Efficient Cryptanalysis of RSE(2)PKC and RSSE(2)PKC

Christopher Wolf, An Braeken, and Bart Preneel

Department Electrical Engineering, ESAT/COSIC,
Katholieke Universiteit Leuven, Kasteelpark Arenberg 10,
B-3001 Heverlee-Leuven, Belgium
{christopher.wolf, an.braeken, bart.preneel}@esat.kuleuven.ac.be

Abstract. In this paper, we study the new class step-wise Triangular Schemes (STS) of public key cryptosystems (PKC) based on multivariate quadratic polynomials. In these schemes, we have m the number of equations, n the number of variables, L the number of steps/layers, r the number of equations/variables per step, and q the size of the underlying field. We present two attacks on the STS class by exploiting the chain of the kernels of the private key polynomials. The first attack is an inversion attack which computes the message/signature for given ciphertext/message in $O(mn^3Lq^r + n^2Lrq^r)$, the second is a structural attack which recovers an equivalent version of the secret key in $O(mn^3Lq^r + mn^4)$ operations. Since the legitimate user has workload q^r for decrypting/computing a signature, the attacks presented in this paper are very efficient. As an application, we show that two special instances of STS, namely RSE(2)PKC and RSSE(2)PKC, recently proposed by Kasahara and Sakai, are insecure.

1 Introduction

1.1 PKC Schemes Based on Multivariate Quadratic Equations

In the last two decades, several public key cryptoschemes (PKC) have been proposed which use $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic equations ($\mathcal{MQ}$) over a finite field $\mathbb{F}$. A typical multivariate PKC public key $\mathcal{P}$ has the structure $S \circ \mathcal{P}' \circ T$. Here, $S \in \mathrm{GL}_n(\mathbb{F})$ and $T \in \mathrm{GL}_m(\mathbb{F})$ represent two linear transformations over the finite field $\mathbb{F}$. The system $\mathcal{P}'$ of m central equations in n variables of degree 2 is constructed with a trapdoor in order to speed up the decryption process. The secret key of the system consists of the triple $(S, \mathcal{P}', T)$. Depending on the structure of $\mathcal{P}'$, these schemes can be divided into several classes: *e.g.*, the initial polynomial substitution scheme from Fell and Diffie [8], C* schemes [17], HFE-like schemes [19, 6] or unbalanced oil-vinegar schemes [14]. All of them rely on the fact that the $\mathcal{MQ}$-problem, *i.e.*, finding a solution $x \in \mathbb{F}^n$ for a given system $\mathcal{P}$ is computationally difficult, namely $\mathcal{NP}$-complete (cf [9–p. 251] and [20–App.] for a detailed proof). Also decomposing $\mathcal{P}$ into $T, \mathcal{P}', S$ — called the Isomorphism of Polynomials Problem — is considered to be a hard problem if $S, \mathcal{P}', T$ do not have a special structure [21].

C. Blundo and S. Cimato (Eds.): SCN 2004, LNCS 3352, pp. 294–309, 2005.
© Springer-Verlag Berlin Heidelberg 2005

In this paper, we concentrate on a special sub-class of $\mathcal{MQ}$-schemes, namely schemes which have a triangular structure for their central equations $\mathcal{P}'$ — triangular schemes for short. This idea is due to Shamir [22] who developed such schemes (Birational Permutations) over large finite rings. To guard against special types of attacks, he removed some initial equations. Goubin *et al.* specialised the approach from [22] to the case of small finite fields, denoted TPM schemes (Triangle Plus Minus, [10]). They add to Shamir's construction some equations in the last step ("Plus" modification) and fall in a similar class as the scheme described in this paper (cf Fig. 2).
We now consider a further generalisation of the Birational Permutation and TPM family. These schemes are called STS (step-wise triangular schemes), which differ from the TPM class by allowing a "step" of more than one variable/equation in the triangular structure (cf Fig. 1 for regular STS). The step-width (number of new variables) and the step-height (number of new equations) is controlled by the parameter r. For Birational Permutations and TPM, the parameter r is fixed to 1. Therefore, they are a special case of STS (cf Sect. 1.2). The main part of this paper consists of the description of two very efficient attacks on STS schemes. They break STS in $O(mn^3Lq^r + mn^4)$ and $O(mn^3Lq^r + n^2Lrq^r)$ — for m the number of equations, n the number of variables, L the number of layers, q the size of the ground field $\mathbb{F}$, and r the step-width/step-hight. The attacks are mainly based on the fact that the kernels of the private central polynomials p'_i form a descending chain of subspaces (cf Sect. 2.1). As the recently proposed schemes RSE(2)PKC and RSSE(2)PKC by Kasahara and Sakai belong to the STS family (cf Sect. 3), both schemes are covered by these attacks and thus highly insecure. As an application of the attacks described in this paper, we broke the challenge for RSE(2)PKC (cf Sect. 3.2).

$$
\begin{array}{l}
\text{Step } 1 \begin{cases} y'_1 = p'_1\,(x'_1,\dots,x'_r) \\ \quad\vdots \\ y'_r = p'_r\,(x'_1,\dots,x'_r) \end{cases} \qquad\qquad \text{with } x'_i \in \mathbb{F} \\
\quad\vdots \\
\text{Step } l \begin{cases} y'_{(l-1)r+1} = p'_{(l-1)r+1}\,(x'_1,\dots,x'_r,\,\dots,x'_{(l-1)r+1},\dots,x'_{lr}) \\ \quad\vdots \\ y'_{lr} = p'_{lr}\,(x'_1,\dots,x'_r,\,\dots,x'_{(l-1)r+1},\dots,x'_{lr}) \end{cases} \\
\quad\vdots \\
\text{Step } L \begin{cases} y'_{(L-1)r+1} = p'_{(L-1)r+1}\,(x'_1,\dots,x'_r,\,\dots,x'_{(l-1)r+1},\dots,x'_{lr},\,\dots,x'_{n-r+1},\dots,x_n) \\ \quad\vdots \\ y'_{Lr} = p'_{Lr}\,(x'_1,\dots,x'_r,\,\dots,x'_{(l-1)r+1},\dots,x'_{lr},\,\dots,x'_{n-r+1},\dots,x_n) \end{cases}
\end{array}
$$

Fig. 1. Central Equations p'_i in a Regular STS Scheme

1.2 Step-Wise Triangular Systems

A step-wise triangular scheme is defined over a finite field $\mathbb{F}$ with $q := |\mathbb{F}|$ elements and prime characteristic char($\mathbb{F}$). Over this field, we define multivariate quadratic polynomials

$$p_i(x_1, \ldots, x_n) := \sum_{1 \leq j \leq k \leq n} \gamma_{i,j,k} x_j x_k + \sum_{1 \leq j \leq n} \beta_{i,j} x_j + \alpha_i \,, \tag{1}$$

for $1 \leq i \leq m$ and $\alpha_i, \beta_{i,j}, \gamma_{i,j,k} \in \mathbb{F}$ (constant, linear, and quadratic terms). These polynomials form the public key as a system of equations $\mathcal{P} := (p_1, \ldots, p_m)$. The plaintext $x \in \mathbb{F}^n$ is transformed to the ciphertext $y \in \mathbb{F}^m$ as

$$y_i := p_i(x_1, \ldots, x_n) \text{ with } 1 \leq i \leq m \,.$$

The decryption, *i.e.*, the inversion of this mapping, uses a trapdoor (cf Fig. 2). This trapdoor consists of two linear transformations $S \in \mathrm{GL}_n(\mathbb{F})$, $T \in \mathrm{GL}_m(\mathbb{F})$ and central equations as outlined in Fig. 1. The public equations $\mathcal{P}$ are constructed as a composition of $\mathcal{P} := T \circ \mathcal{P}' \circ S$ where $\mathcal{P}'$ has a special triangular structure (cf Fig. 1). The two linear transformations may be seen as invertible matrices, we hence have $S \in \mathbb{F}^{n \times n}$ and $T \in \mathbb{F}^{m \times m}$, respectively. In our description, we always use a prime ($'$) for denoting the secret central part of the system.

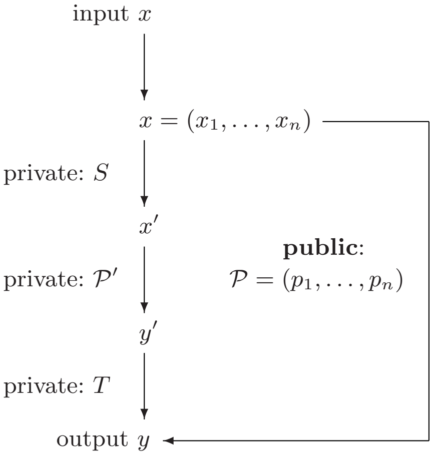

Fig. 2. $\mathcal{MQ}$-trapdoor $(S, \mathcal{P}', T)$ in STS

Let $r_1, \ldots, r_L$ be L integers such that $r_1 + \cdots + r_L = n$, the number of variables, and $m_1, \ldots, m_L \in \mathbb{N}$ such that $m_1 + \cdots + m_L = m$, the number of equations. Here $L \in \mathbb{N}$ denotes the number of layers or steps in the scheme, r_l represents the number of new variables (step-width) and m_l the number of equations (step-height), both in step l for $1 \leq l \leq L$. In a general step-wise Triangular Scheme (gSTS), the m_l private quadratic polynomials of each layer l, contain only the variables x'_k with $k \leq \sum_{j=1}^{l} r_j$, *i.e.*, only the variables defined in all previous steps plus r_l new ones. The overall shape of the private polynomials leads to the name step-wise Triangular Scheme (STS).

When not mentioned otherwise, we concentrate on regular STS schemes (rSTS or STS for short) in this paper. For regular STS schemes we set $r_1 = \cdots = r_N = m_1 = \cdots = m_L$, which we denote by r. Moreover, $L = m/r$ and $m = n$. Note that the attacks we propose are also valid for the general STS schemes (cf Sect. 4.1). The structure of a regular STS has been outlined in fig. 1 and 2.

As we see in Fig. 1, there are exactly r new variables in an rSTS for each layer. This way one can compute an x for a given vector y with q^r attempts in each step. But as the legitimate user has a workload growing exponentially in r, this value has to be small in order to obtain a scheme of practical interest. The parameter r plays an important role for the complexity of our attack.

In order to decrypt a given ciphertext y, we need to invert the following steps: $x \xrightarrow{S} x' \xrightarrow{\mathcal{P}'} y' \xrightarrow{T} y$. While S, T are bijections and also easy to invert, this is not so obvious for the central equations $\mathcal{P}'$. In particular, these central equations may not form a bijection. Adding redundancy to the original message x or transmitting some additional redundancy, *e.g.*, in form of its hash-value $h := H(x)$ where $H(\cdot)$ denotes a cryptographically secure hash function (*e.g.*, see [18]), allows to pick the correct message x for a given input y. For a signature scheme, we do not need this redundancy as it is enough to obtain one $x \in \mathbb{F}^n$ such that $\mathcal{P}(x) = y$ for a given y; in most cases, this will be the hash of a longer message. As this point is not important for our attack, we refer to [19, 12] for a broader discussion of this problem.

Remark: As already pointed out in the introduction, the Birational Permutation Schemes of Shamir are regular STS schemes with $r = 1$. However, they are not defined over a (small) finite field but over a (large) finite ring. The TPM class of Goubin and Courtois coincides with STS for the parameters $r_1 = u$, $m_L = v$, $m_1 = \cdots = m_{L-1} = r_2 = \cdots = r_L = 1$, *i.e.*, we remove $u \in \mathbb{N}$ initial layers, add $v \in \mathbb{N}$ polynomials in the last step, and have exactly one new variable at all intermediate levels. As STS, this class is not defined over a ring but over a field.

Shamir's scheme was broken shortly after its publication in [2, 23, 3]. The TPM scheme of Goubin and Courtois has been broken in the paper that proposed it [10]. In fact, the aim of their construction was to show that Moh's TTM construction is weak. While we dwell on the basic ideas of the above attacks, it is necessary to extend them as they are not directly applicable to STS. In particular, Kasahara and Sakai conclude (cf [13–Sect. 4.3.III] and [12–Sect. 4.1.III]) that their constructions are secure against all known attacks — in particular, mentioning [10]. Although this observation is true, we will show in Sect. 2 that it is possible to generalise these attacks in a way that STS and consequently RSE(2)PKC and RSSE(2)PKC are covered by them, too.

1.3 Organisation

This paper is organised as follows: after this introduction, we move on to a cryptanalysis of regular STS schemes, showing both an inversion and a structural attack in Sect. 2. The following section deals with special instances like RSE(2)PKC and RSSE(2)PKC. In Sect. 4, we generalise STS. This paper concludes with Sect. 5.

2 Cryptanalysis

We now present two different types of attacks on STS. In the inversion attack (cf Sect. 2.3), we recover for given ciphertext y the corresponding message x. In the structural attack (cf Sect. 2.4), we build a linear equivalent version of the private key, denoted $(\tilde{S}, \tilde{\mathcal{P}}', \tilde{T})$. Using $(\tilde{S}, \tilde{\mathcal{P}}', \tilde{T})$, the attacker is in the same position as the legitimate user for deciphering a given message y or forging a signature on it. For both attacks, we first need some observations on kernels.

2.1 Chain of Kernels

Let p_i be a public key polynomial. For characteristic $\neq 2$, we can uniquely express its homogeneous quadratic parts in a symmetric matrix $P_i \in \mathbb{F}^{n\times n}$. We show this with a toy-example with three variables:

$$\begin{pmatrix} \gamma_{1,1} & \frac{\gamma_{1,2}}{2} & \frac{\gamma_{1,3}}{2} \\ \frac{\gamma_{1,2}}{2} & \gamma_{2,2} & \frac{\gamma_{2,3}}{2} \\ \frac{\gamma_{1,3}}{2} & \frac{\gamma_{2,3}}{2} & \gamma_{3,3} \end{pmatrix} ,$$

where the $\gamma_{i,j}$ represent the quadratic coefficients of $x_i x_j$ from the public polynomials as defined in (1). So, instead of evaluating the quadratic parts of p_i by the vector x, we may also perform xP_ix^t as matrix-vector multiplications (here ^{t} denotes transposition). As division by 2 is not defined for characteristic 2, we use the form $P_i := L_i + L_i^t$ for lower triangular matrices L_i instead to obtain unique symmetric matrices. This way, we loose the quadratic coefficients $\gamma_{i,i}$ of the public polynomials. However, in characteristic 2, these quadratic terms are linear and we can therefore ignore them. To the knowledge of the authors, the above observation has been initially reported in [14] and is there credited to *Don Coppersmith.*

The private key polynomials p'_i may also be represented in the above matrix form. Following the notation outlined in the previous section, we denote the corresponding matrices P'_i. Obviously, the rank of each such matrix depends on its layer l. The matrices P'_i have a rank of rl in each layer l for $1 \leq l \leq L$ and we have

$$\ker'_l = \{a' \in \mathbb{F}^n | \ a'_1 = \ldots = a'_{rl} = 0\}$$

as common kernels of the matrices P'_i for $(l-1)r < i \leq lr$. As these kernels are hidden by the linear transformation S, we also mark them with a prime $'$. Moreover, we denote by $a'_i \in \mathbb{F}$ for $1 \leq i \leq n$ the coefficients of the vectors $a' \in \mathbb{F}^n$.

We now study the effect of the linear transformation S, *i.e.*, the change of variables. As we have $\hat{p_i} := p'_i \circ S$ and $x' = S(x)$, we obtain $\hat{P_i} := SP'_iS^t$ in terms of the corresponding matrices. As S is invertible, we have $\text{Rank}(\hat{P_i}) = \text{Rank}(P'_i)$ and

$$\ker_l = \{a'S^{-1} \mid a' \in \mathbb{F}^n \wedge a'_1 = \ldots = a'_{rl} = 0\} \tag{2}$$

for the kernels of $\hat{P_i}$ for $(l-1)r < i \leq lr$ and an unknown matrix S. Moreover,

$$\ker'_L \subset \ldots \subset \ker'_1 \text{ and consequently } \ker_L \subset \ldots \subset \ker_1 \ .$$

With the notation $T = (\tau_{i,j})_{1 \leq i,j \leq m}$, each individual public key matrix P_i can be expressed by

$$P_i = \sum_{j=1}^{m} \tau_{i,j}[SP'_iS^t] = \sum_{j=1}^{m} \tau_{i,j}\hat{P_i} \, .$$

The problem of finding the transformation T^{-1} and thus T has therefore been reduced to finding a linear combination of the public key (in matrix notation) which has a specific rank. In the following two subsections, we describe two algorithms which can be used for this purpose.

2.2 Recovering the Transformation T

Attacking the High-Rank Side. We start with an attack on the high-rank side (cf the algorithm in Fig. 3). The overall idea of this algorithm is to exploit the step-structure of STS. To do so, we observe that a correct random guess of a row-vector in T^{-1} will lead to a condition on the rank of the linear combination of the corresponding public key equations — expressed in matrix notation. More formally and also to verify the correctness of this algorithm, we consider the vector spaces

$$J_l := \{b'T^{-1} \mid b' \in \mathbb{F}^m \wedge b'_{lr+1} = \ldots = b'_m = 0\} \text{ for } 1 \leq l \leq L \ . \tag{3}$$

Obviously, they form a descending chain of subspaces and each of them has dimension $m - lr$. Therefore, when picking a random element $v \in_R J_{l+1}$, we have a probability of q^{-r} that the expression $v \in J_l$ holds. In addition, we have two efficient methods (`matrixCheck` or `polynomialCheck`, respectively) to check whether $v \in J_l$ or $v \notin J_l$. First, we concentrate on `matrixCheck`:

$$\texttt{matrixCheck}(P_1, \ldots, P_m,\ v, l) \text{ returns } \textbf{true} \text{ iff } Rank(\sum_{i=1}^{m} v_iP_i) \leq lr \ .$$

For the sake of the argument, we look at the problem in the T^{-1}-space, *i.e.*, after the linear transformation T^{-1} has been applied. Using the notation from (3), we consider vectors b' instead of v. Hence we have

$$M := \sum_{i=1}^{m} b'_i\hat{P_i} = \sum_{i=1}^{rl} b'_i\left(SP'_iS^t\right) = S\left(\sum_{i=1}^{rl} b'_iP'_i\right)S^t \ .$$

Observing the step-wise structure of the private key polynomials p'_i we conclude that the $\text{Rank}(M) \leq lr$. This yields the result.
The expected running time of the algorithm from Fig. 3 is therefore bounded by $O(mn^3Lq^r)$: by picking at most cmq^r vectors for each layer (c being a small constant, *e.g.*, 10), we can compute the vector spaces $J_1, \ldots, J_L$ with very high probability. Checking the matrix condition costs an additional factor of n^3 as we are processing matrices from $\mathbb{F}^{n\times n}$. In comparison, the running time of the other steps of the algorithm are negligible.

```
**procedure** highRankAttack($\mathcal{P}$)
  Input:   $\mathcal{P}$: system of public equations
  Output: $\tilde{T}$: an equivalent copy of the transformation $T$
  $P_i \leftarrow$ computeMatrix($p_i$); $J_L \leftarrow \mathbb{F}^m$
  **for** $l \leftarrow L - 1$ **downto** 1 **do**
    $J_l \leftarrow \{0\}$
    **repeat**
      $v \in_R J_{l+1}$
      **if** matrixCheck($P_1, \ldots, P_m,\ v, l$) $\vee$ polynomialCheck($p_1, \ldots, p_m,\ v, l$) **then**
        $J_l \cup \leftarrow \{v\}$
    **until** Dimension($J_l$) $\overset{?}{=} lr$
    $\tilde{J} \leftarrow J_{l+1} \cap J_l$
    **for** $i \leftarrow 1$ **to** $r$ **do**
      RowVector($\hat{T}, lr + i$) $\leftarrow$ BasisVector($\tilde{J}, i$)
  **endfor**
  **return** $\tilde{T} \leftarrow \hat{T}^{-1}$
**endproc**
```

Fig. 3. High-Rank Algorithm for Computing the Transformation $\tilde{T}$ for a Given System of Equations

In characteristic 2 we may apply Dickson's theorem instead to check directly for a given polynomial if it may be reduced to a form with less variables (procedure `polynomialCheck`). Unfortunately, the proof is a bit lengthy, we therefore refer to [16–Sec. 15.2, Thm. 4] for both the theorem and its proof. An algorithmic version of it can be found in [4–Sec. 3.2]. The time complexity of this algorithm is there estimated to be $O(n^3)$. Therefore, the overall complexity of the above algorithm remains the same: $O(mn^3Lq^r)$.

Remark: In both cases, we will not be able to recover the original transformation T but the inverse of a linear equivalent copy of it, denoted $\hat{T}$ for the inverse and $\tilde{T}$ for the linear equivalent of T. In fact, we will recover versions of T in which the rows of $\tilde{T}$ are linear combinations of the rows of T within the same layer.

Attacking the Low-Rank Side. Instead of obtaining an equivalent copy of the transformation T directly, we can also exploit the fact that the kernels $K_i := \ker_i$ (cf (2)) form a descending chain — starting with the large kernel $\ker_1$. This algorithm (cf Fig. 4) is a little more subtle as it makes use of two different observations. The first one is that the kernels $\ker_i$ form a descending chain.

```
procedure lowRankAttack($\mathcal{P}$)
  Input:   $\mathcal{P}$: system of public equations
  Output: $\tilde{T}$: an equivalent copy of the transformation $T$
  $P_i \leftarrow$ computeMatrix($p_i$); $K_0 \leftarrow \mathbb{F}^n$; $J_0 \leftarrow \{0\}$
  for $l \leftarrow L$ downto 1 do
    repeat
      $w \in_R K_{l-1}$
      $J_l \leftarrow$ SolutionSpace($\sum_{i=1}^{m} v_i(wP_i) = 0$) for an unknown $v \in \mathbb{F}^m$
    until Dimension($J_l$) $\overset{?}{=} lr$.
    $\tilde{J} \leftarrow J_l \cap \overline{J_{l-1}}$
    for $i \leftarrow 1$ to $r$ do
      $\hat{t} \leftarrow$ BasisVector($\tilde{J}, i$); RowVector($\hat{T}, lr + i$) $\leftarrow \hat{t}$; $\hat{P}_{(l-1)r+i} \leftarrow \sum_{j=1}^{m} \hat{t}_j P_j$
    $K_l \leftarrow$ Kernel($P_{lr}$)
  endfor
  return $\tilde{T} \leftarrow \hat{T}^{-1}$
endproc
```

Fig. 4. Low-Rank Algorithm for Computing the Transformation $\tilde{T}$ for a Given System of Equations

Therefore, setting $\ker_0 := \mathbb{F}^n$, the statement $w \in \ker_l$ is true with probability q^{-r} for all $w \in_R \ker_{l-1}$ and $1 \leq l \leq L$. Second, the linear equation $\sum_{i=1}^{m} v_i(wP_i) = 0$ has q^{lr} solutions for unknown $v \in \mathbb{F}^m$ if and only if the vector w is in the kernel $\ker_l$. With $\tilde{J} := J_l \cap \overline{J_{l-1}}$ where $\overline{J_{l-1}}$ denotes the complement of the vector space J_{l-1}, we obtain dimension r for $\tilde{J}$ which yields r new linearly independent rows of the matrix T^{-1}. The algorithm will therefore terminate with a correct solution $\tilde{T}$ after $O(Ln^3q^r)$ steps on average. Thus it outperforms the algorithm from the previous section by a factor of m. As for the previous algorithm, we will not recover the original transformation T but an equally useful variant of it.
Remark: Specialised versions of the algorithms from fig. 3 and 4 can be found in [10] for the case of schemes with step-width 1 of the intermediate layers.

2.3 Inversion Attack

In the previous section, we discussed two different approaches to recover a linear transformation $\tilde{T}$ for given public key equations. In this section, we will use $\tilde{T}$ and the polynomials $\hat{p}_i := \tilde{T}^{-1} \circ p_i$ to solve the problem $y = \mathcal{P}(x)$ for a given vector $y \in \mathbb{F}^m$, *i.e.*, for the $\mathcal{MQ}$-problem. We do so by computing a successive affine approximation of x, cf Fig. 5. This algorithm exploits the fact that the kernels $K_i := \ker_i$ for $1 \leq i \leq L$ have the form $\ker_l = \{a'S^{-1} \mid a' \in \mathbb{F}^n \wedge a'_1 = \ldots = a'_{rl} = 0\}$. Setting $K_0 := \mathbb{F}^n$ we have

$$\tilde{K}_l = K_{l-1} \cap \overline{K_l} = \{a'S^{-1} \mid a' \in \mathbb{F}^n \wedge a'_1 = \ldots = a'_{(l-1)r} = a'_{lr+1} = \ldots = a'_n = 0\}$$

for $1 \leq l \leq L$. Using this observation, we can "switch on" groups of r (hidden) variables x' and therefore manipulate the output of the polynomials $\hat{p}_i$ layer by layer. This is possible although we do not know the actual value of the secret matrix S. As the polynomial system $\hat{\mathcal{P}}$ inherits the layer structure of the original private polynomial system $\mathcal{P}'$, the solutions form a chain of affine subspaces $x + < K_l >$ — where K_l has dimension $n - rl$ in step l. Therefore, we learn $r \log_2 q$ bits about the vector x for each level of recursion.

```
procedure inversionAttack($\mathcal{P}$, $\tilde{T}$, $K_1, \ldots, K_L$, $y$)
  Input:   $\mathcal{P}$: system of public equations, $\tilde{T}$: linear transformation,
           $K_1, \ldots, K_L$: descending chain of kernels, $y$: target-value
  Output:  $X$: a set of solutions for the problem $y = \mathcal{P}(x)$

  procedure recursivePart($x$, $l$)
    if $l > L$ then return $\{x\}$
    $\tilde{K} \leftarrow K_{l-1} \cap \overline{K_l}$; $X \leftarrow \emptyset$
    for $w \in \tilde{K}$ do
      if $(\hat{p}_i(x + w) \overset{?}{=} \tilde{y}_i \;:\; (l-1)r < i \leq lr)$ then $X \cup \leftarrow$ recursivePart($x + w$, $l$)
    return X
  endproc

  $\hat{p}_i \leftarrow p_i \circ \tilde{T}^{-1} : 1 \leq i \leq m$
  $\tilde{y} \leftarrow y\tilde{T}^{-1}$; $K_0 \leftarrow \mathbb{F}^n$
  return recursivePart(0,1)
endproc
```

Fig. 5. Inversion Attack for $y = \mathcal{P}(x)$ and Given $\tilde{T}$

With this inversion attack, we are now in a similar position as the legitimate user: at each level, we have to try cq^r possible vectors and to evaluate r polynomials $\hat{p}_i$ — each step costing $O(rn^2)$. In case the STS is not a bijection, we may need to branch — but this is the same situation as for the legitimate user. The only additional overhead is the computation of the complement of vector spaces and to intersect them. Both can be done in $O(n^2)$. Assuming that $\mathcal{P}$ is a bijection, one application of this inversion attack has time-complexity $O(n^2 Lrq^r)$.

2.4 Structural Attack

The starting point of the structural attack (cf Fig. 6) is the same as for the inversion attack, namely $ker_1 \supset \ldots \supset ker_L$. As we have computed the transformation $\tilde{T}$ in the previous step, we are able to compute the system of equations $\hat{\mathcal{P}}$, the corresponding matrices $\hat{P}_l$ and therefore their kernels for each layer $l : 1 \leq l \leq L$. Due to its internal structure, the vector space $\tilde{K} := K_{l-1} \cap \overline{K_l}$ consists of exactly r row-vectors of $\tilde{S}^{-1}$. We recover them in the for loop. As soon as we have recovered $\tilde{S}$, we apply it to the intermediate system of equations $\hat{P}$, yielding $\tilde{\mathcal{P}}$', an equivalent copy of the private key polynomials.

In terms of complexity, the second step of the structural attack is dominant: we need to evaluate m polynomials with $O(n^2)$ quadratic terms each. As each quadratic term has two variables, this costs $O(n^2)$ for each term. The overall time complexity is therefore $O(mn^4)$. So depending on the value q^r, either the structural or the inversion attack has a lower asymptotic running time as the constants are in the same range.

```
procedure structuralAttack($\hat{\mathcal{P}}$, $K_1, \ldots, K_L$)
  Input:  $\hat{\mathcal{P}}$: system of equations; $K_1, \ldots, K_L$: descending chain of kernels
  Output: $\tilde{S}$: an equivalent copy of the secret transformation $S$
          $\tilde{\mathcal{P}}'$: an equivalent copy of the private key polynomials
  $K_0 \leftarrow \mathbb{F}^n$
  for l ←1 to $L$ do
    $\tilde{K} \leftarrow K_{l-1} \cap \overline{K_l}$
    RowVector($\hat{S}$, $(l-1)r+i$) ←BasisVector($\tilde{K}, i$) : $1 \le i \le r$
  $\tilde{S} \leftarrow \hat{S}^{-1}$
  $\tilde{p}'_i \leftarrow \hat{p}_i \circ \tilde{S}^{-1}$ : $1 \le i \le m$
  return $\tilde{S}, \tilde{P}'$
endproc
```

Fig. 6. Structural Attack for a Given Sequence of Kernels $\ker_1, \ldots, \ker_L$

3 Special Instances of STS

In this section, we show that the two schemes RSE(2)PKC [13] and RSSE(2)PKC [12], recently proposed by Kasahara and Sakai, are special instances of STS — and will therefore fall for the attacks discussed in the previous section. In particular, we were able to break the challenge proposed in [13–Sect. 6] using an inversion attack (cf Sect. 2.3) in both cases.

3.1 RSSE(2)PKC

In RSSE(2)PKC, the private polynomials p'_i for $1 \le i \le r$ have a special form, namely

$$p'_{(l-1)r+i}(x') := \phi_{l,i}(x'_{(l-1)r+1}, \ldots, x'_{lr}) + \psi_{l,i}(x'_1, \ldots, x'_{(l-1)r}) \text{ for } 1 \le l \le L\,,$$

where $\phi_{l,i}$ and $\psi_{l,i}$ are random quadratic polynomials over $\mathbb{F}$ in r and $(l-1)r$ variables, respectively. In both cases, the constant part is omitted. To simplify programming, the linear terms βx_i are considered to be quadratic terms βx_i^2, for all $i \in \{1, \ldots, n\}$. This may be done as RSSE(2)PKC is defined over GF(2) and we hence have $x^2 = x$ for all $x \in$ GF(2).

We observe that this special construction of the private key polynomials does not affect our attack. In particular, the maximum rank for the corresponding matrices P'_i stays the same, namely lr for each layer. Unfortunately, for small values of r (in particular, $2 \le r \le 4$), there is a high probability that two polynomials $\phi_{l,i}, \phi_{l,j}$ for $i \neq j$ have the same coefficients: for $r = 2$, there is only one non-linear coefficient, for $r = 3$, there are only 3, and for $r = 4$, we obtain 6. The corresponding probabilities are therefore $2^{-1}, 2^{-3}$ and 2^{-6}, respectively, that the polynomials $\phi_{l,i}, \phi_{l,j}$ share the same quadratic coefficients. In a linear combination of these two polynomials, the rank of the corresponding matrix will therefore drop by r. This change defeats the `lowRank` algorithm from Fig. 4 as it only uses the matrix representation of the public key polynomials p_i. That way, it will not only find solutions of the layer l, but also for such linear combinations. To attack RSSE(2)PKC, it is therefore advisable to use the `highRank` algorithm from Fig. 3 in connection with Dickson's theorem (cf Sect. 2.2).

3.2 RSE(2)PKC

The system RSE(2)PKC is a special case of RSSE(2)PKC: the polynomials $\phi_{l,i}$ are required to be step-wise bijections, *i.e.*, we have $(\phi_{l,1}, \ldots, \phi_{l,r}) : \mathbb{F}_2^r \to \mathbb{F}_2^r$ is a bijection for all $l \in \{1, \ldots, N\}$. This way, the whole system $\mathcal{P}$ becomes a bijection and it is possible to recover the solution x step by step without any ambiguity. As being a bijection is a rather strong requirement for a system of multivariate polynomials, the problem described in the previous section becomes more severe as we have far less choices for the coefficients in the quadratic terms. Still, using the high-rank rather than the low-rank attack should overcome this problem.

In [13–Sect. 3.2], the authors suggest $r \leq 10$ for their scheme which leads to $q^r = 2^{10}$. Therefore, we expect all attacks from the previous section to be efficient against these schemes.

Challenges. In [13–Sect. 6], Kasahara and Sakai propose two challenges with the following parameters: $\mathbb{F} = \mathrm{GF}(2)$, $n = 100$ and $r = 4, 5$. Using a (highly unoptimised) Magma [1] programme, we were able to break this challenge in a few hours on an AMD Athlon XP 2000+. For our attack, we implemented the inversion attack against the low-rank side (cf sect. 2.2 and 2.3). As pointed out earlier, the attack should have been more efficient using an attack against the high-rank side in combination with Dickson's theorem (cf Sect. 2.2). In particular, we computed the solution x for the given value y. The two solutions are (in vector-notation, starting with x_1 at the left):

- $r = 4$: (0 0 1 1 0 1 0 0 1 0 0 0 0 1 1 0 0 0 1 1 1 0 0 1 1 0 0 1 0 0 0 1 1 1 0 0 0 1 1 1 1 1 0 0 1 1 1 0 1 0 0 0 0 0 1 1 0 1 1 0 0 0 1 0 0 1 1 1 1 1 0 0 0 1 1 1 0 0 1 0 1 1 1 1 1 1 0 0 1 0 0 1 1 0 1 0 1 0 0 1),

- $r = 5$: (1 1 1 0 0 1 1 0 1 0 1 1 1 0 0 0 1 0 0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 1 0 0 1 1 0 0 0 0 1 0 1 0 1 1 0 0 1 0 1 1 1 0 0 1 0 1 0 1 1 0 1 1 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 0 1 0 1 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1).

These results have been confirmed by Kasahara and Sakai [11].

Apart from the attacks presented in this paper, we also want to point out that the generic birthday attack for signature schemes applies against the parameter choice $q = 2$ and $n = 100$. In this case, the workload becomes only

$O(2^{50})$. As Kasahara and Sakai do not use special constructions as, *e.g.*, Feistel-Patarin-Networks [5], the generic birthday attack applies against RSE(2)PKC, RSSE(2)PKC, and also the hybrid type construction from the following section.

3.3 Hybrid Type Construction

In [12–Sect. 4.2], Kasahara and Sakai propose a so-called “hybrid type construction” to enhance the security of RSSE(2)PKC. To simplify explanation, we restrict to the case with two branches as this is sufficient to point out its vulnerability to the attacks described in this paper.

In this case, the private polynomials p'_i are partitioned into two sets: the polynomials $p'_1, \ldots, p'_{m/2}$ are constructed as for RSSE(2)PKC (see above). However, the construction of the other polynomials now involves a third type of polynomial, denoted σ. For $L/2 < l \leq L$ and $1 \leq i \leq r$ we have:

$$p'_{lr+i}(x') := \phi_{l,i}(x'_{(l-1)r+1}, \ldots, x'_{lr}) + \psi_{l,i}(x'_1, \ldots, x'_{(l-1)r}) + \sigma_{lr+i}(x'_1, \ldots, x'_{(L/2)}) \ .$$

As for $\phi_{l,i}$ and $\psi_{l,i}$, the polynomials σ_{lr+i} are quadratic polynomials with randomly chosen coefficients and no constant term α. All of them depend on the first $L/2$ variables only. Therefore, the overall structure of the private polynomials p'_i in terms of the rank of their matrix representation P'_i does not change and the attacks of this paper are still applicable.

4 Extensions of STS and Their Vulnerabilities

4.1 General Step-Wise Triangular Systems

As outlined in Sect. 1.2, regular STS may be generalised by different step-sizes and also different number of equations in each individual level, denoted $r_1, \ldots, r_L \in \mathbb{N}$ and $m_1, \ldots, m_L \in \mathbb{N}$, respectively. Moreover, we may consider these L-tuples as part of the private key; only their sums n and m are public. However, the internal structure of the private key keeps the same, in particular, we still obtain the chain of kernels of the private key polynomials. The only part of the attack we have to be careful about are the values r_1 and m_L, *i.e.*, the number of variables in the first layer and the number of equations in the last layer. If the first is too large, the attack at the low-rank side is no longer effective while a high value of the latter may preclude the attack from the high-rank side. Using gSTS for a signature scheme allows us $r_1 \gg m_1$. However, in this case we may not allow $r_L \ll m_L$ as this leads to a highly overdetermined system of equations — which has only very few solutions on average. The situation is reverse for encryption schemes. Here, we may have $r_L \ll m_L$ but not $r_1 \gg m_1$. As the system has a solution for $y = \mathcal{P}(x)$ by construction, a large value of m_L does not provide a problem here. Unfortunately, we are not able to find it back if the value for r_1 and consequently q^{r_1} is too large.

Therefore, gSTS will either fall to an attack from the high-rank or from the low-rank side. In both cases the construction is insecure. We want to point out that gSTS is a generalisation of the Triangular Plus-Minus (TPM) construction.

In particular, we relax the condition that there is only one new variable and one new equation at each intermediate level (cf Sect. 1.2).

4.2 Affine Transformations

In an attempt to strengthen gSTS, we investigate the replacement of the linear transformations S, T by affine ones, *i.e.*, to include additional vectors $v_s \in \mathbb{F}^n$ and $v_t \in \mathbb{F}^m$.

Consider two affine transformations $S \in \mathrm{AGL}_n(\mathbb{F})$ and $T \in \mathrm{AGL}_m(\mathbb{F})$. Then there exists a unique, invertible matrix $M_S \in \mathbb{F}^{n\times n}$ (resp. $M_T \in \mathbb{F}^{m\times m}$) and an unique vector $v_s \in \mathbb{F}^n$ (resp. $v_t \in \mathbb{F}^m$) which describes the affine transformation S (resp. T) by $S(x) = M_S x + v_s$ where $x \in \mathbb{F}^n$ is an input vector (resp. $T(x) = M_T x + v_t$ for $x \in \mathbb{F}^m$). Moreover, we can rewrite the affine transformation S as $S(x) = (\overline{x} + v_s) \circ (M_S x)$ where $\overline{x}$ denotes the output of $M_S x$. In addition, we can rewrite the affine transformation T as $T(x) = (M_T \hat{x}) \circ (x + M_T^{-1} v_t)$, where $\hat{x}$ denotes the output of $x + M_T^{-1} v_t$. As M_T is an invertible matrix, the matrix $M_T^{-1} \in \mathbb{F}^{m\times m}$ exists and is unique. We now express the public key as a composition of the private key

$$\begin{aligned}\mathcal{P} &= T \circ \mathcal{P}' \circ S\\ &= [(M_T \hat{x}) \circ (\tilde{x} + M_T^{-1} v_t)] \circ \mathcal{P}' \circ [(\overline{x} + v_s) \circ (M_S x)]\end{aligned}$$

where $\tilde{x}$ is the output of $\mathcal{P}' \circ [(x' + v_s) \circ (M_S x)]$ and $\hat{x}$ is the output of $(\tilde{x} + M_T^{-1} v_t) \circ \mathcal{P}' \circ [(x' + v_s) \circ (M_S x)]$. We have

$$\begin{aligned}\mathcal{P} &= (M_T \hat{x}) \circ [(\tilde{x} + M_T^{-1} v_t) \circ \mathcal{P}' \circ (\overline{x} + v_s)] \circ (M_S x)\\ &= (M_T \hat{x}) \circ \mathcal{P}'' \circ (M_S x)\end{aligned}$$

for some system of equations $\mathcal{P}''$. As both $(\overline{x} + v_s)$ and $(\tilde{x} + M_T^{-1} v_t)$ are transformations of degree 1, they do not change the overall degree of $\mathcal{P}''$, *i.e.*, as $\mathcal{P}'$ consists of equations of degree 2 at most, so will $\mathcal{P}''$. In addition, due to its construction, $(M_S, \mathcal{P}'', M_T)$ forms a private key for the public key $\mathcal{P}$ and the layer-structure of STS is not affected by these two operations.

Therefore, we can conclude that the use of affine instead of linear transformations does not enhance the overall security of STS. In fact, we are able to draw a similar conclusion for all such systems — as long as it is possible to replace the equation $\mathcal{P}'$ by an equation of similar shape. The corresponding observation for HFE has been made by Toli [24].

4.3 Degree Larger Than 2

In [13] and [12], Kasahara and Sakai generalise their construction to the schemes RSE(d)PKC and RSSE(d)PKC where $d \in \mathbb{N}$ denotes the degree of the public polynomials and $d \geq 2$. In their construction, terms of all degrees $1, \ldots, d$ appear in the public polynomials, *e.g.*, linear and quadratic terms in RSSE(2)PKC and RSE(2)PKC (cf sect. 3.2 and 3.1). Therefore, we may apply the structural attack using the degree 2 terms in RSSE(d)PKC for $d > 2$, consequently retrieving the

transformations $\tilde{S}$ and $\tilde{T}$, and then the corresponding private polynomials in the larger degree d. Similar, we may apply the inversion attack.
This construction is therefore not more secure. In addition, it leads to a much larger public key: the number of terms grows in $O(mn^d)$ for $d > 2$.

4.4 Highly Overdetermined Schemes

When the scheme has more equations than variables, *i.e.*, for $m > n$, we need to adapt the algorithm `LowRankAttack` (cf Section 2.2). Instead of picking one vector in each layer, we need to consider $\lambda := \lceil \frac{m}{n} \rceil$ vectors $v^1, \ldots, v^\lambda \in \mathbb{F}^n$ simultaneously. Now we have to solve the system of equations $\sum_{i=0}^{m} v_i^j (wP_i) = 0$ for $j \in \{1, \ldots, \lambda\}$ in order to have enough information for recovering the rows of $\tilde{T}$. As for the case $m \leq n$, this system of linear equations has q^{lr} solutions if and only if all vectors $v^1, \ldots, v^\lambda$ are in the kernel $\ker_l$. Consequently, the complexity for the LowRankAttack increases exponentially with λ and is equal to $O(mn^3 Lq^{\lambda r})$. In practice we will have small values for λ as highly overdetermined systems of quadratic equations are easy to solve [4].

5 Conclusion

In this paper, we have generalised the systems TPM, RSE(2)PKC, and RSSE(2) PKC to the step-wise triangular system (STS). In particular, we allow "steps" which contain more than one new variable (restriction in TPM) and give the private key polynomials p'_i more flexibility than in RSE(2)PKC or RSSE(2)PKC. We have presented two different types of attacks against the STS schemes: an inversion attack with complexity $O(mn^3 Lq^r + n^2 Lrq^r)$ and a structural attack with complexity $O(mn^3 Lq^r + mn^4)$. As the value of q^r has to be chosen rather small to derive a practical scheme, we conclude that STS is broken for all practical values (TPM uses 2 here while RSE(2)PKC and RSSE(2)PKC allow 1024 as maximal value). This is a new result for the special cases RSE(2)PKC and RSSE(2)PKC which have been considered to be secure against rank-attacks by their inventors. In particular, we were able to compute the solutions for the challenges proposed by Kasahara and Sakai (cf Sect. 3.2).
We have demonstrated that the existing generalisations of STS are either insecure or impractical. At present, it does not seem likely that there will ever be secure versions of STS schemes. In particular, we see no way of avoiding both the large kernel at one end and the small kernel at the other end — leave alone the chain of kernels — and still obtaining a scheme which may be used in practice for either encryption or signing.

Acknowledgements

We want to thank Taizo Shirai (Sony, Japan) for his help in connection with the challenge.

This work was supported in part by the Concerted Research Action (GOA) GOA Mefisto 2000/06, GOA Ambiorix 2005/11 of the Flemish Government and the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT. The second author was mainly supported by the FWO.

Disclaimer

The information in this document reflects only the author's views, is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

References

1. Computational Algebra Group, University of Sydney. *The MAGMA Computational Algebra System for Algebra, Number Theory and Geometry.* `http://magma.maths.usyd.edu.au/magma/`.
2. Don Coppersmith, Jacques Stern, and Serge Vaudenay. Attacks on the birational permutation signature schemes. In Cr [7], pages 435–443.
3. Don Coppersmith, Jacques Stern, and Serge Vaudenay. The security of the birational permutation signature schemes. *Jounal of Cryptology*, 10:207–221, 1997.
4. Nicolas Courtois, Louis Goubin, Willi Meier, and Jean-Daniel Tacier. Solving underdefined systems of multivariate quadratic equations. In *Public Key Cryptography — PKC 2002*, volume 2274 of *Lecture Notes in Computer Science*, pages 211–227. David Naccache and Pascal Paillier, editors, Springer, 2002.
5. Nicolas Courtois, Louis Goubin, and Jacques Patarin. *Quartz: Primitive specification (second revised version)*, October 2001. `https://www.cosic.esat.kuleuven.ac.be/nessie/workshop/submissions/quartzv21-b.zip`, 18 pages.
6. Nicolas T. Courtois. The security of Hidden Field Equations (HFE). In *The Cryptographer's Track at RSA Conference 2001*, volume 2020 of *Lecture Notes in Computer Science*, pages 266–281. D. Naccache, editor, Springer, 2001. `http://www.minrank.org/hfesec.{ps|dvi|pdf}`.
7. Douglas R. Stinson, editor. *Advances in Cryptology — CRYPTO 1993*, volume 773 of *Lecture Notes in Computer Science*. Springer, 1993. ISBN 3-540-57766-1.
8. Harriet Fell and Whitfield Diffie. Analysis of public key approach based on polynomial substitution. In *Advances in Cryptology — CRYPTO 1985*, volume 218 of *Lecture Notes in Computer Science*, pages 340–349. Hugh C. Williams, editor, Springer, 1985.
9. Michael R. Garay and David S. Johnson. *Computers and Intractability — A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, 1979. ISBN 0-7167-1044-7 or 0-7167-1045-5.
10. Louis Goubin and Nicolas T. Courtois. Cryptanalysis of the TTM cryptosystem. In *Advances in Cryptology — ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 44–57. Tatsuaki Okamoto, editor, Springer, 2000.
11. Masao Kasahara and Ryuichi Sakai. private communication, 3$^{\text{rd}}$ of April 2004.
12. Masao Kasahara and Ryuichi Sakai. A construction of public-key cryptosystem based on singular simultaneous equations. In *Symposium on Cryptography and Information Security — SCIS 2004*. The Institute of Electronics, Information and Communication Engineers, January 27–30 2004. 6 pages.

13. Masao Kasahara and Ryuichi Sakai. A construction of public key cryptosystem for realizing ciphtertext of size 100 bit and digital signature scheme. *IEICE Trans. Fundamentals*, E87-A(1):102–109, January 2004. Electronic version: `http://search.ieice.org/2004/files/e000a01.htm\#e87-a,1,102`.
14. Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced oil and vinegar signature schemes. In *Advances in Cryptology — EUROCRYPT 1999*, volume 1592 of *Lecture Notes in Computer Science*, pages 206–222. Jacques Stern, editor, Springer, 1999. Extended version: [15].
15. Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced oil and vinegar signature schemes — extended version, 2003. 17 pages, `citeseer/231623.html`, 2003-06-11, based on [14].
16. F.J. MacWilliams and N.J.A. Sloane. *The Theory of Error-Correcting Codes*. Elsevier Science Publisher, 1991. ISBN 0-444-85193-3.
17. Tsutomu Matsumoto and Hideki Imai. Public quadratic polynomial-tuples for efficient signature verification and message-encryption. In *Advances in Cryptology — EUROCRYPT 1988*, volume 330 of *Lecture Notes in Computer Science*, pages 419–545. Christoph G. Günther, editor, Springer, 1988.
18. Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996. ISBN 0-8493-8523-7, online-version: `http://www.cacr.math.uwaterloo.ca/hac/`.
19. Jacques Patarin. Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): two new families of asymmetric algorithms. In *Advances in Cryptology — EUROCRYPT 1996*, volume 1070 of *Lecture Notes in Computer Science*, pages 33–48. Ueli Maurer, editor, Springer, 1996. Extended Version: `http://www.minrank.org/hfe.pdf`.
20. Jacques Patarin and Louis Goubin. Trapdoor one-way permutations and multivariate polynomials. In *International Conference on Information Security and Cryptology 1997*, volume 1334 of *Lecture Notes in Computer Science*, pages 356–368. International Communications and Information Security Association, Springer, 1997. Extended Version: `http://citeseer.nj.nec.com/patarin97trapdoor.html`.
21. Jacques Patarin, Louis Goubin, and Nicolas Courtois. Improved algorithms for Isomorphisms of Polynomials. In *Advances in Cryptology — EUROCRYPT 1998*, volume 1403 of *Lecture Notes in Computer Science*, pages 184–200. Kaisa Nyberg, editor, Springer, 1998. Extended Version: `http://www.minrank.org/ip6long.ps`.
22. Adi Shamir. Efficient signature schemes based on birational permutations. In Cr [7], pages 1–12.
23. Thorsten Theobald. How to break shamir's asymmetric basis. In *Advances in Cryptology — CRYPTO 1995*, volume 963 of *Lecture Notes in Computer Science*, pages 136–147. Don Coppersmith, editor, Springer, 1995.
24. Ilia Toli. Cryptanalysis of HFE, June 2003. arXiv preprint server, `http://arxiv.org/abs/cs.CR/0305034`, 7 pages.