# A Cryptanalysis of the Double-Round Quadratic Cryptosystem

Antoine Scemama

Johann Wolfgang Goethe University
Frankfurt am Main, Germany
scemama@math.uni-frankfurt.de

**Abstract.** In the 80's Matsumoto and Imai [8] proposed public key cryptosystems based on the difficulty of solving systems of polynomials in several variables. Although these first schemes were broken, many others followed, leading to a very active field known as Multivariate cryptography. In this paper, we show how to break one of these schemes, the Double-Round Quadratic cryptosystem from [12]. We stress that this cryptosystem has, in practice, already been cryptanalysed in [5]. However their attack uses several "non-standard" heuristics, they provide experimental evidence, but no proof is given, as opposed to this present article. Our attack uses a very general technique introduced in [9] to break the cryptosystem.

**Keywords:** Multivariable Cryptography, Double-Round Quadratic Cryptosystem, 2R Cryptosystem.

## 1 Introduction

The introduction of multivariate cryptography corresponded to the need for primitives based on new problems. Indeed, today's widely used (cryptographic) schemes are based on the factorisation or on the discrete logarithm problem, and we can not be sure (although it seems very unlikely) that no efficient algorithm to solve these problems exists. Thus, it is natural to look for alternatives.

Considering also that both factorisation and discrete logarithm problems are easily solved on quantum computers, even if operational ones won't exist for decades, one must be prepared.

Multivariate cryptography could be a credible alternative in the sense that the underlying problem (solving a system of polynomial equations over some finite field) is NP-hard, seems to stay hard "on average" and is believed to remain hard even with a quantum computer.

An additional advantage is that it allows the building of efficient schemes that are particularly well suited to low-cost devices such as smartcard.Unfortunately, nowadays, most of the multivariate encryption schemes have been broken. However, several modifications have enabled the creation of a large variety of very efficient signature algorithms. Even if these schemes are quite recent, in 2003 a scheme from Patarin et al. [3] was considered strong enough to be selected by the NESSIE project [1] for standardization. This last scheme was recently broken in [6].

K.-H. Nam and G. Rhee (Eds.): ICISC 2007, LNCS 4817, pp. 27–36, 2007.

After his cryptanalysis of the Imai-Matsumoto cryptosystem [8], Patarin et al. proposed in a series of articles ([12] [7]), a collection of new multivariate schemes which the author called $2R$ (for "2 Round"), one of which is also called the "Double-Round Quadratic cipher" in [10], and which is at the center of the present paper.

A cryptanalysis of a large class of the $2R$ cipher (particularly of the Double-Round Quadratic cipher) was published in 1999 in [5], followed a year later by another cryptanalysis [2] which, however does not apply to the Double-Round Quadratic cipher.

In this article we present another Cryptanalysis of the Double-Round Quadratic Cryptosystem that differs significantly from [5]. Based on the work from [9], we show that the scheme can be described differently (in another mathematical structure) and that in this other formalism, relinearization techniques enable the computing, in polynomial time, of the private keys from the public one. Our proof is also heuristic in the sense that we assume the linearized equations to be independent. This assumption is very general and has been used and verified many times in similar cases. The author feels that the formalism used here is preferable when describing Multivariate ciphers, because it allows us to present it in a unified (and cleaner) way.

**Organisation of the paper**
In the first section, we describe the Double-Round Quadratic cryptosystem. In the second, we introduce the results from [9] which establish an equivalence between a system of polynomials in $n$ variables over a finite field $\mathbb{F}_q$ and an univariate polynomial in the degree $n$ extension of $\mathbb{F}_q$. Then, we recall results based on so called re-linearization technique (mainly from [9]) in section 3. Finally we present the cryptanalysis in section 4, 5, 6 and 7.

## 2 Description of the Double-Round Quadratic Cryptosystem

### 2.1 Notation

$\mathbb{K} = \mathbb{F}_{q^n}$ denotes the field with $q^n$ elements. Moreover, $q$ must satisfy $q = 3 \mod 4$.

Let $(\beta_1, \cdots, \beta_n)$ be a basis of $\mathbb{K}$ seen as a $\mathbb{F}_q$-vector space.

For an element $x$ of $\mathbb{K}$, we will use both vector and field representations to refer to $x$.

We note $\mathbf{x} = x_1\beta_1 + \cdots + x_n\beta_n \in \mathbb{K}$ (with $(x_1, \cdots, x_n) \in (\mathbb{F}_q)^n$) to refer to the field element. We also refer to $x$ according to the vector notation : $\bar{x} = (x_1, \cdots, x_n)$.

### 2.2 Idea of the Cryptosystem

The public key consist of a system of $n$ degree 4 polynomials (in $n$ variables) $P_1(x_1, \cdots, x_n), \cdots, P_n(x_1, \cdots, x_n)$ defined over $\mathbb{F}_q$.

Given a message $\bar{m} = (m_1, \cdots, m_n) \in (\mathbb{F}_q)^n$, the corresponding ciphertext is the vector corresponding to the image of the polynomial system at the point $\bar{m}$. Hence the ciphertext is $\bar{c} = (P_1(m_1, \cdots, m_n), \cdots, P_n(m_1, \cdots, m_n))$.

To produce the public key (i.e. the system of public polynomials) one chooses 3 nonsingular matrices in $M_n(\mathbb{F}_q)$ which are kept secret. Given a vector $\bar{x} = (x_1, \cdots, x_n) \in (\mathbb{F}_q)^n$ one alternatively applies a private linear transformations (with the help of the private matrices) and a public non-linear (invertible) transformation. The result of this mix of (private) linear, and (public) non-linear transformations, gives rise to a system of $n$ polynomials which form the public key.

Finally, to decrypt the vector $\bar{c}$, one has to apply the inverse linear transformations (with the private matrices) and the inverse non-linear transformation, in the reverse order.

The general idea here (and it is the case for all multivariate cryptosystems) is that if a malicious adversary wants to recover the plaintext from the ciphertext $(c_1, \cdots, c_n)$ he has to inverse a system of polynomials of the following form:

$$\begin{cases} P_1(x_1, \cdots, x_n) = c_1 \\ \vdots \\ P_n(x_1, \cdots, x_n) = c_n \end{cases}$$

The hope is that, even if he knows the machinery used to produce the system (except the private keys), he can not effectively inverse it other than by directly solving the public system of equations, which is in general a NP-hard problem (even when the polynomials are quadratic over the finite field $\mathbb{F}_2$).

### 2.3 How It Works

*Scheme of the different parameters used in the Double-Round Quadratic Cryptosystem.*

| *Public* | *Private* |
|---|---|
| $q, n$ | |
| $P_1(x_1, \cdots, x_n)$ | |
| $\vdots$ | $A, B, C \in M_n(\mathbb{F}_q)$ nonsingular |
| $P_n(x_1, \cdots, x_n)$ | |

**Production of the public key**
Let $\bar{x} = (x_1, \cdots, x_n)$. First, one chooses the parameters $n$ and $q$ along with the (nonsingular) matrices $A, B$ and $C$. To produce the public system of equations, one computes the following:

$$\bar{u} = A\bar{x} \Rightarrow \mathbf{v} = \mathbf{u}^2 \Rightarrow \bar{w} = B\bar{v} \Rightarrow \mathbf{z} = \mathbf{w}^2 \Rightarrow \bar{c} = C\bar{z} = \begin{pmatrix} P_1(x_1, \cdots, x_n) \\ \vdots \\ P_n(x_1, \cdots, x_n) \end{pmatrix}.$$

*Note that we alternate between transformations on vectors and transformations on field elements*

**Encryption**
Given the message $\bar{m} = (m_1, \cdots, m_n)$, the ciphertext is $\bar{c} = (P_1(m_1, \cdots, m_n), \cdots, P_n(m_1, \cdots, m_n))$ in $(\mathbb{F}_q)^n$.

**Decryption**
To decrypt, one must be able to inverse all the transformations which led to the public key. The linear transformations are easy to inverse, given the matrix $A, B$ and $C$. The squaring transformation is also easily "almost" invertible as follows:

$$(\mathbf{x}^2)^{\frac{q^n+1}{4}} = \mathbf{x}^{\frac{q^n-1}{2}}\mathbf{x} = \pm\mathbf{x} \text{ ( remember that } q = 3 mod 4)$$

Hence, given the ciphertext $\bar{c}$, the decryption algorithm work as follows:

$$C^{-1}\bar{c}=\bar{z} \Rightarrow \mathbf{z}^{\frac{q+1}{4}} = \pm\mathbf{w} \Rightarrow B^{-1}(\pm\bar{w}) = \pm\bar{v} \Rightarrow (\pm\mathbf{v})^{\frac{q+1}{4}} = \pm\mathbf{u} \Rightarrow A^{-1}(\pm\bar{u}) = \pm\bar{x}$$

We can not exactly inverse the squaring transformation, but it is not really a problem. Hence, if $\bar{m}$ is the original message and $\bar{x} = (x_1, \cdots, x_n)$ is the decrypted one, we know that either $\bar{m} = (x_1, \cdots, x_n)$ or $\bar{m} = (q-x_1, \cdots, q-x_n)$. We can take, for instance, the convention that each message $\bar{m}$ to be encrypted must have the first non zero component $m_i$ in the interval $[1, \frac{q-1}{2}]$, so that we know which of $\bar{x}$ and $-\bar{x}$ correspond to $m$.

*Remarks.* The cryptosystem needs two rounds, i.e. two applications of the (non-linear) squaring transformation. Indeed, the one round version can be attacked, see [12] or the book [10].

## 3 The Kipnis-Shamir Formalism

In the description of the cryptosystem, we alternate between transformations on elements of $(\mathbb{F}_q)^n$ and transformations in $\mathbb{F}_{q^n}$. This different framework makes it difficult to analyse and to attack the cryptosystem. In [9], the authors showed that one could have a unified framework, with only transformations in $\mathbb{F}_{q^n}$. In section 5, we will show that in this framework, the Double-Round Quadratic cryptosystem is easily breakable.

The next two theorems are taken directly from [9]. We add the proof for completeness.

**Theorem 1 (Kipnis, Shamir 99).** *Let $M$ be a linear mapping from $n$-tuples to $n$-tuples of values in $\mathbb{F}_q$. Then there are coefficients $a_1, \cdots, a_n$ in $\mathbb{F}_{q^n}$ such that for any two $n$-tuples over $\mathbb{F}_q$, $(x_1, \cdots, x_n)$ (which represents $\mathbf{x} = \sum_{i=1}^n x_i\beta_i$ in $\mathbb{F}_{q^n}$) and $(y_1, \cdots, y_n)$ (which represents $\mathbf{y} = \sum_{i=1}^n y_i\beta_i$ in $\mathbb{F}_{q^n}$), $(y_1, \cdots, y_n) = M(x_1, \cdots, x_n)$ if and only if $\mathbf{y} = \sum_{i=1}^n a_i\mathbf{x}^{q^i}$.*

**Proof**
It is well known that the application $\mathbf{F}$ with $\mathbf{F}(\mathbf{x}) = \mathbf{x}^{q^i}$ is a linear application in $\mathbb{F}_{q^n}$ (seen as a $\mathbb{F}_q$-vector space). Hence, every polynomial $P(\mathbf{x}) = \sum_{i=0}^{n-1} a_i\mathbf{x}^{q^i} \in \mathbb{F}_{q^n}[X]$ is also a linear application.

It follows that there exists a matrix $M$ in $M_n(\mathbb{F}_q)$ such that for any two $n$-tuples over $\mathbb{F}_q$, $(x_1, \cdots, x_n)$ (which represents $\mathbf{x} = \sum_{i=1}^{n} x_i\beta_i$ in $\mathbb{F}_{q^n}$) and $(y_1, \cdots, y_n)$ (whith $\mathbf{y} = \sum_{i=1}^{n} y_i\beta_i$ in $\mathbb{F}_{q^n}$), $(y_1, \cdots, y_n) = M(x_1, \cdots, x_n)$ if $\mathbf{y} = \sum_{i=1}^{n} a_i\mathbf{x}^{q^i}$.

Moreover there are $q^{n^2}$ different matrices in $M_n(\mathbb{F}_q)$ and also $q^{n^2}$ different Polynomials of the form $P(\mathbf{x}) = \sum_{i=0}^{n-1} a_i\mathbf{x}^{q^i}$ in $\mathbb{F}_{q^n}[X]$. Finally, it is clear that two different polynomials can not lead to the same matrix, since otherwise it would mean that a non-zero polynomial is represented by a zero matrix.

*Remark.* Given a linear mapping $M$, it is clear that we can find the coefficients $a_i$ of the corresponding polynomial by identification. This can be done in time roughly $O(n^5)$.

Now, we can state an even stronger result. Indeed, starting from the previous theorem, once can easily show that any system of $n$ equations in $n$ variables over $\mathbb{F}_q$, can be represented by an equivalent polynomial in $\mathbb{F}_{q^n}[X]$ of a special shape. The next theorem (and its proof) is important for us, as it provides a method of building such an equivalent polynomial (in $\mathbb{F}_{q^n}[X]$).

**Theorem 2 (Kipnis, Shamir 99).** *Let $P_1(x_1, \cdots, x_n), \cdots, P_n(x_1, \cdots, x_n)$ be any set of $n$ multivariate polynomials in $n$ variables over $\mathbb{F}_q$. Then, there are coefficients $a_1, \cdots, a_{q^n}$ in $\mathbb{F}_{q^n}$ such that for any two $n$ tuples $(x_1, \cdots, x_n)$ and $(y_1, \cdots, y_n)$ in $(\mathbb{F}_q)^n$, $y_j = P_j(x_1, \cdots, x_n)$ for all $1 \leq j \leq n$ if and only if $\mathbf{y} = \sum_{i=1}^{q^n} a_i\mathbf{x}^i$ (where $\mathbf{x} = \sum_{i=1}^{n} x_i\beta_i$ and $\mathbf{y} = \sum_{i=1}^{n} y_i\beta_i$ are the elements of $\mathbb{F}_{q^n}$ which correspond to the two vectors over $\mathbb{F}_q$).*

We also reproduce the proof, as it leads to a lemma, which we will use in the cryptanalysis.

**Proof**
The mapping $\overline{\phi_{1,i}} : (x_1, \cdots, x_n) \rightarrow (x_i, 0, \cdots, 0)$ is $\mathbb{F}_q$-linear. Thus, from the first theorem, there exists a corresponding polynomial over $\mathbb{F}_{q^n}$, ie there exists $P_{1,i} \in \mathbb{F}_{q^n}[X]$ such that $\overline{P_{1,i}(\mathbf{x})} = \overline{\phi_{1,i}}(\bar{x})$.

To represent the mapping $(x_1, \cdots, x_n) \rightarrow (\prod_{i=1}^{k} x_i^{c_i}, 0, \cdots, 0)$ we can simply multiply the polynomials $P_{1,i}$ corresponding to each linear transformation.

The polynomial corresponding to the following mapping, $(x_1, \cdots, x_n) \rightarrow (0, \cdots, 0, \prod_{i=1}^{k} x_i^{c_i}, 0, \cdots, 0)$ (where the non-zero component of the image is the component of $\beta_j$) is simply the product of all the polynomials $P_{1,i}^{c_i}$ $(i = 1, \cdots, n)$ multiplied with $\beta_1^{-1}\beta_j$.

This proof leads to the following lemma:

**Lemma 1.** *Let $C$ be any collection of $n$ homogeneous multivariate polynomials of degree $d$ in $n$ variables over $\mathbb{F}_q$. Then, the only powers of $x$ which can occur with non-zero coefficients in its univariate polynomial representation $G(x)$ over $\mathbb{F}_{q^n}$ are sums of exactly $d$ (not necessarily distinct) powers of $q$ : $q^{i_1} + q^{i_2} + \cdots + q^{i_n}$. If $d$ is a constant, then $G(x)$ is sparse, and its coefficients can be found in polynomial time.*

*Remark.* Hence, given a system of $n$ polynomials (in $n$ variables) of maximum total degree $d$ over $\mathbb{F}_q$, we can find the coefficients of the corresponding polynomial over $\mathbb{F}_{q^n}$ easily. Indeed, one can compute all the polynomials $P_{1,i}$ $(i = 1, \cdots, n)$ and then compute their product (as mentioned in the later proof). Another strategy to find the polynomial is by using simple interpolation based on sufficiently many Input/Output pairs.

## 4 Relinearization Technique

In this section, we are interested in solving an overdefined system of quadratic equations in some finite field.

Let us say that this system has $m$ variables, and many more equations, i.e. $\epsilon m^2$ for $\epsilon > 0$ . In [9] the authors demonstrate an easy algorithm to solve this system, if $\epsilon$ is not too small.

We briefly recall this technique, which will be used in the cryptanalysis.

First, if $\epsilon \gtrsim \frac{1}{2}$ the system is easily solvable because one can set new variables $y_{ij} = x_i x_j$. The equations becomes linear in the $\approx \frac{n^2}{2}$ variables $y_{ij}$, and when $\epsilon \gtrsim \frac{1}{2}$ we have more linear equations than variables, we can solve the system using standard linear algebra.

*Remark.* We assume that all (or most) of the linearized equations are linearly independent. Moreover, we assume that if the number of equations is (much) larger than $\frac{n^2}{2}$ we will not have (or will only have very few) parasitic solutions for the $y_{i,j}$ which do not correspond to the solution for $x$ that we are looking for. In [3] the authors have led extensive experiments in the same context, and this heuristic always turned out to be right.

Essentially, if $\epsilon < \frac{1}{2}$ we can still set the new variables $y_{ij}$, but we will have fewer (linear) equations than variables. The set of solutions to this new system is a vector space of dimension $(\frac{1}{2} - \epsilon)m^2$, and we can easily find a basis $(b_1, \cdots, b_{(\frac{1}{2}-\epsilon)m^2})$ of such a space. The particular solution $(y_{11}, \cdots, y_{nn})$ we are looking for can be expressed as a linear combination of the basis element, i.e. $(y_{11}, \cdots, y_{nn}) = \sum_{i=1}^{(\frac{1}{2}-\epsilon)m^2} z_i b_i$. So we have in fact $\epsilon m^2$ equations and $(\frac{1}{2} - \epsilon)m^2$ variables (the $z_i$).

Now we notice that the linearization step also produces new equations, indeed:

$$(x_a x_b)(x_c x_d) = (x_a x_c)(x_b x_d) = (x_a x_d)(x_c x_b) \Rightarrow y_{ab} y_{cd} = y_{ac} y_{bd} = y_{ad} y_{bc}$$

For each sorted list $(a, b, c, d)$ we get 2 equations (for simplicity we neglect the case where $a, b, c, d$ are not distinct), hence we get in total $\approx \frac{m^4}{12}$ new quadratic equations in $y_{ij}$, which lead to the same amount of quadratic equations in the $z_i$.

Now we can linearize these quadratic equations again in the $(\frac{1}{2} - \epsilon)m^2$ variables $z_k$, we get $\frac{m^4}{12}$ linear equations in $\frac{((\frac{1}{2}-\epsilon)m^2)^2}{2}$ variables.

This system is uniquely solvable if $\frac{m^4}{12} \geq \frac{((\frac{1}{2}-\epsilon)m^2)^2}{2}$ which correspond to $\epsilon \gtrsim 0.1$.

Hence we see that for $\epsilon \gtrsim 0.1$ we can solve the system. For a detailed and precise analysis of the method, see [11]. Of course, the method presented above can be generalised in many ways (see [9]) and in [4] it is shown that the heuristical argument works very well in practice. It is also conjectured that one can solve the system for every $\epsilon > 0$ in time $n^{O(\frac{1}{\sqrt{\epsilon}})}$. So, even for $\epsilon$ much smaller than 0.1 this type of method could work.

## 5 Cryptanalysis

### 5.1 Recovering the $C$ Matrix

With the help of the theorems from section 3, we are able to present the Double-Round Quadratic cryptosystem in a unified framework.

The private values (the matrix $A, B$ and $C$) are now the polynomials $P_A, P_B, P_C$ in $\mathbb{F}_{q^n}[X]$ which, according to the first theorem of section 3 are of the following form: $P_A = \sum_{i=0}^{n-1} a_i \mathbf{x}^{q^i}$, $P_B = \sum_{i=0}^{n-1} b_i \mathbf{x}^{q^i}$, $P_C = \sum_{i=0}^{n-1} c_i \mathbf{x}^{q^i}$.

The public system of equations is also represented by a polynomial ( $P_{public}(\mathbf{x})$) in $\mathbb{F}_{q^n}[X]$ and satisfies:

$$P_C(P_B(P_A(\mathbf{x})^2)^2) = P_{public}(\mathbf{x}) \tag{1}$$

Moreover, we see from the shape of $P_A, P_B, P_c$ that $P_{public}$ has the form.

$$P_{public} = \sum_{0 \le i_1 \le i_2 \le i_3 \le i_4 \le n-1} p_{i_1,i_2,i_3,i_4} \mathbf{x}^{q^{i_1}+q^{i_2}+q^{i_3}+q^{i_4}}$$

The shape of the public polynomial in $\mathbb{F}_{q^n}[X]$ can also be viewed as a direct consequence of the lemma from section 3. Moreover, the coefficients $p_{i_1,i_2,i_3,i_4}$ can be found by the constructive method presented in the proof of the theorem (from section 3).

As $P_C$ is the polynomial corresponding to the linear transformation with matrix $C$, it follows that $(P_C)^{-1}$ corresponds to the linear transformation with matrix $C^{-1}$. From the first theorem we know that $(P_C)^{-1}$ is of the form:

$$(P_C)^{-1} = \sum_{t=0}^{n-1} c_t^{'} \mathbf{x}^{q^t}$$

We can write the equation (1) in the following way:

$$P_B(P_A(\mathbf{x})^2)^2 = (P_C)^{-1}(P_{public}(\mathbf{x})) = \sum_{\mathbf{t=0}}^{\mathbf{n-1}} \sum_{\mathbf{0 \le i_1 \le i_2 \le i_3 \le i_4 \le n-1}} \mathbf{c_t^{'} p_{i_1,i_2,i_3,i_4}^{q^t} x^{q^{i_1+t}+q^{i_2+t}+q^{i_3+t}+q^{i_4+t}}} \tag{2}$$

Moreover $P_A(\mathbf{x}) = \sum_{i=0}^{n-1} a_i \mathbf{x}^{q^i} \Rightarrow (P_A(\mathbf{x}))^2 = \sum_{0 \le i \le j \le n}^{n-1} a_{i,j}^{'} \mathbf{x}^{q^i+q^j}$
so $P_B(P_A(\mathbf{x})^2) = \sum_{k=0}^{n-1} b_k [\sum_{0 \le i \le j \le n}^{n-1} a_{i,j}^{'} \mathbf{x}^{q^i+q^j}]^{q^k}$

It follows that there exist $b_{i,j}$ in $\mathbb{F}_{q^n}$ so that $P_B(P_A(\mathbf{x})^2)$ satisfies:

$$P_B(P_A(\mathbf{x})^2) = \sum_{0 \le i_1 \le i_2 \le n-1} b_{i_1,i_2} \mathbf{x}^{q^{i_1}+q^{i_2}} \tag{3}$$

Which means that the following equation must hold:

$$\Big(\sum_{0 \le i_1 \le i_2 \le n-1} b_{i_1 i_2} \mathbf{x}^{q^{i_1}+q^{i_2}}\Big)^2 = \sum_{t=0}^{n-1} \sum_{0 \le i_1,i_2,i_3,i_4 \le n-1} c_t^{'} p_{i_1,i_2,i_3,i_4}^{q^t} \mathbf{x}^{q^{i_1+t}+q^{i_2+t}+q^{i_3+t}+q^{i_4+t}} \tag{4}$$

Hence, we have an equality between two polynomials, which leads to as many equations as the number of different terms in the polynomials. All the terms of the form $\mathbf{x}^{q^{i_1}+q^{i_2}+q^{i_3}+q^{i_4}}$ with $0 \le i_1, i_2, i_3, i_4 \le n-1$ are present.

And clearly if $(i_1, i_2, i_3, i_4)$ (with $i_1 \le i_2 \le i_3 \le i_4$), and $(i_1^{'}, i_2^{'}, i_3^{'}, i_4^{'})$ (with $i_1^{'} \le i_2^{'} \le i_3^{'} \le i_4^{'}$) are different, then $q^{i_1} + q^{i_2} + q^{i_3} + q^{i_4} \neq q^{i_1^{'}} + q^{i_2^{'}} + q^{i_3^{'}} + q^{i_4^{'}}$. So the number of different terms is $\binom{n+3}{4} = \frac{1}{24}n^4 + o(n^4)$, and the number of unknowns (the $b_{ij}$ and $c_k^{'}$) is $\frac{n(n+1)}{2} + n$.

We obtain another system of quadratic equations, but instead of having $n$ equations in $n$ unknown (over $\mathbb{F}_q$) we now have $\approx \frac{1}{24}n^4$ equations, in $\approx \frac{n^2}{2}$ variables (over $\mathbb{F}_{q^n}$). Using the relinearization technique, we can easily solve this system, as in our case $\epsilon \approx \frac{4}{24} = \frac{1}{6} > 0.1$. We recover the matrix $C^{'}$ via the coefficients $c_i^{'}$, and we can compute $C = (C^{'})^{-1}$.

*Remark.* It makes sense to look at the asymptotical values for $n$, because it is the security parameter of the cryptosystems. Which means that $n$ is the parameter to be increased if one wants to keep the overall security for the system with regards to the growth in terms of computational power to perform attacks.
Hence the system is theoretically broken, if it is broken for $n \to \infty$.

In practice, the proposed values were $q = 251$ and $n = 9$, an exact computation leads to $\epsilon = 0.17$ which means that the system is also practically solvable for any values of $n$ (as $\epsilon$ increases with $n$).

### 5.2 Recovering the $B$ and $A$ Matrices

The coefficients $b_{i,j}$ which were found in the previous paragraph lead us to the polynomials $Q(\mathbf{x})$ with:
$Q(\mathbf{x}) = P_B(P_A(\mathbf{x})^2) = \sum_{0 \le i_1 \le i_2 \le n-1} b_{i_1 i_2} \mathbf{x}^{q^{i_1}+q^{i_2}}$.

Now we can use exactly the same method as above to find the matrices $B$ and $A$.

We know that $P_B(\mathbf{x}) = \sum_{i=0}^{n-1} b_i \mathbf{x}^{q^i}$, Hence $P_B^{-1}(\mathbf{x}) = \sum_{i=0}^{n-1} b_i^{'} \mathbf{x}^{q^i}$.
$P_A(\mathbf{x})^2 = P_B^{-1}(Q(\mathbf{x}))$.
Hence, we get (remember $P_A(\mathbf{x}) = \sum_{i=0}^{n-1} a_i \mathbf{x}^{q^i}$):

$$\Big(\sum_{i=0}^{n-1} a_i \mathbf{x}^{q^i}\Big)^2 = \sum_{i=0}^{n-1} b_i (Q(\mathbf{x}))^{q^i}$$

We have a quadratic system of $2n$ variables and $\binom{n+2}{2} = \frac{n^2}{2} + o(n^2)$ equations over $\mathbb{F}_{q^n}$. In this case again $\epsilon = \frac{1}{8} > 0.1$, we can recover the variables (so also the matrices $A$ and $B$) with the relinearization technique mentioned above.

As for the recovering of the matrix $C$ the attack works as well in practice, as for $n = 9$ we find $\epsilon = 0.17$.

## 6 The Affine Case

It is common in Multivariate Cryptography that the private transformations are chosen to be affine (and not linear), because it does not cost more (at least asymptotically), and may enhance the security of the scheme. Here, instead of using only the matrix $A, B$ and $C$, we would also have vectors $A^{'}, B^{'}$ and $C^{'}$ to make these three transformations affine.

It is easy to see that even when the transformations are affine, the same cryptanalysis would work. Indeed, in the Shamir-Kipnis formalism the only change would be to add a constant term to $P_A, P_B$ and $P_C$.

So we would have $P_A(\mathbf{x}) = \sum_{i=0}^{n-1} a_i \mathbf{x}^{q^i} + a_c$, $P_B(\mathbf{x}) = \sum_{i=0}^{n-1} b_i \mathbf{x}^{q^i} + b_c$ and $P_C(\mathbf{x}) = \sum_{i=0}^{n-1} c_i \mathbf{x}^{q^i} + c_c$.

If we use the same technique as above, the number of unknowns and the number of equations changes. Obviously, we have 3 more unknowns (in $F_{q^n}$) in the affine case. On the other hand we have many more equations to be completed. i.e. in the linear case each equation corresponded to one monom of the form $\mathbf{x}^{q^{i_1}+q^{i_2}+q^{i_3}+q^{i_4}}$, whereas in the affine case we also have to take into account all the monoms of the form $\mathbf{x}^{q^{i_1}+q^{i_2}+q^{i_3}}$,$\mathbf{x}^{q^{i_1}+q^{i_2}}$, $\mathbf{x}^{q^{i_1}}$ and the constant term.

Overall, there are many new equations and only 3 new variables, so the same technique (as in the linear case) will also work.

## 7 Complexity Analysis

The relinearization technique has polynomial time complexity. Moreover, we use it for quadratic systems of $O(n^2)$ variables, where $n$ is our security parameter. Hence, our attack is clearly polynomial time, so the system is theoretically broken.

Now let us take a deeper look at the actual complexity. It is known that one can solve a linear system of dimension $m$ over a finite field using Coppersmith/Winograd method in $O(m^{2.4})$. The relinearization seeks to solve a system of dimension roughly $m^4$ hence an overall complexity of $O(m^{10})$.

In the cryptanalysis we use the relinearization with $m = n^2$, hence having a complexity of roughly $O(n^{20})$. The proposed value was $n = 9$, hence the attack is practically feasible.

**Acknowledgements.** We would like to thank the referees for their many helpful comments.

## References


1. Nessie project (2003), https://www.cosic.esat.kuleuven.be/nessie/
2. Biham, E.: Cryptanalysis of Patarin 2-Round Public Key System with S Boxes (2R). In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, Springer, Heidelberg (2000)
3. Courtois, N., Goubin, L., Patarin, J.: SFLASH, a fast asymmetric signature scheme (2003), available at http://eprint.iacr.org/2003/211/
4. 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, Springer, Heidelberg (2000)
5. Din-Feng, Y., K-Yan, L., Zong-Duo, D.: Cryptanalysis of $2R$ Schemes. In: Wiener, M.J. (ed.) CRYPTO 1999. LNCS, vol. 1666, Springer, Heidelberg (1999)
6. Dubois, V., Fouque, P., Shamir, A., Stern, J.: Practical Cryptanalysis of SFLASH. In: Crypto 2007. LNCS, vol. 1807, Springer, Heidelberg (2007)
7. Goubin, L., Patarin, J.: Asymmetric Cryptography with S-Boxes. In: ICICS 1997. LNCS, vol. 1807. Springer, Heidelberg (1997)
8. Imai, H., Matsumoto, T.: Algebraic Methods for Constructing Asymmetric Cryptosystems. In: Calmet, J. (ed.) Algebraic Algorithms and Error-Correcting Codes. LNCS, vol. 229, Springer, Heidelberg (1986)
9. Kipnis, A., Shamir, A.: Cyptanalysis of the HFE Public Key Cryptosystem. In: Wiener, M.J. (ed.) CRYPTO 1999. LNCS, vol. 1666, Springer, Heidelberg (1999)
10. Koeblitz, N.: Algebraic Aspects of Cryptography. Springer, Heidelberg (1998)
11. Moh, T.: The Method of Relinearization of Kipnis and Shamir and its Applications to TTM (1999), available at http://citeseer.ist.psu.edu/371723.html
12. Patarin, J., Goubin, L.: Trapdoor One-Way Permutations and Multivariate Polynomials. In: Han, Y., Quing, S. (eds.) ICICS 1997. LNCS, vol. 1334, Springer, Heidelberg (1997)