Des Codes Crypt (2006) 40:285–302
DOI 10.1007/s10623-006-0015-5

On the security of stepwise triangular systems

Christopher Wolf · An Braeken · Bart Preneel

Received: 28 April 2005 / Revised: 25 October 2005 /
Accepted: 6 March 2006

Abstract In 2003 and 2004, Kasahara and Sakai suggested the two schemes RSE(2)PKC and RSSE(2)PKC, respectively. Both are examples of public key schemes based on $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic equations. In this article, we first introduce Step-wise Triangular Schemes (STS) as a new class of $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic public key schemes. These schemes have m equations, n variables, L steps or layers, r the number of equations and new variables per step and q the size of the underlying finite field $\mathbb{F}$. Then, we derive two very efficient cryptanalytic attacks. 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. As the legitimate user also has a workload growing with q^r to recover a message/compute a signature, q^r has to be small for efficient schemes and the attacks presented in this article are therefore efficient. After developing our theory, we demonstrate that both RSE(2)PKC and RSSE(2)PKC are special instances of STS and hence, fall to the attacks developed in our article. In particular, we give the solution for the crypto challenge proposed by Kasahara and Sakai. Finally, we demonstrate that STS cannot be the basis for a secure $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic public key scheme by discussing all possible variations and pointing out their vulnerabilities.

Keywords Multivariate cryptography · Rank attacks · Efficient cryptanalysis · Triangular systems

AMS Classification 94A60 · 11T55 · 12F99 · 68W40 · 68Q25 · 51E26 · 14Q99 · 14N10

Communicated by H. Imai.

C. Wolf (✉)· A. Braeken · B. Preneel
Department Electrical Engineering, ESAT/COSIC,
Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Heverlee-Leuven, Belgium
e-mail: christopher.wolf@esat.kuleuven.be; chris@christopher-wolf.de

A. Braeken
e-mail: an.braeken@esat.kuleuven.be

B. Preneel
e-mail: bart.preneel@esat.kuleuven.be

1 Introduction

Public key cryptography is used in e-commerce systems for authentication (electronic signatures) and secure communication (encryption). Both applications proved to be a vital backbone for today's information society: buying a book at Amazon, doing online-banking, or casting a vote secured through TLS, ... An Internet user is constantly using public key techniques to ensure the authenticity of data and also its confidentiality. The security of the public key schemes currently centres on the difficulty of solving certain classes of problems: RSA relies on the difficulty of factoring large integers, while the difficulty of solving discrete logarithms provide the basis for Elliptic Curve Cryptography (ECC) [1]. Hence, all systems used in practice rely on the difficulty of two problems only. We want to point out that these problems are only *believed* to be hard but that no proof is known which establishes, e.g., $\mathcal{NP}$-completeness or even $\mathcal{NP}$-hardness for these two problems. In addition, important results on the potential weaknesses of these public key schemes are emerging as techniques for factorisation and solving discrete logarithm continually improve. For example, polynomial time quantum algorithms [2] are able to solve both problems and therefore, the existence of quantum computers in the range of 1,000 bits would be a real-world threat to systems based on factoring or the discrete log problem; the effects on our Internet-driven economy would be disastrous. This points to the importance of research into new algorithms for asymmetric cryptography. We want to stress at this point that there are not many results known about the vulnerability of cryptographic hard problems against quantum algorithms. We are only aware of shor [2] at this point. Hence, more research effort in this direction seems to be imperative if we assume the existence of quantum computers within the next decades.

1.1 PKC schemes based on multivariate quadratic $\mathcal{MQ}$ equations

One important alternative to schemes based on factoring or elliptic curves are public key protocols based on the intractability of the problem of solving a simultaneous system of $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic polynomial equations over a finite field $\mathbb{F}$. This is called the $\mathcal{MQ}$-problem. In the last 15 years, several such public key cryptoschemes (PKC) have been proposed, see [3] for an up-to-date overview. Generally speaking, a multivariate PKC public key $\mathcal{P}$ has the structure $S \circ \mathcal{P}' \circ T$ where $\circ$ denotes the composition of functions. 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 central map $\mathcal{P}'$ consists of m central equations in n variables each. For an $\mathcal{MQ}$-scheme, the degree of these equations is 2. Moreover, the central map $\mathcal{P}'$ must be easy to invert to allow the decryption or signing of messages. So the secret key of the $\mathcal{MQ}$-system is composed of the triple $(S, \mathcal{P}', T)$. We want to point out that the different proposals only differ in the structure of their central equations $\mathcal{P}'$. Hence, depending on this structure, we are able to identify several classes: e.g., the initial polynomial substitution scheme from Fell and Diffie [4], C* schemes [5], HFE-like schemes [6, 7] or Unbalanced Oil Vinegar schemes [8]. 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 10, Appendix] for a detailed proof). In addition, factoring $\mathcal{P}$ into its components $T, \mathcal{P}', S$ is considered to be a hard problem if $S, \mathcal{P}', T$ do not have a special structure. This problem has previously been studied under the name Isomorphism of Polynomials Problem [11–13].

In this article, we turn to a sub-class of $\mathcal{MQ}$-schemes, i.e., to schemes which use a triangular structure for their central equations $\mathcal{P}'$. We call them triangular schemes for short. This idea can be found in [14] and was used to develop birational permutation schemes over large finite rings. To guard these schemes against special types of attacks, the author of [14] removed some equations of the public key. The approach from [14] has been specialised by Goubin et al. to the case of small finite fields. This construction is denoted Triangle Plus Minus (TPM [15]). Apart from changing the focus from large rings to small fields, they add to the construction of [14] some equations in the last step ("Plus" modification). Due to the specialisation to fields, TPM falls in the same class as the scheme described in this article (cf. Fig. 2). Actually, step-wise triangular schemes (STS) can be viewed as a generalisation of TPM (cf. Section 2.3 for further details).

In our construction STS, we allow steps of more than only one variable but keep the triangular structure of TPM (cf. Fig. 1 for regular STS). As shown in this figure, the step-width, i.e., the number of new variables and the step-height, i.e., the number of new equations, is controlled by the parameter r. For comparison: in Birational Permutations and TPM, the parameter r is fixed to 1.

1.2 Organisation and outline

In the main part of this article, we will describe two very efficient attacks on STS schemes. They defeat STS in $O(mn^3Lq^r + mn^4)$ and $O(mn^3Lq^r + n^2Lrq^r)$ operations—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-height. All constructions of STS known so far need a workload proportional to q^r for the legitimate user to decrypt messages/compute signatures. Therefore, the number q^r has to be rather small to allow efficient and therefore practical constructions. Consequently, the attacks described in this article are efficient in practice. The main observation for our attacks is the fact that the kernels of the private central polynomials p'_i form a descending chain of subspaces (cf. Section 3.1). As we can demonstrate that the recently proposed schemes RSE(2)PKC and RSSE(2)PKC by Kasahara and Sakai belong to the STS family (cf. Section 4), it follows that they are covered by these attacks. We therefore conclude that they are highly insecure. As an application of the attacks described in this article, we computed the solution for the RSE(2)PKC challenge (cf. Section 4.2). This challenge was proposed in [16].

The remainder of this article is organised as follows: after this introduction, we describe Step-wise Triangular Systems in Section 2. Then, we move on to a cryptanalysis of regular STS schemes, showing both an inversion and a structural attack in Section 3. In Section 4 deals with special instances like RSE(2)PKC and RSSE(2)PKC. In Section 5, we study some generalisations of STS schemes. This article concludes with Section 6.

$$
\begin{array}{ll}
\text{Step } 1 & \left\{ \begin{array}{ll} p'_1 & (x'_1,\ldots,x'_r) \\ & \quad\vdots \\ p'_r & (x'_1,\ldots,x'_r) \end{array}\right. \\
\vdots & \\
\text{Step } l & \left\{ \begin{array}{ll} p'_{(l-1)r+1} & (x'_1,\ldots,x'_r,\quad \ldots,\quad x'_{(l-1)r+1},\ldots,x'_{lr}) \\ & \qquad\qquad\qquad \vdots \\ p'_{lr} & (x'_1,\ldots,x'_r,\quad \ldots,\quad x'_{(l-1)r+1},\ldots,x'_{lr}) \end{array}\right. \\
\vdots & \\
\text{Step } L & \left\{ \begin{array}{ll} p'_{(L-1)r+1} & (x'_1,\ldots,x'_r,\quad \ldots,\quad x'_{(l-1)r+1},\ldots,x'_{lr},\quad \ldots,\quad x'_{n-r+1},\ldots,x'_n) \\ & \qquad\qquad\qquad \vdots \qquad\qquad\qquad\qquad \vdots \\ p'_{Lr} & (x'_1,\ldots,x'_r,\quad \ldots,\quad x'_{(l-1)r+1},\ldots,x'_{lr},\quad \ldots,\quad x'_{n-r+1},\ldots,x'_n) \end{array}\right.
\end{array}
$$

Fig. 1 Central equations p'_i in a regular STS scheme

2 Step-wise triangular systems

In this section, we will define public key schemes which use $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic equations. We start with the definition and properties of their public keys.

2.1 Encryption and public key size

As explained earlier, all schemes based on the $\mathcal{MQ}$-problem have the same structure for the public key. This observation leads to the following

Definition 2.1 Let $\mathbb{F}$ be a finite field and $n, m \in \mathbb{N}$ the number of variables and equations, respectively. Moreover, let for $1 \le i \le m$ and $1 \le j \le k \le n$: $\alpha_i, \beta_{i,j}, \gamma_{i,j,k} \in \mathbb{F}$. We call them constant, linear and quadratic coefficients, respectively. Then we have the public key polynomials

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

Based on these polynomials, we define the polynomial vector $\mathcal{P} := (p_1, \dots, p_m)$ and denote the class of all such polynomial vectors by $\mathcal{MQ}_m(\mathbb{F}^n)$. Note that the m entries $p_1, \dots, p_m$ of $\mathcal{P}$ are multivariate quadratic polynomials in n input variables each. We call $\mathcal{P}$ the "public key" of an $\mathcal{MQ}$-system.

Lemma 2.2 *Let $\mathcal{P}$ be a public key as in Definition* 2.1, *n the number of variables, m the number of equations*, $\mathbb{F}$ *the ground field and $q := |\mathbb{F}|$ the number of its elements. Then the following formula can be used to compute the number of coefficients for an individual polynomial $p_i : 1 \le i \le m$:*

$$\tau(n) := \begin{cases} 1 + n + \frac{n(n-1)}{2} = 1 + \frac{n(n+1)}{2}, & \textit{if } \mathbb{F} = GF(2), \\ 1 + n + \frac{n(n+1)}{2} = 1 + \frac{n(n+3)}{2}, & \textit{otherwise}. \end{cases}$$

Proof The first row in the above expression comes from the fact that we have $x_i^2 = x_i$ for $\mathbb{F}$ = GF(2) and $1 \le i \le n$, i.e., quadratic terms of the form x_i^2 over GF(2) reduce to linear terms. The rest of the formula follows from a simple combinatoric count. □

Corollary 2.3 *Using the formula from Lemma* 2.2, *we obtain $m\tau(n) = O(mn^2)$ for the number of coefficients and hence a memory requirement of* $\log_{256}(q)m\tau(n)$ *byte for the public key.*

Remark For a secure $\mathcal{MQ}$-system, the public key polynomials should behave similar to random equations. Therefore, we do not expect to find efficient compression techniques for these keys. The size requirement from Corollary 2.3 are therefore tight.

2.2 Private key and decryption

After defining the public key of general $\mathcal{MQ}$-systems, we now move on to the special class of general Stepwise Triangular Systems:

Definition 2.4 Let $\mathbb{F}$ be a finite field, n, m two integers and $S \in \mathrm{GL}_n(\mathbb{F})$, $T \in \mathrm{GL}_m(\mathbb{F})$ two linear transformations. Moreover, let $r_1, \dots, r_L$ be L integers such that $r_1 + \cdots + r_L = n$, the number of variables, and $m_1, \dots, m_L \in \mathbb{N}$ 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

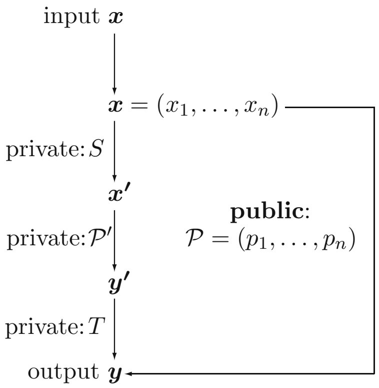

Fig. 2 $\mathcal{MQ}$-trapdoor $(S, \mathcal{P}', T)$ in STS

the number of variables (step-width) and m_l the number of equations (step-height), both in step l for $1 \leq l \leq L$. Now define $\mathcal{P}' \in \mathcal{MQ}_m(\mathbb{F}^n)$ as the system of m multivariate quadratic equations in n variables where 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. Then we call the triple $(S, \mathcal{P}', T) \in \mathrm{GL}_n(\mathbb{F}) \times \mathcal{MQ}_m(\mathbb{F}^n) \times \mathrm{GL}_m(\mathbb{F})$ the private key of a general STS system (gSTS).

The overall shape of the private polynomials leads to the name STS. We define its public key as $\mathcal{P} := T \circ \mathcal{P}' \circ S$ where $\circ$ denotes the composition of functions, cf. Fig. 2 for the overall structure of STS schemes.

The structure of an STS can be seen in Fig. 2, and the form of its central map is outlined in Fig. 1. In addition, we want to point out that the two linear transformations S and T can be expressed as invertible matrices, i.e., we can write $S \in \mathbb{F}^{n \times n}$ and $T \in \mathbb{F}^{m \times m}$, respectively.

Remark In this article, we always use a prime ($'$) for denoting the secret central part of the system, e.g., the variables $x'_1, \ldots, x'_n$, the output $y'_1, \ldots, y'_m$ of the polynomials $p'_1, \ldots, p'_m$ over $\mathbb{F}$, and hence their coefficients $\alpha'_i, \beta'_{i,j}, \gamma'_{i,j,k} \in \mathbb{F}$ for $1 \leq i \leq m$ and $1 \leq j \leq k \leq n$. By convention, we have $j < k$ in the case of $\mathbb{F} = \mathrm{GF}(2)$ (cf. Lemma 2.2).

To simplify the explanations, we concentrate on regular STS schemes (rSTS or STS for short) in this article. For regular STS schemes we set $r_1 = \cdots = r_N = m_1 = \cdots = m_L$, which we denote by r. Moreover, $m = Lr$ and $m = n$. Note that the attacks we propose are also valid for the gSTS schemes (cf. Sect. 5.1). The structure of a STS has been outlined in Figs. 1 and 2.

Lemma 2.5 *Let $(S, \mathcal{P}', T) \in GL_n(\mathbb{F}) \times \mathcal{MQ}_m(\mathbb{F}^n) \times GL_m(\mathbb{F})$ be the private key of a rSTS scheme and $\mathcal{P}$ the corresponding public key. If there is no additional trapdoor embedded into the different layers $1 \leq l \leq L$, the legitimate user has a workload proportional to Lq^r to invert the equation $\mathcal{P}(x) = y$ for $y \in \mathbb{F}^m$ and given private key $(S, \mathcal{P}', T)$.*

Proof In order to decrypt a given ciphertext $\boldsymbol{y}$, we need to invert $\boldsymbol{x} \xrightarrow{S} \boldsymbol{x}' \xrightarrow{\mathcal{P}'} \boldsymbol{y}' \xrightarrow{T} \boldsymbol{y}$. Both S, T are bijections and the legitimate user needs $O(n^3)$ steps for inversion (or $O(n^2)$ if S^{-1} and T^{-1} have been precomputed). However, the central equations do not have any trapdoor embedded in their equations. Hence, the legitimate user can only use brute force, i.e., try q^r different possibilities for each of the L layers. This establishes the lemma. □

General STS schemes are no bijections. Hence, in order to recover the correct message for a given ciphertext, we need to either add 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 [1]). This 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 [7, 17] for a broader discussion of this problem.

2.3 Comparison with other schemes

As already pointed out in the introduction, the Birational Permutation Schemes of Shamir are 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 [18–20]. The TPM scheme of Goubin and Courtois has been broken in the very article that proposed it [15]. 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. [16, Sect. 4.3.III] and [17, Sect. 4.1.III) that their constructions are secure against all known attacks—in particular, mentioning [15]. Although this observation is true, we will show in Sect. 3 that it is possible to generalise these attacks in a way that STS and consequently RSE(2)PKC and RSSE(2)PKC are broken.

3 Cryptanalysis

We now present two different types of attacks on STS. In the inversion attack (cf. Section 3.3), we recover for given ciphertext $\boldsymbol{y}$ the corresponding message $\boldsymbol{x}$. In the structural attack (cf. Section 3.4), 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 for forging a signature on it. For both attacks, we first need some observations on kernels.

3.1 Chain of kernels

Let p_i be a public key polynomial as defined in (1). In order to uniquely express its homogeneous quadratic parts in a symmetric matrix $P_i \in \mathbb{F}^{n\times n}$, we need to distinguish odd and even characteristic.

- For characteristic $\neq 2$, the matrix elements $(P_i)_{a,b}$ on row a and column b of the symmetric matrix P_i are determined by

$$\begin{cases} (P_i)_{a,b} = \frac{\gamma_{a,b}}{2} & \text{for } 1 \leq a < b \leq n, \\ (P_i)_{a,a} = \gamma_{a,a} & \text{for } 1 \leq a \leq n\,. \end{cases}$$

$$\begin{pmatrix}
\gamma'_{1,1} & \gamma'_{1,2}/2 & \cdots & \cdots & \gamma'_{1,rl}/2 & 0 & \cdots & 0 \\
\gamma'_{2,1}/2 & \gamma'_{2,2} & & & \gamma'_{2,rl}/2 & 0 & & 0 \\
\vdots & \vdots & \ddots & & \vdots & \vdots & & \vdots \\
\gamma'_{rl-1,1}/2 & \gamma'_{rl-1,2}/2 & & \gamma'_{rl-1,rl-1} & \gamma'_{rl-1,rl}/2 & 0 & \cdots & 0 \\
\gamma'_{rl,1}/2 & \gamma'_{rl,2}/2 & \cdots & \gamma'_{rl,rl-1}/2 & \gamma'_{rl,rl} & 0 & & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0
\end{pmatrix}$$

Fig. 3 Matrix representation of the private key p'_i for layer l

So, instead of evaluating the quadratic parts of p_i by the vector $\boldsymbol{x}$, we may also perform $\boldsymbol{x} P_i \boldsymbol{x}^T$ as matrix-vector multiplications (here ^{T} denotes transposition), cf. Fig. 3 for a graphical representation of this idea.

- For even characteristic, division by 2 is not defined. Therefore, the form $P_i := L_i + L_i^T$ for lower triangular matrices L_i is used. 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 [8] and is there credited to Coppersmith (private communication).

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

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

as common kernels of the matrices P'_i for $(l-1)r < i \leq lr$, cf. Fig. 3 for the corresponding matrix. 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 $\boldsymbol{a'} \in \mathbb{F}^n$.

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 $\hat{P}_i := S P'_i S^T$ in terms of the corresponding matrices. As the transformation S is invertible, we have Rank($\hat{P}_i$) = Rank(P'_i) and

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

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

$$\ker'_L \subset \cdots \subset \ker'_1 \quad \text{and consequently } \ker_L \subset \cdots \subset \ker_1 \,.$$

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

$$P_i = \sum_{j=1}^{m} \tau_{i,j} [S P'_i S^T] = \sum_{j=1}^{m} \tau_{i,j} \hat{P}_i \,.$$

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.

3.2 Recovering the transformation T

As we saw in the previous section, it is crucial for an attack of STS schemes to recover the transformation T. In this section, we describe two algorithms which can be used for this purpose.

3.2.1 Attacking the high-rank side

We start with an attack on the high-rank side (cf. the algorithm in Fig. 4). 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 following vector spaces.

Definition 3.1 Define an ascending chain of subspaces J_l of dimension $(m - lr)$ for $1 \leq l \leq L$ as

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

When picking a random element $\boldsymbol{v} \in_R J_{l+1}$, we have a probability of q^{-r} that the expression $\boldsymbol{v} \in J_l$ holds because of the definition of the subspaces J_l, J_{l+1}. In addition, we have two efficient methods (`matrixCheck` or `polynomialCheck`, respectively) to check whether $\boldsymbol{v} \in J_l$ or $\boldsymbol{v} \notin J_l$. First, we concentrate on `matrixCheck`.

Theorem 3.2 *The method* `matrixCheck` *will check if* $\boldsymbol{v} \in J_l$ *and is defined by*

$$\texttt{matrixCheck(}P_1, \ldots, P_m,\ v, l\texttt{)}\ \textit{returns}\ \textbf{\textit{true}}\quad \textit{iff Rank}\left(\sum_{i=1}^{m} v_i P_i\right) \leq lr.$$

```
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 $l \leftarrow L \leftarrow 1$ downto 1 do
    $J_l \leftarrow \emptyset$
    repeat
      $\boldsymbol{v} \in_R J_{l+1}$
      if matrixCheck($P_1, \ldots, P_m,\ \boldsymbol{v}, l$) $\vee$ polynomialCheck($p_1, \ldots, p_m,\ \boldsymbol{v}, l$) then
        $J_l \cup \leftarrow \{\boldsymbol{v}\}$
    until Dimension($J_l$) $\stackrel{?}{=} lr$
    $\tilde{J} \leftarrow J_{l+1} \cap J_l$
    for $i \leftarrow 1$ to $r$ do
      RowVector($\hat{T}, lr + i$) $\leftarrow$ BasisVector($\tilde{J}, i$)
  endfor
  return $\tilde{T} \leftarrow \hat{T}^{\leftarrow 1}$
endproc
```

Fig. 4 High-rank algorithm for computing the transformation $\tilde{T}$ for a given system of equations

Proof 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 $\boldsymbol{b'}$ instead of $\boldsymbol{v}$. Hence, we have

$$M := \sum_{i=1}^{m} b'_i \hat{P}_i = \sum_{i=1}^{rl} b'_i \left(S P'_i S^T \right) = S \left(\sum_{i=1}^{rl} b'_i P'_i \right) S^T \ .$$

Observing the step-wise structure of the private key polynomials p'_i we conclude that the Rank$(M) \leq lr$. This yields the result. □

The expected running time of the algorithm from Fig. 4 is therefore bounded by $O(mn^3 Lq^r)$: 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.

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 `polynomial-Check`). Unfortunately, the proof is a bit lengthy, we therefore refer to [21, Sect. 15.2, Theorem 4] for both the theorem and its proof. An algorithmic version of it can be found in [22, Sect. 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: $O(mn^3 Lq^r)$.

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.

3.2.2 Attacking the low-rank side

In the previous algorithm, we constructed the linear equivalent copy of T^{-1} stepwise by means of the r basis vectors from the subspace $\tilde{J} = J_{l+1} \cap J_l$. Therefore, we call it an attack from the low-rank side. We now show how we can also perform an attack from the high-rank side. For this purpose, we define $\overline{J_l} := \{\boldsymbol{b'} T^{-1} \mid \boldsymbol{b'} \in \mathbb{F}^m \text{ and } b'_1 = \cdots = b'_{lr} = 0\}$ as *complement* of the vector space J_l, $0 \leq 1 \leq L$.

Theorem 3.3 *The subspace* $\tilde{J} := J_l \cap \overline{J_{l-1}}$ *where* $\overline{J_{l-1}}$ *denotes the complement of the vector space* J_{l-1}, *has dimension r and will determine r new linearly independent rows of the matrix* T^{-1}.

Proof The proof is based on two different observations. The first one is that the kernels $\ker_i$ form a descending chain. Therefore, setting $\ker_0 := \mathbb{F}^n$, the statement $\boldsymbol{w} \in \ker_l$ is true with probability q^{-r} for all $\boldsymbol{w} \in_R \ker_{l-1}$ and $1 \leq l \leq L$. Second, the linear equation $\sum_{i=1}^{m} v_i (w P_i) = 0$ has q^{lr} solutions for unknown $\boldsymbol{v} \in \mathbb{F}^m$ if and only if the vector $\boldsymbol{w}$ is in the kernel $\ker_l$. □

The algorithm will therefore terminate with a correct solution $\tilde{T}$ after a total of $O(Ln^3 q^r)$ 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 Figs. 4 and 5 can be found in [15] for the case of schemes with step-width 1 of the intermediate layers.

```
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
      $\boldsymbol{w} \in_R K_{l \leftarrow 1}$
      $J_l \leftarrow$ SolutionSpace($\sum_{i=1}^m v_i(\boldsymbol{w}P_i) = 0$) for an unknown $\boldsymbol{v} \in \mathbb{F}^m$
    until Dimension($J_l$) $\overset{?}{=} lr$.
    $\tilde{J} \leftarrow J_l \cap \overline{J_{l \leftarrow 1}}$
    for $i \leftarrow 1$ to $r$ do
      $\hat{t} \leftarrow$ BasisVector($\tilde{J}, i$); RowVector($\hat{T}, lr + i$) $\leftarrow \hat{t}$; $\hat{P}_{(l \leftarrow 1)r+i} \leftarrow \sum_{j=1}^m \hat{t}_j P_j$
    $K_l \leftarrow$ Kernel($P_{lr}$)
  endfor
  return $\tilde{T} \leftarrow \hat{T}^{\leftarrow 1}$
endproc
```

Fig. 5 Low-rank algorithm for computing the transformation $\tilde{T}$ for a given system of equations

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

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

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

Fig. 6 Inversion attack for $\boldsymbol{y} = \mathcal{P}(\boldsymbol{x})$ and given $\tilde{T}$

3.3 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 $\boldsymbol{y} = \mathcal{P}(\boldsymbol{x})$ for a given vector $\boldsymbol{y} \in \mathbb{F}^m$, i.e., for the $\mathcal{MQ}$-problem. We do so by computing a successive affine approximation of $\boldsymbol{x}$ (cf. Fig. 6). Define $K_i := \ker_i$ for $1 \leq i \leq L$. In addition, we write $\overline{K_l} := \{\boldsymbol{a'}S^{-1} \mid \boldsymbol{a'} \in \mathbb{F}^n \text{ and } a'_{rl+1} = \cdots = a'_n = 0\}$ for its *complement*.

Theorem 3.4 *The solutions form a chain of affine subspaces $\boldsymbol{x} + \langle K_l \rangle$—where K_l has dimension $n - rl$ in step l.*

Proof Recall that the kernels $K_i := \ker_i$ for $1 \leq i \leq L$ have the form $\ker_l = \{\boldsymbol{a'}S^{-1} \mid \boldsymbol{a'} \in \mathbb{F}^n \wedge a'_1 = \cdots = a'_{rl} = 0\}$. Setting $K_0 := \mathbb{F}^n$ we have

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

for $1 \leq l \leq L$. Using this observation, we can "switch on" groups of r (hidden) variables $\boldsymbol{x'}$ 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. The statement in the theorem then follows from the fact that the polynomial system $\hat{\mathcal{P}}$ inherits the layer structure of the original private polynomial system $\mathcal{P}'$. □

Therefore, we can conclude that we learn $r \log_2 q$ bits about the vector $\boldsymbol{x}$ for each level of recursion.

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 $O(n^2Lrq^r)$.

3.4 Structural attack

The starting point of the structural attack (cf. Fig. 7) is the same as for the inversion attack, namely $\ker_1 \supset \cdots \supset \ker_L$. 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 $\leftarrow$1 to $L$ do
    $\tilde{K} \leftarrow K_{l\leftarrow 1} \cap \overline{K_l}$
    RowVector($\hat{S}$, $(l \leftarrow 1)r + i$) $\leftarrow$BasisVector($\tilde{K}, i$) : $1 \leftarrow i \leftarrow r$
  $\tilde{S} \leftarrow \hat{S}^{\leftarrow 1}$
  $\tilde{p}'_i \leftarrow \hat{p}_i \leftarrow \tilde{S}^{\leftarrow 1}$ : $1 \leftarrow i \leftarrow m$
  return $\tilde{S}, \tilde{P}'$
endproc
```

Fig. 7 Structural attack for a given sequence of kernels $\ker_1, \ldots, \ker_L$

4 Special instances of STS

In this section, we show that the two schemes RSE(2)PKC [16] and RSSE(2)PKC [17], 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 [16, Sect. 6] using an inversion attack (cf. Section 3.3) in both cases.

4.1 RSSE(2)PKC

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

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

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 attacks. 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 $i \ne j$ 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. 5 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. 4 in connection with Dickson's theorem (cf. Section 2.2).

4.2 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 [16, Section 3.2], the authors suggest $r \le 10$ 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.

4.2.1 Challenges

In [16, Sect. 6], Kasahara and Sakai propose two challenges with the following parameters: $\mathbb{F}$ = GF(2), $n = 100$ and $r = 4, 5$. Using a (highly unoptimised) Magma [23] 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. Sects. 3.2.2 and

3.3). 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. Sections 3.2.1). 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 (Private communication).

Apart from the attacks presented in this article, 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 [24], the generic birthday attack applies against RSE(2)PKC, RSSE(2)PKC, and also the hybrid type construction from the following section.

4.3 Hybrid type construction

In [17, 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 article.

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:

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

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 article are still applicable.

5 Extensions of STS and their vulnerabilities

5.1 General step-wise triangular systems

As outlined in Section 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 $\boldsymbol{y} = \mathcal{P}(\boldsymbol{x})$ 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 TPM construction. In particular, we relax the condition that there is only one new variable and one new equation at each intermediate level (cf. Section 2).

5.2 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 $\boldsymbol{v_s} \in \mathbb{F}^n$ and $\boldsymbol{v_t} \in \mathbb{F}^m$.

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 $M_S \in \mathbb{F}^{n\times n}$ (respectively $M_T \in \mathbb{F}^{m\times m}$) and a unique vector $\boldsymbol{v_s} \in \mathbb{F}^n$ (respectively, $\boldsymbol{v_t} \in \mathbb{F}^m$) which describes the affine transformation S (respectively, T) by $S(\boldsymbol{x}) = M_S\boldsymbol{x} + \boldsymbol{v_s}$ where $\boldsymbol{x} \in \mathbb{F}^n$ is an input vector (respectively, $T(\boldsymbol{x}) = M_T\boldsymbol{x} + \boldsymbol{v_t}$ for $\boldsymbol{x} \in \mathbb{F}^m$). Moreover, we can rewrite the affine transformation S as $S(\boldsymbol{x}) = (\overline{\boldsymbol{x}} + \boldsymbol{v_s}) \circ (M_S\boldsymbol{x})$ where $\overline{\boldsymbol{x}}$ denotes the output of $M_S\boldsymbol{x}$. In addition, we can rewrite the affine transformation T as $T(\boldsymbol{x}) = (M_T\hat{\boldsymbol{x}}) \circ (\boldsymbol{x} + M_T^{-1}\boldsymbol{v_t})$, where $\hat{\boldsymbol{x}}$ denotes the output of $\boldsymbol{x} + M_T^{-1}\boldsymbol{v_t}$. As M_T is an invertible matrix, the matrix $M_T^{-1} \in \mathbb{F}^{m\times m}$ exists and is unique.

Theorem 5.1 *Consider the PKC with public key $\mathcal{P} = (S, \mathcal{P}', T) \in AGL_n(\mathbb{F}) \times \mathcal{MQ}_m(\mathbb{F}^n) \times AGL_m(\mathbb{F})$, where $\mathcal{P}'$ satisfies the gSTS structure. Then the scheme is equivalent to a PKC with an equivalent public key but where S and T are linear transformations.*

Proof We first express the public key as a composition of the private key

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

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

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

for some system of equations $\mathcal{P}''$. As both $(\overline{\boldsymbol{x}} + \boldsymbol{v_s})$ and $(\tilde{\boldsymbol{x}} + M_T^{-1}\boldsymbol{v_t})$ 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 [25]. A comprehensive study of equivalent keys for several multivariate quadratic systems is given in [26, 27].

5.3 Degree larger than 2

In [16, 17], 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. Sections 4.2 and 4.1). 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.

An obvious way of modifying the cipher is to make sure that there are no terms of degree 2 in the public key—and consequently in the private key. In particular, such a system avoids terms of the form $x_i x_i x_j$ for the ground field $\mathbb{F} = \mathrm{GF}(2)$. In this case, we cannot associate matrices to the polynomials (see Section 3.1). However, we now show how to use an equivalent notion of rank.

Theorem 5.2 *Denote by $\boldsymbol{\eta_i} \in \mathbb{F}^n$, the vector with 1 on position i and zeros elsewhere. The rank of the matrix P which corresponds to the quadratic polynomial p coincides with the dimension of the subspace $\Delta(p)$:*

$$\Delta(p) := \{p(\boldsymbol{x} + \boldsymbol{\eta_i}) - p(\boldsymbol{x}) \mid i \in \{1, \ldots, n\}\}.$$

Proof By definition $p(\boldsymbol{x} + \boldsymbol{\eta_i}) - p(\boldsymbol{x})$ represents the linear polynomial consisting of the variables x_j for which $\gamma_{i,j} \neq 0$. Therefore, computing the rank of the matrix P by Gaussian elimination corresponds with the computation of the dimension of the subspace $\Delta(p)$. □

Consequently, the rank of a polynomial matrix P is equal to the number of terms on which the polynomial nonlinearly depends, i.e., the minimum number of variables involved in the quadratic part of the polynomial after applying any affine transformation. The definition of $\Delta(p)$ introduced in Theorem 5.2 does not depend on the degree of the polynomial and can therefore be used in order to replace the rank for polynomials of degree higher than 2.

As a consequence, the overall attack-complexity in both cases does not grow, while the complexity of checking the signature of a given message increases as the public key consists of far more terms now: the number of terms grows in $O(mn^d)$ for $d > 2$.

Another way of breaking such a cipher uses d-tensors, cf. [1] instead of matrices. The overall attack-complexity will grow in this case—but is still polynomial. Moreover, the complexity of checking the signature of a given message will also increase as the public key consists of far more terms now. Hence, from a cryptanalytic point of view, such a cipher should be avoided.

5.4 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 $\boldsymbol{v^1}, \ldots, \boldsymbol{v^\lambda} \in \mathbb{F}^n$ 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 $\boldsymbol{v^1}, \ldots, \boldsymbol{v^\lambda}$ 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, using general purpose algorithms as given in, e.g., [22].

5.5 Tame-like schemes

Moh proposed in 1999 a new type of triangular schemes which were based on the "Tame-transformation" method [28]. Since, the attack from Goubin et al. [15], the design of such schemes has drastically been improved. The latest variation on Tame-signature schemes, called the "Enhanced TTS" due to Yang and Cheng [29, Section 4.2], uses the following construction for the central equations $\mathcal{P}'$

$$
\begin{aligned}
p'_i &:= x_i + \sum_{j=1}^{7} \gamma'_{i,j} x'_j x'_{8+(i+j \bmod 9)} \quad \text{for } i = 8, \ldots, 16,\\
p'_{17} &:= x_{17} + \gamma'_{17,1} x'_1 x'_6 + \gamma'_{17,2} x_2 x_5 + \gamma'_{17,3} x'_3 x'_4 + \gamma'_{17,4} x'_9 x'_{16}\\
&\quad + \gamma'_{17,5} x'_{10} x'_{15} + \gamma'_{17,6} x_{11} x_{14} + \gamma'_{17,7} x'_{12} x'_{13},\\
p'_{18} &:= x_{18} + \gamma'_{18,1} x'_2 x'_7 + \gamma'_{18,2} x'_3 x'_6 + \gamma'_{18,3} x'_4 x'_5 + \gamma'_{18,4} x'_{10} x'_{17}\\
&\quad + \gamma'_{18,5} x'_{11} x'_{16} + \gamma'_{18,6} x'_{12} x'_{15} + \gamma'_{18,7} x'_{13} x'_{14},\\
p'_i &:= x_i + \gamma'_{i,0} x_{i-11} x'_{i-9} + \sum_{j=19}^{i} \gamma'_{i,j-18} x'_{2(i-j)} x'_j\\
&\quad + \sum_{j=i+1}^{27} \gamma'_{i,j-18} x'_{i-j+19} x'_j \quad \text{for } i = 19, \ldots, 27
\end{aligned}
$$

for $\gamma'_{i,j} \in \mathrm{GF}(256)$. In lower-triangular representation, the polynomials $p_8, \ldots, p_{16}$ have a rank of 7 each. The Tame-equations p_{17}, p_{18} yield 7, too, and the equations $p_{19}, \ldots, p_{27}$ give 10. This scheme is a type of triangular scheme since the decryption of the private system of polynomials can be done by easy serial computation through substitution of linear equations. However, it is easy to see that no chain of kernels in the subsequent private polynomials exists, which makes our proposed attacks unfeasible. Hence, the security of these schemes is an open problem. Several constructions for "Scaled-Up" Enhanced TTS schemes were also proposed in [29, Appendix A]. We want to point out in this context that an earlier version of this construction has been broken in [30], exploiting the existence of UOV-type properties.

6 Conclusion

In this article, we have generalised the systems TPM, RSE(2)PKC, and RSSE(2)PKC to the 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 $O(mn^3Lq^r + n^2Lrq^r)$ and a structural attack with complexity $O(mn^3Lq^r + mn^4)$. 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 two here while RSE(2)PKC and RSSE(2)PKC allow 1,024 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. Section 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.

Acknowledgments We want to thank Taizo Shirai (Sony, Japan) for his help in connection with the challenge and Ludovic Perret (ENSTA, France) for pointing out the use of tensors for the case $d > 2$ (cf. Section 5.3). 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.

References

1. Menezes AJ, van Oorschot PC, Vanstone SA (1996) Handbook of Applied Cryptography. CRC Press, Boca Raton, ISBN 0-8493-8523-7, online-version: `http://www.cacr.math.uwaterloo.ca/hac/`
2. Shor PW (1997) Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J Comput 26(5):1484–1509
3. Wolf C, Preneel B (2005) Taxonomy of public key schemes based on the problem of multivariate quadratic equations. Cryptology ePrint Archive, Report 2005/077, `http://eprint.iacr.org/2005/077/`, p 64
4. Fell H, Diffie W (1985) Analysis of public key approach based on polynomial substitution. In: Williams HC (ed) Advances in cryptology—CRYPTO 1985, volume 218 of Lecture Notes in Computer Science, Springer, Berlin, pp 340–349
5. Matsumoto T, Imai H (1988) Public quadratic polynomial-tuples for efficient signature verification and message-encryption. In: Günther CG (ed) Advances in cryptology—EUROCRYPT 1988, volume 330 of Lecture Notes in Computer Science, Springer, Berlin, pp 419–545
6. Courtois NT (2001) The security of Hidden Field Equations (HFE). In: Naccache D (ed) The cryptographer's track at RSA conference 2001, volume 2020 of Lecture Notes in Computer Science, Springer, Berlin pp 266–281, `http://www.minrank.org/hfesec.{ps|dvi|pdf}`
7. Patarin J (1996) Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): two new families of asymmetric algorithms. In: Maurer U (ed) Advances in cryptology—EUROCRYPT 1996, volume 1070 of Lecture Notes in Computer Science, Springer, Berlin, pp 33–48, Extended Version: `http://www.minrank.org/hfe.pdf`
8. Kipnis A, Patarin J, Goubin L (1999) Unbalanced Oil and Vinegar signature schemes. In: Stern J (ed) Advances in cryptology—EUROCRYPT 1999, volume 1592 of Lecture Notes in Computer Science, Springer, Berlin, pp 206–222
9. Garey MR, Johnson DS (1979) Computers and intractability — A guide to the theory of NP-completeness. W.H. Freeman and Company, New York, ISBN 0-7167-1044-7 or 0-7167-1045-5
10. Patarin J, Goubin L (1997) Trapdoor one-way permutations and multivariate polynomials. In: International conference on information security and cryptology 1997, volume 1334 of Lecture Notes in Computer Science, International Communications and Information Security Association, Springer, Berlin, pp 356–368, Extended Version: `http://citeseer.nj.nec.com/patarin97trapdoor.html`
11. Geiselmann W, Meier W, Steinwandt R (2002) An attack on the Isomorphisms of Polynomials problem with one secret. Cryptology ePrint Archive, Report 2002/143, `http://eprint.iacr.org/2002/143`, version from 2002-09-20, p 12 (2000)
12. Levy-dit-Vehel F, Perret L (2003) Polynomial equivalence problems and applications to multivariate cryptosystems. In: Johanson T, Maitra s (eds) Progress in cryptology—INDOCRYPT 2003, volume 2904 of Lecture Notes in Computer Science, Springer, Berlin, pp 235–251
13. Patarin J, Goubin L, Courtois N (1998) Improved algorithms for Isomorphisms of Polynomials. In: Nyberg K (ed) Advances in cryptology—EUROCRYPT 1998, volume 1403 of Lecture Notes in Computer Science, Springer, Berlin, pp 184–200, Extended Version: `http://www.minrank.org/ip6long.ps`
14. Shamir A Efficient signature schemes based on birational permutations. In Cr [11], pp 1–12
15. Goubin L, Courtois NT (2000) Cryptanalysis of the TTM cryptosystem. In: Okamoto T (ed) Advances in cryptology—ASIACRYPT 2000, volume 1976 of Lecture Notes in Computer Science, Springer, Berlin, pp 44–57

16. Kasahara M, Sakai R (2004) 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, Electronic version: `http://search.ieice.org/2004/files/e000a01.htm\#e87-a,1,102`
17. Kasahara M, Sakai R (2004) 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, p 6
18. Coppersmith D, Stern J, Vaudenay S Attacks on the birational permutation signature schemes. In: Cr [11], pp 435–443
19. Coppersmith D, Stern J, Vaudenay S (1997) The security of the birational permutation signature schemes. J Cryptol 10:207–221
20. Theobald T (1995) How to break Shamir's asymmetric basis. In: Coppersmith D (ed) Advances in Cryptology—CRYPTO 1995, volume 963 of Lecture Notes in Computer Science, Springer, Berlin, pp 136–147
21. MacWilliams FJ, Sloane NJA (1991) The theory of error-correcting codes. Elsevier Science Publisher, ISBN 0-444-85193-3
22. Courtois N, Goubin L, Meier W, Tacier JD (2002) Solving underdefined systems of multivariate quadratic equations. In: Naccache D, Paillier P (eds) Public key cryptography—PKC 2002, volume 2274 of Lecture Notes in Computer Science, pp 211–227. Springer, Berlin
23. Computational Algebra Group, University of Sydney. The MAGMA computational algebra system for algebra, number theory and geometry, `http://magma.maths.usyd.edu.au/magma/`
24. Courtois N, Goubin L, Patarin J (2001) Quartz: Primitive specification (second revised version), `https://www.cosic.esat.kuleuven.ac.be/nessie` Submissions, Quartz, p 18
25. Toli I (2003) Cryptanalysis of HFE, arXiv preprint server, `http://arxiv.org/abs/cs.CR/0305034`, p 7
26. Wolf C, Preneel B (2005) Equivalent keys in HFE, C*, and variations. In: Vaudenay S (ed) Proceedings of MyCrypt 2005, volume 3715 of Lecture Notes in Computer Science, Springer, Berlin, pp 33–49, Extended version `http://eprint.iacr.org/2004/360/`, p 15
27. Wolf C, Preneel B (2005) Superfluous keys in $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic asymmetric systems. In: Vaudenay S (ed) Public key cryptography—PKC 2005, volume 3386 of Lecture Notes in Computer Science, Springer, Berlin, pp 275–287, Extended version `http://eprint.iacr.org/2004/361/`
28. Moh T (1999) A public key system with signature and master key function. Communi Algebra 27(5):2207–2222. Electronic version: `http://citeseer/moh99public.html`
29. Yang BY, Chen JM (2004) Rank attacks and defence in Tame-like multivariate PKC's. Cryptology ePrint Archive, Report 2004/061, `http://eprint.iacr.org/`, p 21
30. Ding J, Yin Z (2004) Cryptanalysis of TTS and Tame-like multivariate signature schemes. Pre-Proceedings of the The Third International Workshop for Applied PKI, Fukuoka, Japan, October 3–5, 2004.
31. Akivis MA, Goldberg VV (1972) An introduction to linear algebra and tensors. Dover New York
32. Braeken A, Wolf C, Preneel B (2005) A study of the security of unbalanced oil and vinegar signature schemes. In: Menezes AJ (ed) The cryptographer's track at RSA conference 2005, volume 3376 of Lecture Notes in Computer Science. Springer, Berlin, pp 13, cf `http://eprint.iacr.org/2004/222/`
33. Stinson DR (ed) (1993) Advances in cryptology—CRYPTO 1993, volume 773 of Lecture Notes in Computer Science. Springer, Berlin, ISBN 3-540-57766-1.
34. Wolf C (2004) Efficient public key generation for HFE and variations. In: Dawson, Klimm, (eds) Cryptographic algorithms and their uses 2004, QUT University, Brisbane
35. Wolf C, Braeken A, Preneel B (2004) Efficient cryptanalysis of RSE(2)PKC and RSSE(2)PKC. In: Conference on Security in Communication Networks—SCN 2004, volume 3352 of Lecture Notes in Computer Science, Springer, Berlin, pp 294–309, September 8–10, Extended version: `http://eprint.iacr.org/2004/237`
36. Wolf C, Preneel B (2004) Asymmetric cryptography: Hidden Field Equations. In: Neittaanmäki P, Rossi T, Korotov S, Oñate E, Périaux J, Knörzer D (eds) European congress on computational methods in applied sciences and engineering 2004. Jyväskylä University, p 20, extended version: `http://eprint.iacr.org/2004/072/`