. In general, MPKCs have the following structure. Let k be a finite field with q elements. A public key is a map

$$\bar{F} : k^n \to k^m$$

constructed as:

$$\bar{F} = L_1 \circ F \circ L_2,$$

M. Kutyłowski and M. Yung (Eds.): Inscrypt 2012, LNCS 7763, pp. 122–136, 2013.

where L_1 and L_2 are are two random invertible affine transformations over k^m and k^n respectively. The central map $F : k^n \to k^m$ is a nonlinear multivariate polynomial map that has the property of being easily invertible computation-wise. The key of building a good MPKC is to find a good polynomial system F that makes the cryptosystem secure.

There are many attempts in building MPKC for encryption. For MPKCs, the encryption schemes standing in general are much slower than the signature schemes. Therefore, there is still a need to find good constructions for MPKCs for encryption.

In 2009, a new construction was built using a very different idea from before, namely one uses a special function solution to certain special Diophantine equation [8][7]. Even though, this construction also uses a triangular construction, Oil-Vinegar construction[11], the key component comes from certain special solutions of the Diophantine equation:

$$AB = CD + EF + GH + IJ + KL,$$

which is inspired by the construction in MFE[13], a generalization of HFE[10].

For this new family, the authors [7] propose three concrete cases for practical applications. They have very strong security claims, which is at the level of 2^{113} or higher. Also the decryption process is very efficient. By now, no one could yet find any weakness in the system.

1.1 The Contributions of This Paper

Let $X = (x_1, .., x_n)$ and $Y = (y_1, ..., y_m)$ be a pair of corresponding plaintext and ciphertext for any cryptosystem. We define an embedded surface of this cryptosystem as any polynomial equation:

$$E(X, Y) = E(x_1, .., x_n, y_1, ..., y_m) = 0,$$

which is satisfied by all such pairs. This name comes from the fact that this equation defines an algebraic surface in the space k^{m+n}, where all the plaintext and ciphertext pairs for this cryptosystems belong, or we can embed all the such pairs in such a surface.

In the case of MPKCs, the public key equation itself: $y_i = \bar{f}_i(x_1, ..., x_n)$, gives an embedded surface in k^{m+n}. In terms of attack on cryptosystems, one general approach should try to find the embedded surfaces that could help us to attack the systems. We call such an embedded surface a non-trivial embedded surface.

The first non-trivial embedded surface is used as the linerization equation by Patarin to defeat the Matsumomto-Imai MPKCs[10]:

$$\sum a_{i,j} x_i y_j + \sum b_i y_i + \sum c_i x_i + e = 0.$$

If we know the value of a ciphertext, this equation will give us a linear equation satisfied by the plaintext, which is very useful in attacking the system. This is also the idea used to build algebraic attack on AES.

Later, another embedded surface, a high order linerization equation is used in breaking the MFE MPKCs[3].

A natural question is:

can more general embedded surfaces other than linearization type of equations be useful to attack cryptosystems?

What we did in this paper is to give a positive answer to this question by using it to attack the new MPKCs from Diophantine equations.

What we observe is that, in the new MPKCs using the Diophantine equations, the decryption process actually implies that we can use the embedded surfaces to get what is done in the decryption process. Namely special embedded surfaces will help us to decrypt the message efficiently. In this case, the corresponding embedded surfaces will be in the form:

$$\begin{gathered}\sum A_{i,j,s,t}y_iy_jy_sy_t + \sum B_{i,j,s}y_iy_jy_s + \sum C_{i,j}y_iy_j + \\ \sum D_jy_j + \sum A'_{i,j,s,t}y_iy_jx_s^2x_t^2 + \sum B'_{i,s,t}y_ix_s^2x_t^2 + \sum C'_{i,j,s}y_iy_jx_s^2 + \\ \sum D'_{i,s}y_ix_s^2 + \sum E'_{s,t}x_s^2x_t^2 + \sum H'_sx_s^2 + E = 0. \end{gathered} \tag{1}$$

The embedded surfaces will produce equations in the form:

$$\sum A''_{s,t}x_s^2x_t^2 + \sum B''_sx_s^2 + C'' = 0, \tag{2}$$

once the values of the ciphertext y_i are given. This enables us to derive new quadratic equations due to the fact that the field is of characteristic 2. This allows us to break the system efficiently. We could break the three systems proposed at the complexity of 2^{52}, 2^{61} and 2^{52} over the corresponding fields respectively.

This paper is organized as follows. In Section 2, we introduce the new MPKCs from Diophantine equations. In Section 3, we will present the cryptanalysis of the new MPKCs by using embedded surfaces. We conclude in Section 4.

2 Multivariate Public Key Cryptosystems from Diophantine Equations

In this section, we will present the MPKCs from Diophantine equations, and we will follow the notations in[7].

Let k be a finite field with q elements, and let $\mathcal{F}$ be an extension of k with degree d. In an MFE type ("medium field") construction, we fix a basis $\alpha_1, ..., \alpha_d$ of $\mathcal{F}$ over k, which identifies $\mathcal{F}$ with k^d via the natural map p : $\mathcal{F} \to k^d$:

$$p(a_1\alpha_1 + \ldots + a_d\alpha_d) = (a_1, ..., a_d). \tag{3}$$

Then we view a polynomial $f \in \mathcal{F}[X_1, ..., X_n]$ component-wise over k by writing

$$X_i = x_{i1}\alpha_1 + \ldots + x_{id}\alpha_d,$$

and then

$$f = f_1\alpha_1 + \ldots + f_d\alpha_d,$$

with $f_i \in k[x_{11}, ..., x_{nd}]$.

Throughout this paper, we assume that the finite field $\mathcal{F}$ has characteristic two. If the field is $GF(2)$, the embedded surfaces of (1) and (2) should be modified slightly, since there are not square terms.

2.1 The Origin of the Diophantine Equations

In the MFE MPKC, the key idea comes the fact that

$$\mathrm{d}et(M_1M_2) = \mathrm{d}et(M_1) \times \mathrm{d}et(M_2),$$

for two 2 × 2 matrices:

$$M_1 = \begin{pmatrix} X_1 & X_2 \\ X_3 & X_4 \end{pmatrix}, \quad M_2 = \begin{pmatrix} X_5 & X_6 \\ X_7 & X_8 \end{pmatrix}.$$

This gives a quadratic polynomial solution to the the Diophantine equation over a polynomial ring $\mathcal{F}[X_1, ..., X_8]$:

$$AB = CD + EF,$$

namely

$$\begin{aligned}(X_1X_4 + X_2X_3)(X_5X_8 + X_6X_7) = (X_1X_5 + X_2X_7)(X_3X_6 + X_4X_8) + \\ (X_1X_6 + X_2X_8)(X_3X_5 + X_4X_7).\end{aligned}$$

To build new type of MPKCs, Gao and Heindl were able to find solutions to the following new Diophantine equation:

$$AB = CD + EF + GH + IJ + KL,$$

over the ring $\mathcal{F}[X_1, ..., X_8, Y_1, ..., Y_8]$. In the context of their work, they rewrite this equation as

$$\psi_1\psi_2 = f_1f_2 + ... + f_9f_{10}, \tag{4}$$

where each polynomial is quadratic and

- $\psi_1 \in \mathcal{F}[X_1, ..., X_8], \psi_2 \in \mathcal{F}[Y_1, ..., Y_n]$;
- $f_i \in \mathcal{F}[X_1, ..., X_8, Y_1, ..., Y_8]$, for $0 < i < 9$, are oil-vinegar polynomials;
- $f_i \in \mathcal{F}[X_1, ..., X_8, Y_1, ..., Y_8]$, i = 9, 10.

An oil-vinegar polynomial is a quadratic polynomial, where we divide the variables into two sets: the oil variables and the vinegar variables, and an oil-vinegar polynomial has not any quadratic terms with only oil variables:

$$\sum a_{ij}x_ix'_j + \sum b_{ij}x'_ix'_j + \sum c_jx_i + \sum d_jx'_j + e = 0,$$

where x_i are oil variables and x'_j are the vinegar variables.

The design starts with the polynomial ring

$$R = \mathcal{F}[x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4, z_1, z_2, z_3, z_4, w_1, w_2, w_3, w_4].$$

Let

$$p^{ij}_{xy} = x_i y_j + x_j y_i,$$

for $1 = i < j = 4$;

$$p^{ij}(x, y, z, w) = p^{ij}_{xz} + p^{ij}_{yz} + p^{ij}_{yw},$$

for $1 = i < j = 4$.

In algebraic geometry terms, the p^{ij}_{xy} are simply Plück coordinates, which are known to satisfy the quadratic relations:

$$\begin{aligned} 0 = (p^{12}_{xy} + p^{12}_{zw})p^{34}(x, y, z, w) + (p^{13}_{xy} + p^{13}_{zw})p^{24}(x, y, z, w) + \\ (p^{14}_{xy} + p^{14}_{zw})p^{23}(x, y, z, w) + (p^{23}_{xy} + p^{23}_{zw})p^{14}(x, y, z, w) + \\ (p^{24}_{xy} + p^{24}_{zw})p^{13}(x, y, z, w) + (p^{34}_{xy} + p^{34}_{zw})p^{12}(x, y, z, w). \end{aligned} \tag{5}$$

2.2 The Central Map

Let ρ be a ring homomorphism from R to $\mathcal{F}[X_1, ..., X_8, Y_1, ..., Y_8]$ induced by the map:

$$\begin{aligned} (x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4, z_1, z_2, z_3, z_4, w_1, w_2, w_3, w_4) \rightarrow \\ (X_1, X_3, Y_1 + Y_5, Y_3 + Y_7, X_4, X_2, Y_5, Y_7, X_5, X_7, Y_4 + Y_8, Y_2 + Y_6, X_8, X_6, Y_8, Y_6). \end{aligned}$$

Let

$$\begin{aligned}
\psi_1 &= \rho(p^{12}_{xy} + p^{12}_{zw}) = X_1X_2 + X_3X_4 + X_5X_6 + X_7X_8 \\
\psi_2 &= \rho(p^{34}(x, y, z, w)) = Y_1Y_2 + Y_3Y_4 + Y_5Y_6 + Y_7Y_8 \\
f_1 &= \rho(p^{13}_{xy} + p^{13}_{zw}) = X_4Y_1 + X_8Y_4 + (X_1 + X_4)Y_5 + X_5Y_8 \\
f_2 &= \rho(p^{24}(x, y, z, w)) = (X_2 + X_3)Y_2 + X_7Y_3 + X_2Y_6 + X_6Y_7 \\
f_3 &= \rho(p^{14}_{xy} + p^{14}_{zw}) = X_8Y_2 + X_4Y_3 + X_5Y_6 + (X_1 + X_4)Y_7 \\
f_4 &= \rho(p^{23}(x, y, z, w)) = X_7Y_1 + (X_2 + X_3)Y_4 + X_6Y_5 + X_2Y_8 \\
f_5 &= \rho(p^{23}_{xy} + p^{23}_{zw}) = X_2Y_1 + X_6Y_4 + (X_2 + X_3)Y_5 + X_7Y_8 \\
f_6 &= \rho(p^{14}(x, y, z, w)) = (X_1 + X_4)Y_2 + X_5Y_3 + X_4Y_6 + X_8Y_7 \\
f_7 &= \rho(p^{24}_{xy} + p^{24}_{zw}) = X_6Y_2 + X_2Y_3 + X_7Y_6 + (X_2 + X_3)Y_7 \\
f_8 &= \rho(p^{13}(x, y, z, w)) = X_5Y_1 + (X_1 + X_4)Y_4 + X_8Y_5 + X_4Y_8 \\
f_9 &= \rho(p^{34}_{xy} + p^{34}_{zw}) = Y_1Y_7 + Y_2Y_8 + Y_3Y_5 + Y_4Y_6 \\
f_{10} &= \rho(p^{12}(x, y, z, w)) = X_1X_7 + X_2(X_5 + X_8) + X_3X_5 + X_4(X_6 + X_7)
\end{aligned}$$

Note that $f_1, ..., f_8$ are Oil-vinegar polynomials, where we can take either $X = (X_1, ..., X_8)$ or $Y = (Y_1, ..., Y_8)$ be the vinegar variables. This implies that, if either X or Y is known, we can use the polynomial equations coming from knowing the value of these polynomials to find the value of the other one by plugging in the values of variables given.

In terms of the original notation, we rename each ψ_j and f_j above as $\psi_{1,j}$ and $f_{1,j}$ respectively, and define $\psi_{i,1} = \psi_{1,1}$, and $f_{i,j} = f_{1,j}$, for i = 2, 3, 4, j = 1, 3, 5, 7, 9.

Again, in terms of the original notation, interchanging z with w in (5), we define

$$\begin{aligned} \psi_{2,2} &= \rho(p^{34}(x,y,w,z)), \\ f_{2,2} &= \rho(p^{24}(x,y,w,z)), \\ f_{2,4} &= \rho(p^{23}(x,y,w,z)), \\ f_{2,6} &= \rho(p^{14}(x,y,w,z)), \\ f_{2,8} &= \rho(p^{13}(x,y,w,z)), \\ f_{2,10} &= \rho(p^{12}(x,y,w,z)). \end{aligned} \tag{6}$$

Similarly, by interchanging x with y in (5), we define $\psi_{3,2}$ and $f_{3,j}$, for j = 2, 4, 6, 8, 10; by interchanging x with y, and z with w in (5), we define $\psi_{4,2}$ and $f_{4,j}$, for j = 2, 4, 6, 8, 10. Then we have four identities:

$$\psi_{i,1}\psi_{i,2} = f_{i,1}f_{i,2} + \dots f_{i,9}f_{i,10}, \tag{7}$$

for $0 < i < 5$.

The central map:

$$(Z_1, ..., Z_{74}) = F(X_1, ..., X_{24}, Y_1, ..., Y_{32}),$$

is defined as

$$\begin{aligned} Z_1 =& \ X_1 + \psi_{1,1}(X_1, ..., X_8) + \phi_1(X_1) \\ Z_2 =& \ X_2 + \psi_{1,2}(Y_1, ..., Y_8) + \phi_2(X_1, X_2) \\ Z_3 =& \ X_3 + \psi_{2,2}(Y_9, ..., Y_{16}) + \phi_3(X_1, X_2, X_3) \\ Z_4 =& \ X_4 + \psi_{3,2}(Y_{17}, ..., Y_{24}) + \phi_4(X_1, X_2, X_3, X_4) \\ Z_5 =& \ X_5 + \psi_{2,1}(X_9, ..., X_{16}) + \phi_5(X_1, X_2, X_3, X_4, X_5) \\ Z_6 =& \ X_6 + \psi_{3,1}(X_{17}, ..., X_{24}) + \phi_6(X_1, X_2, X_3, X_4, X_5, X_6) \\ Z_7 =& \ X_7 + \psi_{4,2}(Y_{25}, ..., Y_{32}) + \phi_7(X_1, X_2, X_3, X_4, X_5, X_6, X_7) \\ Z_{7+i} =& \ f_{1,i}(X_1, ..., X_8, Y_1, ..., Y_8) \ \ i = 1, .., 10 \\ Z_{17+i} =& \ f_{2,i}(X_1, ..., X_8, Y_9, ..., Y_{16}) \ \ i = 1, .., 10 \\ Z_{27+i} =& \ f_{2,i}(Y_1, ..., Y_8, Y_9, ..., Y_{16}) \ \ i = 1, .., 8 \\ Z_{36} =& \ f_{2,10}(Y_1, ..., Y_8, Y_9, ..., Y_{16}) \\ Z_{36+i} =& \ f_{3,i}(X_1, ..., X_8, Y_{17}, ..., Y_24) \ \ i = 1, .., 10 \\ Z_{46+i} =& \ f_{2,i}(X_9, ..., X_{16}, Y_9, ..., Y_{16}) \ \ i = 1, .., 8 \\ Z_{55} =& \ f_{2,10}(X_9, ..., X_{16}, Y_9, ..., Y_{16}) \\ Z_{55+i} =& \ f_{3,i}(X_{17}, ..., X_{24}, Y_{17}, ..., Y_{24}) \ \ i = 1, .., 8 \\ Z_{64} =& \ f_{3,10}(X_{17}, ..., X_{24}, Y_{17}, ..., Y_{24}) \\ Z_{64+i} =& \ f_{4,i}(X_9, ..., X_{16}, Y_{25}, ..., Y_{32}) \ \ i = 1, .., 10 \end{aligned}$$

Here ϕ_i are randomly chosen quadratic functions.

Note that $f_{2,9}(Y_1, ..., Y_8, Y_9, ..., Y_{16})$ is not explicitly in the central map due to the redundancy that

$$f_{2,9}(Y_1, ..., Y_8, Y_9, ..., Y_{16}) = f_{2,9}(X_1, ..., X_8, Y_9, ..., Y_{16}) = Z_{26}. \tag{8}$$

Similar situation is also true for:
$f_{2,9}(X_9, ..., X_{16}, Y_9, ..., Y_{16})$ and $f_{3,9}(X_{17}, ..., X_{24}, Y_{17}, ..., Y_{24})$.

There are 7 oil-vinegar polynomial systems inside the central map:

1. $Z_8, ..., Z_{15}$, where $Y_1, ..., Y_8$ are viewed as oil variable and $X_1, ..., X_8$ are viewed as vinegar variables;
2. $Z_{18}, ..., Z_{25}$, where $Y_9, ..., Y_{16}$ are viewed as oil variable and $X_1, ..., X_8$ are viewed as vinegar variables;
3. $Z_{28}, ..., Z_{35}$, where $Y_1, ..., Y_8$ can be viewed either as oil or vinegar variables and $Y_9, ..., Y_{16}$ are viewed as the opposite variables;
4. $Z_{37}, ..., Z_{44}$, where $Y_{17}, ..., Y_{24}$ are viewed as oil variable and $X1, ..., X8$ are viewed as vinegar variables;
5. $Z_{47}, ..., Z_{54}$, where $X_9, ..., X_{16}$ are viewed as oil variable and $Y_9, ..., Y_{16}$ are viewed as vinegar variables;
6. $Z_{56}, ..., Z_{63}$, where $X_{17}, ..., X_{24}$ are viewed as oil variable and $Y_{17}, ..., Y_{24}$ are viewed as vinegar variables;
7. $Z_{65}, ..., Z_{72}$, where $Y_{25}, ..., Y_{32}$ are viewed as oil variable and $X_9, ..., X_{16}$ are viewed as vinegar variables.

The public key is constructed as a map from $k^{56\times d}$ to $k^{74\times d}$:

$$Y = (y_1, ..., y_{74\times d}) = \bar{F}(x_1, ..., x_{56\times d}) = L_1 \circ \Phi \circ F \circ \Phi' \circ L_2(x_1, ..., x_{56\times d}),$$

where Φ' is the map from k^{56d} to $\mathcal{F}^{56}$ induced from the map P in (5), and Φ is the map from $\mathcal{F}^{74}$ to k^{74d} induced from P^{-1}.

2.3 The Decryption Process

The decryption process requires to invert the maps and the key is how to invert the central map.

The key step is first to unmask the triangular system $(Z_1, ..., Z_7)$, and to derive the value of $X_1, .., X_7$ and then X_8. The rest is just to solve the oil-vinegar systems. We start by focusing on the first three equations of the central map system.

We will first write some of the Diophantine equations satisfied by components of the central map. Let

$$\begin{aligned} g_1 &= Z_8Z_9 + Z_{10}Z_{11} + Z_{12}Z_{13} + Z_{14}Z_{15} + Z_{16}Z_{17} \\ &= \psi_{1,1}(X_1...X_8)\psi_{1,2}(Y_1...Y_8), \\ g_2 &= Z_{18}Z_{19} + Z_{20}Z_{21} + Z_{22}Z_{23} + Z_{24}Z_{25} + Z_{26}Z_{27} \\ &= \psi_{2,1}(X_1...X_8)\psi_{2,2}(Y_9...Y_{16}), \\ g_3 &= Z_{28}Z_{29} + Z_{30}Z_{31} + Z_{32}Z_{33} + Z_{34}Z_{35} + Z_{26}Z_{36} \\ &= \psi_{2,1}(Y_1...Y_8)\psi_{2,2}(Y_9...Y_{16}). \end{aligned} \tag{9}$$

Note that Z_{26} appears in both g_2 and g_3 because of (7), (8).

Since

$$\psi_{2,1}(X_1, ..., X_8) = \psi_{1,1}(X_1, ..., X_8), \quad \psi_{2,1}(Y_1, ..., Y_8) = \psi_{1,2}(Y_1, ..., Y_8),$$

we have

$$h_1 = (g_1 g_2 g_3^{-1})^{1/2} = \psi_{1,1}(X_1, ..., X_8) \tag{10}$$

$$h_2 = g_1 h_1^{-1} = \psi_{1,2}(Y_1, ..., Y_8) \tag{11}$$

$$h_3 = g_2 h_1^{-1} = \psi_{2,2}(Y_9, ..., Y_{16}). \tag{12}$$

This allows us to compute the value of $\psi_{1,1}(X_1, ..., X_8)$, $\psi_{1,2}(Y_1, ..., Y_8)$ and $\psi_{2,2}(Y_9, ..., Y_{16})$.

Since we also have the Diophantine equations:

$$\begin{aligned} g_4 = Z_{37}Z_{38} + Z_{39}Z_{40} + Z_{41}Z_{42} + Z_{43}Z_{44} + Z_{45}Z_{46} \\ = \psi_{3,1}(X_1, .., X_8)\psi_{3,2}(Y_{17}, .., Y_{24}), \end{aligned} \tag{13}$$

$$\begin{aligned} g_5 = Z_{47}Z_{48} + Z_{49}Z_{50} + Z_{51}Z_{52} + Z_{53}Z_{54} + Z_{26}Z_{55} \\ = \psi_{2,1}(X_9, .., X_{16})\psi_{2,2}(Y_9, .., Y_{16}), \end{aligned} \tag{14}$$

$$\begin{aligned} g_6 = Z_{56}Z_{57} + Z_{58}Z_{59} + Z_{60}Z_{61} + Z_{62}Z_{63} + Z_{45}Z_{64} \\ = \psi_{3,1}(X_{17}, .., X_{24})\psi_{3,2}(Y_{17}, .., Y_{24}), \end{aligned} \tag{15}$$

$$\begin{aligned} g_7 = Z_{65}Z_{66} + Z_{67}Z_{68} + Z_{69}Z_{70} + Z_{71}Z_{72} + Z_{73}Z_{74} \\ = \psi_{4,1}(X_9, .., X_{16})\psi_{4,2}(Y_{25}, .., Y_{32}). \end{aligned} \tag{16}$$

Then, since

$$\psi_{3,1}(X_1, ..., X_8) = \psi_{1,1}(X_1, ..., X_8),$$

and

$$\psi_{4,1}(X_9, ..., X_{16}) = \psi_{2,1}(X_9, ..., X_{16}),$$

we have

$$h_4 = g_4 h_1^{-1} = \psi_{3,2}(Y_{17}, ..., Y_{24}), \tag{17}$$

$$h_5 = g_5 h_3^{-1} = \psi_{2,1}(X_9, ..., X_{16}), \tag{18}$$

$$h_6 = g_6 h_4^{-1} = \psi_{3,1}(X_{17}, ..., X_{24}), \tag{19}$$

$$h_7 = g_7 h_5^{-1} = \psi_{4,2}(Y_{25}, ..., Y_{32}). \tag{20}$$

Using the value of $h_1, ..., h_7$, we can restore the triangular structure of $Z_1, ..., Z_7$, and recover the values of $X_1, ..., X_7$.

Then we recover X_8 by using the value of $h_1 = \psi_{1,1}(X_1, ..., X_8)$, as long as X_7 is nonzero, or we can use Z_{17} to recover X_8 as long as X_2 is not zero.

One finishes the inversion of the central map by using the 1,2,4,5,6,7 oil-vinegar systems described in the section above to derive the remaining variables $X_9, ..., X_{24}$ and $Y_1, ..., Y_{32}$.

Note that the decryption process succeeds with a high probability but not 1, and our attack only deals with ciphertexts, whose decryption can be performed successfully as above.

2.4 Practical Parameters and Security Claims

The authors [7] suggested the following practical parameters:

1. $q = 2^{16}$, d=1, the number of variables 56, the number of public key polynomials 74, and the security level claim is 2^{113};
2. $q = 2^{16}$, d=2, the number of variables $56 \times 2 = 112$, the number of public key polynomials $74 \times 2 = 148$, and the security level claim is 2^{221};
3. $q = 2^{32}$, d=1, the number of variables 56, the number of public key polynomials 74, and the security level claim is 2^{114}.

3 Embedded Surface Attack

We will present the attack using the embedded surfaces.

3.1 Embedded Surface Attack

We will now concentrate our attack using the first suggested parameter. Namely we have: $q = 2^{16}$, d=1, $\mathcal{F} = k = GF(2^{16})$, the number of variables is 56, the number of public key polynomials is 74, and the security level claim is 2^{113}.

It is very clear that the key decryption process is in the relations (9), without which the decryption will not be possible. Let us first rewrite these relations explicitly as:

$$g_1 = h_1 \times h_2, \tag{21}$$
$$g_2 = h_1 \times h_3, \tag{22}$$
$$g_3 = h_2 \times h_3. \tag{23}$$

However, from these relations, we can derive the following very interesting relations:

$$\begin{aligned} g_1g_2 &= g_3 \times h_1^2, \\ g_2g_3 &= g_1 \times h_3^2, \\ g_1g_3 &= g_2 \times h_2^2. \end{aligned} \tag{24}$$

Let us look at the first relation explicitly as:

$$\begin{gathered} (Z_8Z_9 + Z_{10}Z_{11} + Z_{12}Z_{13} + Z_{14}Z_{15} + Z_{16}Z_{17}) \times \\ (Z_{18}Z_{19} + Z_{20}Z_{21} + Z_{22}Z_{23} + Z_{24}Z_{25} + Z_{26}Z_{27}) = \\ (Z_{28}Z_{29} + Z_{30}Z_{31} + Z_{32}Z_{33} + Z_{34}Z_{35} + Z_{26}Z_{36}) \times (\psi_{1,1}(X_1, ..., X_8))^2. \end{gathered} \tag{25}$$

which can be further written in the form:

$$\sum a_{i,j,s,t}Z_iZ_jZ_sZ_t = \sum a'_{i,j,s,t}Z_iZ_jX_s^2X_t^2. \tag{26}$$

or

$$\sum a_{i,j,s,t} Z_i Z_j Z_s Z_t - \sum a'_{i,j,s,t} Z_i Z_j X_s^2 X_t^2 = 0. \tag{27}$$

Here we would like to point out that this relation is true due to the fact that $\mathcal{F} = k$ is of characteristics 2, where

$$(a+b)^2 = a^2 + b^2.$$

This means that if

$$g_1 \times g_2 \times g_3 \neq 0,$$

an assumption required for decryption, again, due to the fact that $\mathcal{F} = k$ is of characteristics 2, if we have all relations in the form of (26), given the values of Z_i, we should be able to derive the values of

$$h_1^2 = (\psi_{1,1}(X_1, ..., X_8))^2,$$
$$h_2^2 = (\psi_{1,2}(Y_1, ..., Y_8))^2,$$
$$h_3^2 = (\psi_{2,2}(Y_9, ..., Y_{16}))^2,$$

and therefore the value of $h_1 = \psi_{1,1}(X_1, ..., X_8)$, $h_2 = \psi_{1,2}(Y_1, ..., Y_8)$ and $h_3 = \psi_{2,2}(Y_9, ..., Y_{16})$, by taking squareroot in k, a field of characteristic 2.

The above implies that, for the public key cryptosystem, for a pair of ciphertext and plaintext $(x_1, .., x_{56})$ and $(y_1, ..., y_{74})$, there are relations in the form:

$$\sum a_{i,j,s,t} y_i y_j y_s y_t + \sum b_{i,j,s} y_i y_j y_s + \sum c_{i,j} y_i y_j + \sum d_j y_j +$$
$$\sum a'_{i,j,s,t} y_i y_j x_s^2 x_t^2 + \sum b'_{i,s,t} y_i x_s^2 x_t^2 + \sum c'_{i,j,s} y_i y_j x_s^2 + \sum d'_{i,s} y_i x_s^2 +$$
$$\sum e'_{s,t} x_s^2 x_t^2 + \sum h'_s x_s^2 + e = 0, \tag{28}$$

which comes from (27). This new form is due to the affine transformations L_1 and L_2.

This can give us a none-trivial embedded surface, since if we are given the value of all y_i, we can derive polynomial equations in the form of

$$\sum a''_{s,t} x_s^2 x_t^2 + \sum b''_s x_s^2 + c'' = 0, \tag{29}$$

which gives us the value of polynomials corresponding to h_1 and h_2 and h_3, and they are not components in in the central map. This is true, as long as the corresponding value of $g_1 \times g_2 \times g_3$ is not zero, an assumption required for decryption.

The means that, if we get all the embedded surfaces as a linear space in the form of (28), we will actually be able to derive the corresponding value of h_1 and h_2 and h_3 for any valid (decryption possible)ciphertext $(y'_1, ..., y'_{74})$. If we amend those equations to the original system, in the context of original polynomial system derived from the known ciphertext, we will be able to derive the values corresponding to X_1, X_2, X_3.

Now, let us look at again the relations that are used to derive $h_3, .., h_7$ in the decryption process, which we will rewrite as:

$$h_4h_1 = \psi_{3,2}(Y_{17}, ..., Y_{24})h_1 = g_4 =$$
$$Z_37Z_{38} + Z_{39}Z_{40} + Z_{41}Z_{42} + Z_{43}Z_{44} + Z_{45}Z_{46}, \tag{30}$$
$$h_5h_3 = \psi_{2,1}(X_9, ..., X_{16})h_1 = g_5 =$$
$$Z_{47}Z_{48} + Z_{49}Z_{50} + Z_{51}Z_{52} + Z_{53}Z_{54} + Z_{26}Z_{55}. \tag{31}$$

and

$$h_6h_4 = \psi_{3,1}(X_{17}, ..., X_{24})h_1 = g_6 =$$
$$Z_{56}Z_{57} + Z_{58}Z_{59} + Z_{60}Z_{61} + Z_{62}Z_{63} + Z_{45}Z_{64}, \tag{32}$$
$$h_7h_5 = \psi_{4,2}(Y_{25}, ..., Y_{32}h_5 = g_7 =$$
$$Z_{65}Z_{66} + Z_{67}Z_{68} + Z_{69}Z_{70} + Z_{71}Z_{72} + Z_{73}Z_{74}. \tag{33}$$

This means, in the central map system, if we amend the h_1 and h_2 and h_3 to the map, and if we apply either Groebner basis algorithm like F_4 or F_5 of Faugere or the mutant XL family of algorithms [9],[2], in the first computation round when the algorithm reaches degree 4, we will derive the values of h_4 and h_5 as mutants, and in the next computation round, which is still at degree 4, we will derive h_6 and h_7 as mutants. This implies that we can derive the values of $X_4, .., X_7$ and therefore the value of X_8 as in the decryption process. Since the rest are just Oil-Vinegar type of systems, this further implies that we can solve the system at degree 4 using polynomial solving algorithms once the values h_1, h_2, h_3 are derived using the embedded surfaces in the form (28).

The above enables us to make a complete algorithm to attack the system.

1. **Step 1. Find all the embedded surfaces**

   Randomly pick

   $$\binom{74+4}{4} + \binom{56+2}{2} \times \binom{74+2}{2} = 1426425 + 4711050 = 6137475$$

   ciphertext and plaitext pairs derived from the public key, and substitute them into the equation in the form of (28), where the coefficients of the system are treated as variables. This will give us a set of linear equations with 6137475 variables over $GF(2^{16})$ and the same number of equations.

   Find the solutions for this set of linear equations. The solution space should be of dimension:

   $$(76 \times 75)/2 \times 76 + 76 \times 76 + 76 + 3 = 222455,$$

   where 222455 of them come from equations derived from the trivial relations from terms like $Z_iZ_jZ_k^2$, $Z_j^2Z_i$ and Z_i^2, and only 3 of them is what we really need, namely the ones coming from (24).

2. **Step 2. Derive new equations from the embedded surfaces**

   With the 222455 embedded surfaces, once given any valid ciphertext, we substitute it into the embedded surfaces, we will derive $74+3=77$ linearly independent degree 4 equations in the form:

   $$\sum A_{s,t}x_s^2x_t^2 \sum B_s x_s^2 + C = 0.$$

   Then, we take the square root of these equations due to the fact that it is over a field of characteristic 2, namely

   $$\sum A_{s,t}^{1/2}x_sx_t \sum B_s^{1/2}x_s + C^{1/2} = 0,$$

   since

   $$(\sum A_{s,t}^{1/2}x_sx_t \sum B_s^{1/2}x_s + C^{1/2})^2 = \sum A_{s,t}x_s^2x_t^2 \sum B_s x_s^2 + C,$$

   and comuting the square root over a field of of characteristic 2 is easy to do. This gives us a set of 77 linearly independent quadratic equations. All the public equations derived from the public key and the known ciphertext are already included in the span of this set of equations.

3. **Step 3. Reduce three variables**

   Perform Gaussian elimination on this set of equations to look for an equation in the form of

   $$\sum a_ix_i^2 + \sum b_ix_i + c = 0,$$

   where

   $$a_i = b_i^2 \times \alpha,$$

   for a fixed constant α. Solving this quadratic equation will give us a linear in the form

   $$\sum a_ix_i + b = 0.$$

   This corresponds to deriving the value of X_1, which comes from Z_1 by eliminating $\psi_{1,1}(X_1, ..., X_8)$ due to the known value of $h_1 = \psi_{1,1}(X_1, \ldots, X_8)$. We will then substitute this linear equation into the system, and perform Gaussian elimination, which gives us again an equation in the form of

   $$\sum a_ix_i^2 + \sum b_ix_i + c = 0,$$

   where again

   $$a_i = b_i^2 \times \alpha',$$

   for a some fixed constant α'. Solving this quadratic equation will give us a linear equation in the form $\sum a_ix_i + b = 0$.

   This corresponds to deriving the value of X_2, which comes from Z_2 and the known value of h_2.

   We will repeat the process to derive a new linear equation, which corresponds to deriving the value of X_3 coming from Z_3 and the known value of h_3.

4. **Step 4. Solve the system**

   We will feed the new system including the 3 new linear equations into a Groebner solver like F_4 or the mutant XL algorithm. We will solve the systems at degree 4. This will give us the value of the whole plaintext.

In principle, we can merge Step 3 and Step 4 by using directly the algebraic solver like F_4 or the mutant XL algorithm, which will yield the same results.

3.2 The Complexity of the Attack

From the attacking steps, it is clear that the complexity of the attacks concentrates on **Step 1** and **Step 4**.

In Step 1, the key part is to solve a system of linear equations with $N = 6137475$ variables and the same number of equations. If we use usual Gaussian elimination, the complexity will be roughly

$$2N^3/3 = 154126724635276031250 \approx 2^{68},$$

over the field $GF(2^{16})$. Assume that we use the best optimized linear solver, we should have the complexity $N^{2.3} \approx 2^{52}$ theoretically.

In Step 4, we will need to solve a linear system roughly with $\binom{53+4}{4} = 395010$ variables and the same number of equations. Clearly this system is much smaller than the system above, whose complexity is much smaller.

Therefore, we conclude that if we use the optimized Gaussian elimination, the complexity will be roughly 2^{52} theoretically, and the complexity will be 2^{68} with usual Gaussian elimination. This complexity is based on operations over $GF(2^{16})$. The original security claim for the system is 2^{113} with the assumption of using the optimized Gaussian elimination.

3.3 The Complexity for Attacking the Other Two Systems

For the case, where $q = 2^{32}$ and d=1, the attack complexity will be precisely the same except that everything will be on a field over $GF(2^{32})$. The original security assumption is 2^{114} with the assumption of using the optimized Gaussian elimination.

As for the case, where $q = 2^{16}$, d=2, it is clear the complexity will be determined by solving a set of linear equations with the number of variables and equations as

$$N = \binom{2*74+4}{4} + \binom{2*56+2}{2} * \binom{2*74+2}{2} = 93352225.$$

If we use normal Gaussian elimination, the complexity will be roughly

$$2N^3/3 \approx 2^{79},$$

over the field $GF(2^{16})$. Assume that we use the best optimized linear solver, we should have the complexity $N^{2.3} \approx 2^{61}$ theoretically. The original security claim is 2^{221} with assumption of using the optimized Gaussian elimination.

Due to the memory constraints in our own equipments, we could not perform the experiments to attack the system in practice. But we did perform some small scale toy experiments, where we set some of the variables in the central map to be 0, to confirm that our attack works indeed.

3.4 Direct Algebraic Attack

One may ask what if we use F_4 or the mutant XL directly against the new MPKCs? One can see easily, our embedded surface actually implies that the degree of regularity of the systems [1] is actually 8, since each y_i is of degree 2 and our surface is actually of degree 4 in y_i. This means that the direct algebraic solver would be much less efficient since our method finishes at degree 4.

4 Conclusion

We present a new attack on the new MPKCs from Diophantine equations developed by Gao and Hendl. This attack uses embedded surfaces associated with the new MPKCs. We show that this new attack can break the system efficiently. We believe such an approach is a very useful approach, which can be applied on other types of systems including symmetric systems.

We would like to point out that our attack relies very much on the fact that the field is of characteristic 2. We believe it deserves further attention to seek possibilities to rebuild the system using fields of odd characteristics, whose security could be very different as pointed out in[4].

Acknowledgments. The work was partially supported by **Charles Phelps Taft Foundation** and the NSF of China under the grant #60973131 and the National Natural Science Foundation of China under grant No. U1135004 and 61170080.

References

1. Bardet, M., Faugère, J.-C., Salvy, B.: On the complexity of Gröbner basis computation of semi-regular overdetermined algebraic equations. In: International Conference on Polynomial System Solving - ICPSS, pp. 71–75 (November 2004)
2. Ding, J., Buchmann, J., Mohamed, M., Mohamed, W., Weinmann, R.-P.: Mutant xL. In: First International Conference on Symbolic Computation and Cryptography, SCC 2008 (2008)
3. Ding, J., Hu, L., Nie, X., Li, J., Wagner, J.: High order linearization equation (HOLE) attack on multivariate public key cryptosystems. In: Okamoto, T., Wang, X. (eds.) PKC 2007. LNCS, vol. 4450, pp. 233–248. Springer, Heidelberg (2007a)

4. Ding, J., Schmidt, D., Werner, F.: Algebraic attack on HFE revisited. In: Wu, T.-C., Lei, C.-L., Rijmen, V., Lee, D.-T. (eds.) ISC 2008. LNCS, vol. 5222, pp. 215–227. Springer, Heidelberg (2008)
5. Dubois, V., Gama, N.: The degree of regularity of HFE systems. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 557–576. Springer, Heidelberg (2010)
6. Garey, M.R., Johnson, D.S.: Computers and intractability, A Guide to the theory of NP-completeness. W.H. Freeman, San Francisco (1979)
7. Gao, S., Heindl, R.: Multivariate public key cryptosystems from diophantine equations. Designs, Codes and Cryptography, 1–18 (November 2, 2011), doi:10.1007/s10623-011-9582-1
8. Heindl, R.A.: New directions in multivariate public key cryptography, Ph.D. Thesis, Clemson University. Mathematical Science - 2009 (2009)
9. Mohamed, M.S.E., Cabarcas, D., Ding, J., Buchmann, J., Bulygin, S.: MXL$_3$: An Efficient Algorithm for Computing Gröbner Bases of Zero-Dimensional Ideals. In: Lee, D., Hong, S. (eds.) ICISC 2009. LNCS, vol. 5984, pp. 87–100. Springer, Heidelberg (2010)
10. Patarin, J.: Cryptanalysis of the Matsumoto and Imai Public Key Scheme of Eurocrypt '88. In: Coppersmith, D. (ed.) CRYPTO 1995. LNCS, vol. 963, pp. 248–261. Springer, Heidelberg (1995)
11. Patarin, J.: The oil and vinegar signature scheme. Presented at the Dagstuhl Workshop on Cryptography (1997)
12. Shor, P.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Rev. 41(2), 303–332 (1999)
13. Wang, L.-C., Yang, B.-Y., Hu, Y.-H., Lai, F.: A "Medium-field" multivariate public-key encryption scheme. In: Pointcheval, D. (ed.) CT-RSA 2006. LNCS, vol. 3860, pp. 132–149. Springer, Heidelberg (2006)