ZHFE, a New Multivariate Public Key Encryption Scheme

Jaiberth Porras[1], John Baena[1], and Jintai Ding[2,3,⋆]

[1] Universidad Nacional de Colombia, Calle 59A No 63-20, Medellin, Colombia
{jporras,jbbaena}@unal.edu.co
http://www.unalmed.edu.co/∼escmat/
[2] University of Cincinnati, 4199 French Hall West, Cincinnati, OH 45221-0025, USA
jintai.ding@uc.edu
[3] Chinese Academy of Sciences, 52 Sanlihe Rd., Beijing, China

Abstract. In this paper we propose a new multivariate public key encryption scheme named ZHFE. The public key is constructed using as core map two high rank HFE polynomials. The inversion of the public key is performed using a low degree polynomial of Hamming weight three. This low degree polynomial is obtained from the two high rank HFE polynomials, by means of a special reduction method that uses Hamming weight three polynomials produced from the two high rank HFE polynomials. We show that ZHFE is relatively efficient and that it is secure against the main attacks that have threatened the security of HFE. We also propose parameters for a practical implementation of ZHFE.

Keywords: Multivariate cryptography, HFE polynomials, HFE cryptosystem, trapdoor functions, Zhuang-zi algorithm.

1 Introduction

Post-Quantum Cryptography stands for those cryptosystems which have the potential to resist possible future quantum computer attacks [3]. Multivariate public key cryptosystems (MPKCs) are an interesting option in Post-Quantum Cryptography [12]. Their main ideas come from algebraic geometry and usually the computations over these cryptosystems are very efficient.

The public key of an MPKC consists of a set of multivariate quadratic polynomials over a finite field. Thus, the security of an MPKC is related to the fact that solving a randomly system of multivariate quadratic polynomial equations over a finite field is an NP-hard problem [15]. Moreover, it seems that quantum computers have no advantage over the traditional computers to solve this problem.

⋆ Corresponding author.

M. Mosca (Ed.): PQCrypto 2014, LNCS 8772, pp. 229–245, 2014.

1.1 Hidden Field Equations

One of the most important MPKCs is named Hidden Field Equations (HFE), proposed by Patarin in 1996 [18]. To describe HFE, let us fix a finite field k of size q and a positive integer n. Next, we choose a degree n irreducible polynomial $g(y) \in k[y]$, and consider the field extension $K = k[y]/(g(y))$ and the isomorphism $\varphi\colon K \to k^n$ defined by $\varphi\left(u_1 + u_2 y + \ldots + u_n y^{n-1}\right) = (u_1, u_2, \ldots, u_n)$.

We say that a polynomial has *Hamming weight* W if the maximum of the q-Hamming weights of all its exponents is W. The q-Hamming weight of a non-negative integer is the sum of the q-digits of its q-nary expansion. Let $F : K \to K$ be a Hamming weight two polynomial of the form

$$F(X) = \sum_{0 \le j \le i}^{n-1} a_{ij} X^{q^i + q^j} + \sum_{i=0}^{n-1} b_i X^{q^i} + c \ ,$$

where the coefficients a_{ij}, b_i, c are chosen randomly in K. Such a polynomial F will be called an *HFE polynomial*. If in addition, we require that $\deg(F) \le D$, where D is a fixed positive integer, we will say that F is an *HFE polynomial with bound* D.

We now randomly choose an HFE polynomial $F\colon K \to K$ with bound D. The public key of HFE is $P(x_1, \cdots, x_n) = T \circ \varphi \circ F \circ \varphi^{-1} \circ S(x_1, \cdots, x_n)$, where S and T are two invertible affine transformations over k^n. The private key consists of the core map F together with the transformations S and T. We denote by $\mathrm{HFE}(q, n, D)$ an HFE scheme with the described parameters q, n and D. The degree D of the core polynomial F cannot be too large because the decryption process would be very slow. This restriction over HFE introduces a vulnerability against certain attacks like the direct algebraic attack [14] and the KS MinRank attack [17].

1.2 Previous Work

In [19] we proposed a special reduction method to construct new candidates for multivariate trapdoor functions using HFE polynomials of high degree and high rank. The idea of the construction is inspired by the first steps of the Zhuang-Zi algorithm [11]. Given a finite field k of size q and a degree n extension field K, we consider two high degree HFE polynomials over K of the form $F(X) = \sum a_{ij} X^{q^i + q^j} + \sum b_i X^{q^i} + c$ and $\tilde{F}(X) = \sum \tilde{a}_{ij} X^{q^i + q^j} + \sum \tilde{b}_i X^{q^i} + \tilde{c}$, where the coefficients $a_{ij}, b_i, c, \tilde{a}_{ij}, \tilde{b}_i, \tilde{c} \in K$ are to be determined. The idea behind the method is to construct a low degree polynomial Ψ of Hamming weight three of the form

$$\begin{aligned} \Psi = & X\left(\alpha_1 F_0 + \cdots + \alpha_n F_{n-1} + \beta_1 \tilde{F}_0 + \cdots + \beta_n \tilde{F}_{n-1}\right) + \\ & X^q\left(\alpha_{n+1} F_0 + \cdots + \alpha_{2n} F_{n-1} + \beta_{n+1} \tilde{F}_0 + \cdots + \beta_{2n} \tilde{F}_{n-1}\right) \ , \end{aligned}$$

where $F_0, F_1, \cdots, F_{n-1}$ are the Frobenius powers of F, and $\tilde{F}_0, \tilde{F}_1, \cdots, \tilde{F}_{n-1}$ are the Frobenius powers of $\tilde{F}$.

To obtain such a polynomial Ψ we need to determine the coefficients of F and $\tilde{F}$, also the scalars α_i and β_i, such that the degree of Ψ is less than or equal to a fixed positive integer D_0, which is chosen such that we can easily invert Ψ using Berlekamp's algorithm. The method that we proposed in [19] consists in randomly choosing values for the scalars α_i and β_i, and producing with them a linear system whose solution provides the coefficients of F and $\tilde{F}$. Once the scalars α_i and β_i are randomly chosen, the linear system that is obtained has more variables than equations, and thus we can guarantee nontrivial solutions for it. One could be tempted to randomly choose the variables coming from the coefficients of F and $\tilde{F}$, and then try to solve the linear system for the variables coming from the scalars, with the intention of having generic core polynomials F and $\tilde{F}$. However, this approach produces a linear system with more equations than variables, and hence, in general, this system has no nontrivial solutions.

The new multivariate trapdoor function is built similarly to the way in which HFE is constructed, except that now the core map is replaced by the map $G = (F, \tilde{F})$. The most-consuming-time task during the inversion of the trapdoor function is the inversion of the core map G. But this is an easy task according to the following proposition, which is proved in [19], and the use of Berlekamp's algorithm.

Proposition 1. *Let (Y_1, Y_2) be an element in $\mathrm{Im}(G) \subseteq K \times K$. Then the set of pre-images of (Y_1, Y_2) under the map $G = (F, \tilde{F})$ is a subset of the roots of the low degree polynomial*

$$\Psi' = \Psi - \sum_{j=1}^{2} X^{q^{j-1}} \sum_{i=1}^{n} \alpha_{i+n(j-1)} Y_1^{q^{i-1}} + \beta_{i+n(j-1)} Y_2^{q^{i-1}} \quad .$$

1.3 Contribution of this Paper

Some variants of HFE have been proposed as encryption schemes, but all of them have been proven to be insecure. The reason for this fact is that the polynomials used as core maps for these systems have been of low degree, and hence they have had low rank. This situation leads to the following question:

- Is there any way to enlarge the degree of an HFE polynomial used as core map for an encryption scheme, without affecting the efficiency of the decryption process?

We give here an affirmative answer to this question. We construct a new multivariate public key encryption scheme using the multivariate trapdoor function built in [19]. Since the new scheme utilizes as core map two HFE polynomials and the basic idea for the construction comes from the Zhuang-Zi algorithm [11], we call this new encryption scheme ZHFE. We give theoretical and experimental arguments to show that the encryption and decryption processes for ZHFE are relatively efficient. After performing the main known attacks that can threaten the security of these kind of schemes –the direct algebraic and the MinRank

attacks–, we propose parameters for ZHFE. We also give values for the main features of ZHFE for the suggested parameters.

This paper is organized as follows. In Sect. 2 we describe the new encryption scheme ZHFE, including a toy example and a suggestion of parameters for a practical implementation. In Sect. 3 we carry out a security analysis of ZHFE with respect to the direct algebraic and MinRank attacks. In Sect. 4 we give some conclusions and in the Appendix we provide additional information about the security analysis.

2 The New Encryption Scheme ZHFE

We use the new multivariate trapdoor function constructed in [19] to build ZHFE, utilizing two HFE polynomials of high degree and high rank. The main reason for using these high degree and high rank HFE polynomials is to resist the MinRank attack. However, the use of high degree HFE polynomials makes the decryption process almost impossible, unless those polynomials are constructed in such a way that the decryption is easy, regardless of the high degree of those polynomials. To accomplish this, we produce a low degree polynomial which we will use to decrypt. In addition, since we are utilizing high degree HFE polynomials for the core map, we expect that the public key has high degree of regularity, very different from what was observed by Faugère and Joux [14] for a system of quadratic equations derived from a single HFE polynomial with low degree. We will develop this point in Sect. 3.

One drawback of ZHFE is the generation time of the private key. The complexity of the reduction method introduced in [19] to produce the private key is polynomial: $O\left(\left(n^3\right)^{\omega}\right)$. Here $2 \leq \omega \leq 3$ is a constant that depends on the elimination algorithm used to solve the sparse linear system derived from the reduction method. In this reduction method we have to deal with huge matrices to reach large values of n. On the plus side we have that these matrices are sparse, which is an advantage in terms of efficiency.

2.1 Description of ZHFE

Let k be a finite field of size q. Fix a positive integer n and choose a degree n irreducible polynomial $g(y) \in k[y]$. Consider the field extension $K = k[y]/\left(g(y)\right)$ and the isomorphism $\varphi\colon K \to k^n$ defined by $\varphi\left(u_1 + u_2 y + \ldots + u_n y^{n-1}\right) = (u_1, u_2, \ldots, u_n)$. Let F, $\tilde{F}$ and Ψ be three polynomials in $K\left[X\right]/\left(X^{q^n} - X\right)$ constructed using the method described in Sect. 1.2, i.e., F and $\tilde{F}$ are two high degree HFE polynomials and Ψ is a low degree q-weight three polynomial which allows us to invert the map $G = (F, \tilde{F})$. Then we select two invertible affine transformations $S : k^n \to k^n$ and $T : k^{2n} \to k^{2n}$. The public map of ZHFE is the multivariate trapdoor function

$$P(x_1, \cdots, x_n) = T \circ (\varphi \times \varphi) \circ G \circ \varphi^{-1} \circ S(x_1, \cdots, x_n) \ .$$

Notice that P is a map from k^n to k^{2n} (see Fig. 1).

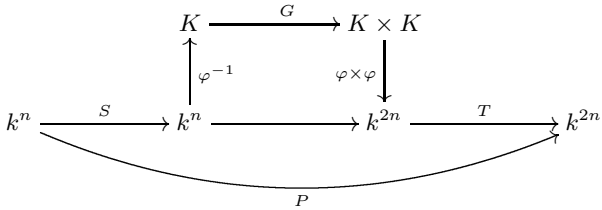

Fig. 1. Public key of ZHFE

Public Key. The public key of ZHFE includes:

- The field k and its structure.
- The trapdoor function $P(x_1, \cdots, x_n) = T \circ (\varphi \times \varphi) \circ G \circ \varphi^{-1} \circ S(x_1, \cdots, x_n)$.

Private Key. The private key of ZHFE includes:

- The low degree polynomial Ψ.
- The two invertible affine transformations S and T.
- The scalars $\alpha_1, \cdots, \alpha_{2n}, \beta_1, \cdots, \beta_{2n}$.

Encryption: To encrypt a plaintext $(x_1, \cdots, x_n) \in k^n$ we simply plug this plaintext into the public key $P = T \circ (\varphi \times \varphi) \circ G \circ \varphi^{-1} \circ S$ to obtain the ciphertext

$$(y_1, \cdots, y_{2n}) = P(x_1, \cdots, x_n) \in k^{2n} \ .$$

Decryption: To recover the plaintext from the ciphertext we must invert each part of P. We perform the following steps:

- We first compute $(w_1, \cdots, w_{2n}) = T^{-1}(y_1, \cdots, y_{2n})$.
- We next calculate $(Y_1, Y_2) = \left(\varphi^{-1}(w_1, \cdots, w_n), \varphi^{-1}(w_{n+1}, \cdots, w_{2n})\right)$.
- At this step we must invert the map $G = \left(F, \tilde{F}\right)$, i.e., we have to solve the equation $G(X) = (Y_1, Y_2)$. The solutions of this equation are part of the roots of the low degree polynomial Ψ', obtained from Ψ and (Y_1, Y_2) as in Proposition 1. Let $\mathcal{Z}$ be the set

  $$\mathcal{Z} = \{X \in K / \Psi'(X) = 0\}.$$

  We must now determine which elements of $\mathcal{Z}$ are solutions of the polynomial equation $G(X) = (Y_1, Y_2)$. In our extensive experiments we always got that only one element of $\mathcal{Z}$ was a solution for this equation.
- For each solution $X \in \mathcal{Z}$ of the equation $G(X) = (Y_1, Y_2)$, we compute the vector $\varphi(X) \in k^n$.
- Finally, we apply the transformation S^{-1} to each vector found in the previous step and these vectors are the candidates to be the plaintext. To determine which of these is the original plaintext, some redundant information must be added to the plaintext[1].

[1] In all our extensive experiments for each ciphertext, there was only one candidate to be the plaintext.

2.2 Toy Example

This example shows how the ZHFE scheme works. Set $q = 3$ and $n = 3$, and consider the field with three elements $k = GF(3)$. We select the irreducible polynomial $g(y) = y^3 + 2y + 1 \in k[y]$. A degree n extension field of k is $K = k[y]/(g(y))$. We can choose a generator $b \in K$ of the multiplicative group of K such that $g(b) = 0$, and we use this element to write the elements of K as powers of it. Let us take $D_0 = 5$. We now randomly choose the scalars $(\alpha_1, \cdots, \alpha_6) = (b^{23}, b^9, b^{22}, b^{16}, b^{24}, b^{22})$ and $(\beta_1, \cdots, \beta_6) = (b^5, b^{10}, b^{16}, 0, b^{17}, b^{14})$. Then, as explained in Sect. 1.2, we construct the polynomials $F(X) = b^{23}X^{18}+b^{16}X^{12}+b^{10}X^{10}+b^{23}X^9+b^{21}X^6+b^{24}X^4+b^{24}X^3+b^2X^2+bX$, $\tilde{F}(X) = b^{15}X^{18}+b^{25}X^{12}+b^{19}X^{10}+b^{14}X^9+bX^6+b^2X^4+b^{11}X^3+b^5X^2+b^{14}X$ and $\Psi(X) = b^{16}X^5 + b^7X^4 + b^{25}X^3 + b^9X^2$. We also select the invertible affine transformations

$$S(x_1, x_2, x_3) = \begin{pmatrix} 1 & 2 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 2 \end{pmatrix}$$

and

$$T(x_1, x_2, x_3, x_4, x_5, x_6) = \begin{pmatrix} 2 & 2 & 2 & 0 & 2 & 1 \\ 1 & 1 & 0 & 2 & 1 & 0 \\ 2 & 0 & 0 & 2 & 0 & 1 \\ 1 & 0 & 1 & 0 & 2 & 2 \\ 1 & 2 & 0 & 1 & 0 & 1 \\ 2 & 1 & 0 & 1 & 2 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 2 \\ 0 \\ 2 \\ 1 \end{pmatrix}.$$

The core map is $G(X) = \left(F(X), \tilde{F}(X)\right)$. The composition $P(x_1, x_2, x_3) = T \circ (\varphi \times \varphi) \circ G \circ \varphi^{-1} \circ S(x_1, x_2, x_3)$ yields the public key polynomials

$$\begin{aligned}
p_1(x_1, x_2, x_3) =& 2x_1^2 + x_1x_2 + x_1x_3 + x_1 + x_3 + 2, \\
p_2(x_1, x_2, x_3) =& 2x_1^2 + x_1x_2 + x_1 + x_2^2 + x_2x_3 + x_2 + 2x_3, \\
p_3(x_1, x_2, x_3) =& x_1^2 + x_1 + x_2 + x_3^2 + 1, \\
p_4(x_1, x_2, x_3) =& 2x_1^2 + 2x_1x_3 + x_1 + x_2^2 + 2x_2x_3 + x_2 + x_3^2 + 2, \\
p_5(x_1, x_2, x_3) =& x_1^2 + 2x_1x_2 + 2x_1 + x_2^2 + 2x_2 + x_3 + 1, \\
p_6(x_1, x_2, x_3) =& x_1x_2 + x_1x_3 + 2x_2^2 + x_2x_3 + x_3^2.
\end{aligned}$$

We now illustrate the encryption and decryption processes. Let $(x_1, x_2, x_3) = (0, 1, 1)$ be a plaintext. After plugging this plaintext into the public key, we obtain the ciphertext

$$(y_1, y_2, y_3, y_4, y_5, y_6) = (0, 2, 0, 1, 2, 1).$$

In order to recover the plaintext from the ciphertext we first compute

$$(w_1, \cdots, w_6) = T^{-1}(0, 2, 0, 1, 2, 1) = (0, 0, 1, 1, 1, 2).$$

We then calculate

$$\begin{aligned}(Y_1, Y_2) &= \left(\varphi^{-1}(w_1, w_2, w_3), \varphi^{-1}(w_4, w_5, w_6)\right) \\ &= \left(\varphi^{-1}(0,0,1), \varphi^{-1}(1,1,2)\right) \\ &= \left(b^2, b^{20}\right).\end{aligned}$$

As explained in Proposition 1, we now create the low degree polynomial Ψ' using the low degree polynomial Ψ, the scalars $\alpha_1, \cdots, \alpha_6, \beta_1, \cdots, \beta_6$ and the vector $(Y_1, Y_2) = \left(b^2, b^{20}\right)$:

$$\Psi' = b^{16}X^5 + b^7X^4 + b^9X^2 + b^{15}X \ .$$

The set of roots of Ψ' is[2] $\mathcal{Z} = \left\{0, b^{11}\right\}$. The only element of $\mathcal{Z}$ which is solution of the equation $G(X) = (Y_1, Y_2) = \left(b^2, b^{20}\right)$ is $X = b^{11}$. If we apply the isomorphism φ we get $\varphi\left(b^{11}\right) = (2,1,1)$. We next apply the transformation S^{-1} and then we recover the plaintext $S^{-1}(2,1,1) = (0,1,1)$.

The main part of the decryption process is the computation of roots of the polynomial Ψ'. For this task we use Berlekamp's algorithm. This algorithm has complexity $\mathcal{O}\left(nD^2 \log_q D + D^3\right)$, where D is the degree of the univariate polynomial. According to this complexity, it is expected that the degree of Ψ', which is determined by the parameter D_0, has the greatest impact on the decryption time. This fact was confirmed by our experiments. Table 1 shows some average encryption and decryption times for several choices of the parameters (q, n, D_0). For each parameter choice we encrypted and decrypted 100 messages. To perform the experiments we used the software Magma V2.20-2 on an Intel Core i5-3210M CPU 2.50 GHz × 4 with 12 GB of memory installed.

Table 1. Encryption and decryption time for ZHFE, 100 messages were tested per key

q	n	D_0	Average encryption time [s]	Average decryption time [s]
7	35	57	0.006	0.089
7	55	105	0.024	0.427
11	35	33	0.003	0.043
11	35	253	0.005	0.760

2.3 Suggestion of Parameters for a Practical Implementation

In this section we propose values for the parameters (q, n, D_0) for a realistic application of ZHFE. We base our choices on the data collected with the extensive experiments of encryption and decryption time, and with the security analysis that we perform in Sects. 3.1 and 3.2.

[2] These roots are found using the Magma implementation of Berlekamp's algorithm.

Our suggestion is $(q, n, D_0) = (7, 55, 105)$, and let us denote the associated scheme by ZHFE(7,55,105). This means that the finite field k has size $q = 7$, the number of variables of the public polynomials is $n = 55$, and the polynomial Ψ has degree $D_0 = 105$. The public map is $P\colon k^{55} \to k^{110}$ and then the public key has $2n = 110$ quadratic polynomials with 55 variables. To store the coefficients of these polynomials we need about 66 KB.

A plaintext is a tuple $(x_1, \cdots, x_{55}) \in k^{55}$ with 165 bits of length and a ciphertext is a tuple $(y_1, \cdots, y_{110}) \in k^{110}$ with 330 bits of length.

The private key comprises the low degree polynomial Ψ, the scalars α_i and β_i and the transformations $S\colon k^{55} \to k^{55}$ and $T\colon k^{110} \to k^{110}$. The polynomial Ψ has at most 14 terms. The coefficients of Ψ and the scalars α_i and β_i are in an extension field of k of degree 55. Thus, to store the private key we need about 11 KB.

In terms of efficiency, we now compare ZHFE to HFE Challenge 1 proposed by Patarin [18]. This system was broken in [14] by means of the direct algebraic attack. We focus on the most-consuming-time task for this kind of schemes, that is, the decryption process. Challenge 1 is the instance HFE(2,80,96). In 2008 Ding, Schmidt and Werner [13] proposed the instance HFE(11,89,132), but this was also broken by Faugère et al. [4]. Compared to Patarin's Challenge 1, HFE(11,89,132) takes about twice the time to decrypt. Decryption for ZHFE(7,55,105) is faster than HFE(11,89,132) because all the parameters are smaller. Therefore, in terms of efficiency, ZHFE(7,55,105) is comparable with Patarin's Challenge 1.

Based on the security analysis that we will explain in Sects. 3.1 and 3.2, we conclude that our choice of parameters gives a security level greater than 2^{80}.

3 Security Analysis

There are two attacks that have broken the security of HFE type schemes: the direct algebraic attack and the KS MinRank attack. Since ZHFE belongs to the HFE scheme family, we must consider these attacks against our new encryption scheme.

3.1 Direct Algebraic Attack

Let us briefly review the direct algebraic attack. Suppose that someone, who does not know the private trapdoor information, wants to invert the public key $P\colon k^n \to k^{2n}$ of the new encryption scheme $(P = (p_1, \ldots, p_{2n}))$. She wants to find the pre-images of an element $(y_1, \ldots, y_{2n}) \in \mathrm{Im}(P) \subseteq k^{2n}$. This person only has access to the public key P. In order to accomplish this, she tries to solve the system of quadratic equations

$$p_1(x_1, \ldots, x_n) - y_1 = 0, \cdots, p_{2n}(x_1, \ldots, x_n) - y_{2n} = 0 \ . \tag{1}$$

Solving the system (1) directly is known as the *direct algebraic attack*. One way to solve this system is finding a Gröbner basis for the ideal of $k[x_1, \cdots, x_n]$ generated by the polynomials $p_1 - y_1, \cdots, p_{2n} - y_{2n}$.

The F_4 function of MAGMA, [5], is the most efficient implementation of the Gröbner basis F_4 algorithm that is currently available. We ran extensive experiments using the F_4 algorithm of MAGMA to perform the direct algebraic attack for several choices of the parameters (q, n, D_0). We show here the results of our experiments for $q = 7$. For each choice of the parameters we used 10 different sets of quadratic equations to run the experiments.

In Table 2 and Fig. 2 we can observe that the time needed to solve the equations coming from the public key of ZHFE has an exponential growth in n. We can also see this behaviour with the memory used by the F_4 algorithm. This situation is different from the one observed by Faugere and Joux in [14]. The difference lies on the fact that in [14] the quadratic equations are produced using a polynomial of fixed low degree as core map in the HFE cryptosystem, and in our new cryptosystem the quadratic equations are generated via two high degree polynomials. In our experiments, in general, these two high degree polynomials have full degree $D = 2q^{n-1}$, so in particular this degree increases as n increases. This is the fundamental security improvement of ZHFE, when compared to traditional HFE type schemes in which D is a fixed positive integer.

Table 2. Algebraic attack against ZHFE for $q = 7$ and $D_0 = 105$. Ten systems were tested for each choice of parameters.

n	Average time [s]	Minimum time [s]	Maximum time [s]	Memory [MB]	$\lfloor \log_q D \rfloor$
12	0.071	0.06	0.09	32	11
14	0.289	0.28	0.31	32	13
16	5.564	5.5	5.64	64	15
18	31.392	31.01	32.19	128	17
20	148.208	143.69	160.73	288	19
22	942.269	663.62	988.45	681	21
24	18114.05	18099.43	18128.67	8334	23

Another evidence that the complexity of the algebraic attack against ZHFE is exponential, is that the degree of regularity of the trapdoor function increases as n increases. This behaviour can be observed in Fig. 3. This trend can also be explained by the fact that $D = 2q^{n-1}$ for ZHFE.

In order to compare ZHFE to the MQ-problem, we chose systems of random quadratic equations of the same dimensions ($k^n \to k^{2n}$) and performed the algebraic attack against these systems too. For each system of random equations, we found that the time needed to solve such equations using Gröbner bases is essentially the same that the one needed to solve the quadratic equations from the public key of ZHFE. These data are shown in Table 3. Notice that the degree of regularity is the same in both cases. Figure 4 shows graphically the time comparison for the two systems. In that graph we can observe that the two curves are indistinguishable.

The reader might think that the low degree of the Hamming weight three polynomial Ψ could introduce a possible weakness to ZHFE against the direct

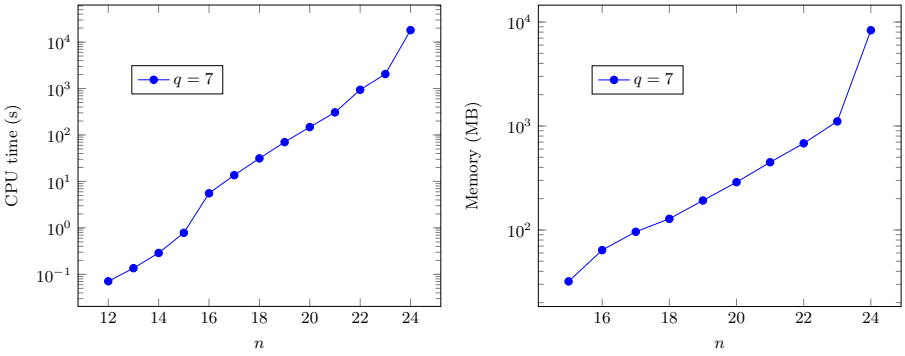

Fig. 2. Algebraic attack against ZHFE for $q = 7$ and $D_0 = 105$

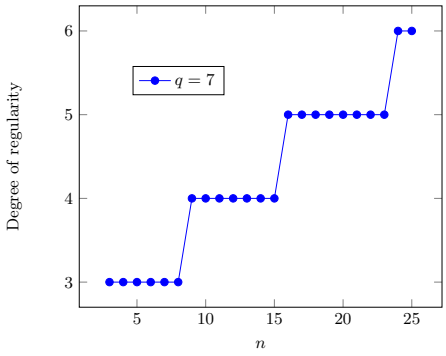

Fig. 3. Algebraic attack against ZHFE for $q = 7$ and $D_0 = 105$

algebraic attack. However, as it was shown in [13,7,8,2], the use of odd characteristic fields in HFE type schemes, provides a resistance against a Gröbner bases attack, regardless of the degree of the core polynomials used to construct the public key. The vulnerability of the schemes proposed in [13,7,8,2] is not against the direct algebraic attack, but against the MinRank attack [4]. The novelty of the present paper is that with the ZHFE cryptosystem we overcome this weakness. We will develop this idea in the next section.

3.2 Kipnis-Shamir MinRank Attack (KS Attack)

In 1999 Kipnis and Shamir [17] proposed a key-recovery attack against HFE that takes advantage of the low rank of the matrix associated to the core map. The KS attack exploits the structure behind the construction of HFE and it links the cryptanalysis of HFE with a linear algebra problem known as the MinRank Problem. Although we are using high degree and high rank polynomials

Table 3. Algebraic attack comparison between ZHFE and a system of random equations for $q = 7$ and $D_0 = 105$

(a) ZHFE

n	Average time [s]	Memory [MB]	Dreg
16	5.564	64	5
18	31.392	128	5
20	148.208	288	5
22	942.269	681	5
24	18114.05	8334	6

(b) Random equations

n	Average time [s]	Memory [MB]	Dreg
16	5.6	64	5
18	32.19	128	5
20	144.09	288	5
22	991.72	681	5
24	18012.19	8334	6

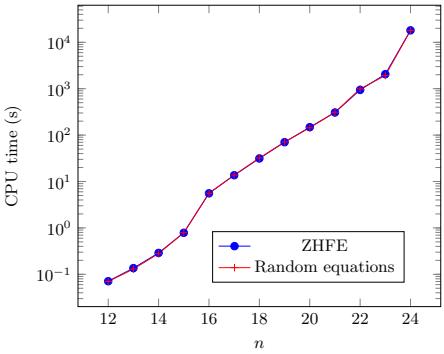

Fig. 4. Algebraic attack comparison between ZHFE and a system of random equations for $q = 7$ and $D_0 = 105$

as the core map in ZHFE, this attack could work if there was a low rank linear combination of their Frobenius powers. Because of this, we have to carefully consider this attack for our new cryptosystem.

The MinRank Problem. Let L be a finite field and consider m matrices $M_1, \cdots, M_m$ over L of size $t \times t$. Given an integer $r \leq t$, the problem is to find, if they exist, scalars $\lambda_1, \cdots, \lambda_m$, not all zero, such that

$$\text{Rank}\left(\sum_{i=1}^{m} \lambda_i M_i\right) \leq r \ .$$

This is in general an NP-hard problem [6]. However for small r, which is the case in HFE, this problem is not too hard. There exist two algebraic ways to attack this problem: the Kipnis-Shamir and the Minors Models (see Appendix).

We now test ZHFE against the KS attack, by performing extensive computer experiments for the case of odd characteristic. For characteristic 2 the attack is slightly different, and we did not perform experiments for this case. All the computations of this section were run using Magma V2.20-2 on a Sun X4440

server, with four Quad-Core AMD OpteronTM Processor 8356 CPUs and 128 GB of main memory (each CPU is running at 2.3 GHz).

The main part of the KS attack, with respect to the complexity, is to solve the MinRank problem. The original version of the KS attack was not as efficient as its authors claimed [16], because the derived MinRank problem worked with matrices with entries in the big field K. Recently, Faugère et al. [4] improved and generalized the KS attack, and were able to break HFE and its generalization Multi-HFE [7] for all practical choices of their parameters. Their main improvement was to restate the MinRank problem with the matrices associated to the public key, whose entries are in the small field k. This makes the improved KS attack significantly faster than the original version.

Let us explain how the KS attack is performed. We begin by noticing that the new trapdoor function P, which is part of the public key of ZHFE, can be seen as a particular case of the public key of an *unbounded* Multi-HFE cryptosystem as presented in [4], with $N = 2$ (for unbounded we mean that the core polynomials have no restrictions for their degrees). Because of this, in this section we perform the KS attack as it was done in [4] for a Multi-HFE scheme. For given parameters q, n and D_0, we generate the $2n$ public key polynomials $p_1, \cdots, p_{2n}$ of the new encryption scheme ZHFE ($P = (p_1, \cdots, p_{2n})$). Then, we compute the symmetric matrix M_i associated to the quadratic part of each public key polynomial p_i, $i = 1, \cdots, 2n$. Let Q-Rank(P) be the minimal rank of elements in the K-linear space generated by the matrices $M_1, \cdots, M_{2n}$. In [4] they showed that Q-Rank(P) coincides with the minimal quadratic rank of elements in the K-linear space generated by the Frobenius powers of the core polynomials F and $\tilde{F}$. The KS attack is successful against Multi-HFE when Q-Rank(P) is low (see [4]). The main purpose of this section is to show that Q-Rank(P) increases as n increases for ZHFE, and therefore the KS attack will not work against this new cryptosystem.

As it has been proved in [4] for a Multi-HFE scheme, in order to accelerate the solution of the MinRank problem, we can randomly fix $N = 2$ of the scalars $\lambda_1, \cdots, \lambda_{2n} \in K$, not all to zero. In our experiments we fixed $\lambda_{n-1} = 0$ and $\lambda_n = 1$, and we used the Kipnis-Shamir modelling for solving this MinRank problem. The reason to choose this modelling is that the minors modelling uses considerably more memory than the KS option.

We now use the MinRank problem to determine the Q-Rank(P) for different combinations of the parameters (q, n, D_0). For each n we start by taking $r = 1$ and then use the KS modelling. We utilize the Magma implementation of the F_4 algorithm to solve the equations produced by this modelling. Table 4 shows the results obtained for $q = 7$ and $D_0 = 105$[3]. If for $r = 1$ the solution set of the MinRank problem is empty, then we set $r = r + 1$ and repeat this process until a solution is found. For example, in Table 4 the expression "> 3" means that for

[3] The instances $n = 8$ and $n = 10$ for $r = 4$ did not terminate since the processes had a 50 GB memory limitation. When this limit was reached the processes automatically stopped after more than 10 days of running time.

$r \in \{1, 2, 3\}$ the solution set obtained for the MinRank problem was empty, so Q-Rank$(P) > 3$ for that case.

Table 4. KS attack against ZHFE, for $q = 7$ and $D_0 = 105$

n	Q-Rank(P)	Average time	Maximum memory
2	1	0.010 s	32
4	1	0.010 s	32
6	2	1.340 s	32
8	> 3	> 10 days	> 50 GB[3]
10	> 3	> 10 days	> 50 GB[3]

In Table 5 we show the time and memory needed to find the solution set for the MinRank problem for $(q, n, D_0) = (7, 8, 110)$ and different values of r. The same situation is observed for other combinations of the parameters. We can see how fast those values increase as r increases. The results in Tables 4 and 5 lead us to think that the larger Q-Rank(P) is the less feasible to solve the MinRank problem is.

Table 5. Time and memory needed to find the solution set for the KS attack against ZHFE, for $q = 7$, $n = 8$ and $D_0 = 105$

r	Average Time	Maximum memory
1	0.040 s	32
2	0.510 s	32 MB
3	297.410 s	462 MB
4	> 10 days	> 50 GB[3]

Now, for a fixed pair (q, n) we randomly choose a set of $2n$ quadratic equations in n variables, and perform the same process that we just used with ZHFE, in order to compare with the results that we obtained for ZHFE. The results are summarized in Table 6. We notice that we get exactly the same results for both cases. We also see that, for ZHFE, the value of Q-Rank(P) is independent of the value of D_0.

Another interesting experiment is to compare the performance of the KS attack against ZHFE with the performance of that attack against a system built in a similar way, but with low rank core polynomials F and $\tilde{F}$, i.e., a standard (bounded) Multi-HFE scheme. Table 7 shows these results for $q = 7$ and several values of n. We can observe that for the standard Multi-HFE the KS MinRank attack succeeds, while for the new encryption scheme ZHFE (Table 4) it does not. According to Tables 4, 6 and 7, we think that the quadratic rank Q-Rank(P) grows as n grows.

According to our experiments and the fact that we are using high rank core polynomials to construct the public key, we believe that ZHFE behaves as if it were a set of random equations with respect to the KS MinRank attack.

Table 6. Q-Rank(P) comparison between ZHFE and random equations for $q = 7$

(a) ZHFE

n	$D_0 = 105$	$D_0 = 399$	$D_0 = 2751$
2	1	1	1
4	1	1	1
6	2	2	2
8	> 3	> 3	> 3
10	> 3	> 3	> 3

(b) Random equations

n	Q-Rank(P)
2	1
4	1
6	2
8	> 3
10	> 3

Table 7. KS attack against a bounded Multi-HFE scheme for $q = 7$ and $\lfloor \log_q D \rfloor = 2$

n	Q-Rank(P)	Average time [s]	Maximum Memory [MB]
2	1	0.050	32
4	1	0.100	32
6	2	1.135	32
8	2	1.190	32
10	2	6.090	32
12	2	23.080	64
14	2	67.500	138
16	2	192.850	211
18	2	479.150	363
20	2	885.720	711

4 Conclusions

We have constructed a new multivariate public key encryption scheme called ZHFE. The core map of ZHFE consists of two high rank HFE polynomials. Until now, no one had proposed any idea of how to use high degree polynomials for the core map in HFE or any of its variants, since we always had the problem of the inversion of such core polynomials. Our novel idea has allowed us to invert a map built with two high degree HFE polynomials by means of a third polynomial of low degree.

We showed that the encryption and decryption processes for ZHFE are relatively efficient. Moreover, we showed that the attacks that have threatened the security of HFE, the direct algebraic and the Kipnis-Shamir MinRank attacks, do not work against ZHFE. We gave theoretical and experimental arguments to show that ZHFE behaves as if it were a system of random quadratic equations against these attacks.

We performed numerous computer experiments to test the security and measure the encryption/decryption times for several sets of parameters of ZHFE. The data we collected guided our choices for the parameters (q, n and D_0) for plausible schemes.

What we present in this paper is the beginning of a new idea and it is necessary to explore more deeply the different features and parameters of ZHFE, in order

to achieve a better understanding of its behaviour and security. For instance, in Sect. 2.3 we chose $D_0 = 105$ so that the polynomial Ψ does not have too few terms, with the intention to avoid having an extremely simple polynomial Ψ. Although this seems reasonable, we will have to study more carefully the effect of the parameter D_0 and the shape of the polynomial Ψ on the security of the new encryption scheme ZHFE.

In principle there seems to be no obvious way to recover the private polynomial Ψ (F, $\tilde{F}$ and the scalars α_i, β_i are secret) from the public key. This is an important point in the study of the security of ZHFE and we will have to consider this aspect more carefully in the future. We also want to study ways of speeding up the reduction method to construct the trapdoor functions. Speeding up the reduction method will also allow us to reach larger values of n and therefore we will be able to implement plausible schemes with smaller values of q, for example $q = 2$.

Acknowledgements. We want to thank Wael Mohamed and Daniel Cabarcas for running essential experiments for us, which helped us complete this paper. J. Ding was partially supported by the CAS/SAFEA International Partnership Program for Creative Research Teams.

References

1. Ars, G., Faugère, J.-C., Imai, H., Kawazoe, M., Sugita, M.: Comparison Between XL and Gröbner Basis Algorithms. In: Lee, P.J. (ed.) ASIACRYPT 2004. LNCS, vol. 3329, pp. 338–353. Springer, Heidelberg (2004)
2. Baena, J.B., Clough, C.L., Ding, J.: New Variants of the Square-Vinegar Signature Scheme, Revista Colombiana de Matemticas (Colombian Journal of Mathematics), Bogotá, 45(2) (2011)
3. Bernstein, D.J., Buchmann, J., Dahmen, E.: Post quantum cryptography. Springer (2009)
4. Bettale, L., Faugère, J.-C., Perret, L.: Cryptanalysis of hfe, multi-hfe and variants for odd and even characteristic. Designs, Codes and Cryptography 69(1), 1–52 (2013)
5. Bosma, W., Cannon, J., Playoust, C.: The Magma algebra system. I. The user language. J. Symbolic Comput. 24(3-4), 235–265 (1997); Computational algebra and number theory, London (1993)
6. Buss, J.F., Frandsen, G., Shallit, J.O.: The computational complexity of some problems of linear algebra. In: Reischuk, R., Morvan, M. (eds.) STACS 1997. LNCS, vol. 1200, pp. 451–462. Springer, Heidelberg (1997)
7. Chen, C.H.O., Chen, M.S., Ding, J., Werner, F., Yang, B.Y.: Odd-char multivariate hidden field equations. cryptology eprint archive (2008)
8. Clough, C., Baena, J., Ding, J., Yang, B.-Y., Chen, M.-S.: Square, a New Multivariate Encryption Scheme. In: Fischlin, M. (ed.) CT-RSA 2009. LNCS, vol. 5473, pp. 252–264. Springer, Heidelberg (2009)
9. Courtois, N.T.: The Security of Hidden Field Equations (HFE). In: Naccache, D. (ed.) CT-RSA 2001. LNCS, vol. 2020, pp. 266–281. Springer, Heidelberg (2001)

10. Courtois, N., Klimov, A., Patarin, J., Shamir, A.: Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 392–407. Springer, Heidelberg (2000)
11. Ding, J., Gower, J.E., Schmidt, D.S.: Zhuang-Zi: A New Algorithm for Solving Multivariate Polynomial Equations over a Finite Field, Preprint, University of Cincinnati (2006)
12. Ding, J., Gower, J.E., Schmidt, D.S.: Multivariate public key cryptosystems. Advances in Information Security, vol. 25. Springer, New York (2006)
13. 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)
14. Faugère, J.-C., Joux, A.: Algebraic Cryptanalysis of Hidden Field Equation (HFE) Cryptosystems Using Gröbner Bases. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 44–60. Springer, Heidelberg (2003)
15. Garey, M.R., Johnson, D.S., et al.: Computers and Intractability: A Guide to the Theory of NP-completeness. WH Freeman, San Francisco (1979)
16. Jiang, X., Ding, J., Hu, L.: Kipnis-Shamir Attack on HFE Revisited. In: Pei, D., Yung, M., Lin, D., Wu, C. (eds.) Inscrypt 2007. LNCS, vol. 4990, pp. 399–411. Springer, Heidelberg (2008)
17. Kipnis, A., Shamir, A.: Cryptanalysis of the HFE public key cryptosystem by relinearization. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 19–30. Springer, Heidelberg (1999)
18. Patarin, J.: Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): Two new families of asymmetric algorithms. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 33–48. Springer, Heidelberg (1996)
19. Porras, J., Baena, J., Ding, J.: New candidates for multivariate trapdoor functions, Cryptology ePrint Archive, Report 2014/387 (2014), `http://eprint.iacr.org/2014/387.pdf`
20. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. on Computing, 1484–1509 (1997)

Appendix: More about the KS MinRank Attack

Kipnis-Shamir Modeling. Kipnis and Shamir [17] proposed to bind the MinRank Problem to the problem of solving an algebraic quadratic system of equations. They noted that, if the matrix $M = \lambda_1 M_1 + \cdots + \lambda_m M_m$ has rank at most r, its left kernel $\{\boldsymbol{x} \in L^t \ : \ \boldsymbol{x}M = 0\}$ has at least $t - r$ linearly independent vectors. Therefore, solving the MinRank problem is equivalent to solving the system that comes from vanishing the entries of the matrix

$$\begin{pmatrix} 1 & & x_{1,1} & \cdots & x_{1,r} \\ & \ddots & \vdots & \vdots & \\ & & 1 \; x_{t-r,1} & \cdots & x_{t-r,r} \end{pmatrix} \left(\sum_{i=1}^{m} \lambda_i M_i \right).$$

This yields an overdetermined quadratic system with $t(t-r)$ equations and $t(t-r)+m$ variables. The authors in [17] proposed a method for solving this system which they called relinearization. Later on, in [10], it has been shown that this method can be seen as a special case of the XL algorithm. In fact, the XL algorithm can be viewed as a redundant variant of the Gröbner basis algorithm F_4 [1]. Therefore, this system is usually solved using Gröbner basis tools like F_4.

Minors Modeling. Courtois proposed another way to solve the MinRank Problem [9]. Since the matrix $\lambda_1 M_1 + \cdots + \lambda_m M_m$ has rank at most r, all its minors of order $(r+1) \times (r+1)$ must be zero. In this way we get a system of $\binom{t}{r+1}^2$ polynomial equations in the m variables λ_i. Notice that this system has many more equations than the system coming from the Kipnis-Shamir Modeling, but the equations have degree $r+1$.