A Generalization of the Rainbow Band Separation Attack and its Applications to Multivariate Schemes

Enrico Thomae

Horst Görtz Institute for IT-security
Faculty of Mathematics
Ruhr-University of Bochum, 44780 Bochum, Germany
enrico.thomae@rub.de

Abstract. The Rainbow Signature Scheme is a non-trivial generalization of the well known Unbalanced Oil and Vinegar (UOV) signature scheme (Eurocrypt '99) minimizing the length of the signatures. By now the Rainbow Band Separation attack is the best key recovery attack known. For some sets of parameters it is even faster than a direct attack on the public key. Unfortunately the available description of the attack does not provide deep insights.
In this article we provide another view on the Rainbow Band Separation attack using the theory of equivalent keys and a new generalization called *good keys*. Thereby we generalize the attack into a framework that also includes Reconciliation attacks. We further formally prove the correctness of the attack and show that it also performs well on all multivariate quadratic ($\mathcal{MQ}$) schemes that suffer from missing cross-terms. We apply our attack to break the MFE encryption scheme based on Diophantine equations, the Enhanced STS signature scheme and all its variants, as well as the MQQ Encryption and Signature schemes. In the case of Rainbow and Enhanced TTS we show that parameters have to be chosen carefully and that the remaining efficiency gain over UOV is small.

Key words: Multivariate Cryptography, Algebraic Cryptanalysis, Band Separation, Key Recovery Attack, Rainbow, Enhanced STS, Enhanced TTS, MFE, Diophantine Equations, MQQ-Enc, MQQ-Sig

1 Introduction

The main idea of our algebraic key recovery attack is the same as for the so-called Reconciliation attack on UOV [BBD09], but involves some new techniques like *good keys*, which are a generalization of *equivalent keys*. In section 3 we will see that the Rainbow Band Separation attack described in [DYC$^+$08] is a special case of our framework. In addition to the brief description of the Rainbow Band Separation attack in [DYC$^+$08], we are able to prove correctness. We revisit the attack on Rainbow $(2^8, 18, 12, 12)$ with complexity 2^{67}. As it is hard to use the additional bihomogeneous structure in a theoretical complexity analysis, we performed various experiments that suggests a real attack complexity of 2^{64}. Also other multivariate signature schemes like Enhanced STS, Enhanced TTS, MFE and MQQ suffer even more from missing cross-terms and thus could be attacked the same way. In section 4 we briefly introduce the STS signature scheme and its variants [TGTF10]. We give an attack, which is better than the best know HighRank attack on the scheme, and also break all variants of Enhanced STS proposed so far. A cryptanalysis of the latest proposal is given in section 5. We strongly disbelieve that there is a way to fix STS without ending up at the Rainbow or Oil, Vinegar and Salt signature scheme. In section 6 we apply our attack to Enhanced TTS [YC05] and show that, in contrast to Rainbow, it slightly benefits from the additional structure. Our attack reduce the claimed security of 2^{88} to 2^{47}. In section 7 we apply our attack to the MFE signature scheme based on Diophantine equations [GH11] and give a key recovery in 2^{57} instead of 2^{113}, as claimed by the authors. In section 8 we apply our attack to MQQ-Enc [GS12] and MQQ-Sig [FGJ$^+$11] and obtain a key recovery attack which is as efficient as the original decryption algorithm for some sets of parameters. For all readers not familiar with multivariate schemes, we briefly introduce the general idea and basic notations in section 2.

2 Basic Facts

In this section we introduce the necessary notation and explain the most famous of all $\mathcal{MQ}$-schemes, namely the Unbalanced Oil and Vinegar signature scheme (UOV). It was proposed by Patarin *et al.* [KPG99] at Eurocrypt 1999 and is one of the oldest $\mathcal{MQ}$-schemes still unbroken. Understanding this simple and smart scheme is fundamental to understand the whole zoo of signatures that arose in the sequel.

The general idea of $\mathcal{MQ}$-signature schemes is to use a public multivariate quadratic map $\mathcal{P} : \mathbb{F}_q^n \to \mathbb{F}_q^m$ with

$$\mathcal{P} = \begin{pmatrix} p^{(1)}(x_1, \ldots, x_n) \\ \vdots \\ p^{(m)}(x_1, \ldots, x_n) \end{pmatrix}$$

and

$$p^{(k)}(x_1, \ldots, x_n) := \sum_{1 \le i \le j \le n} \gamma_{ij}^{(k)} x_i x_j = x^\intercal \mathfrak{P}^{(k)} x,$$

where $\mathfrak{P}^{(k)}$ is the $(n \times n)$ matrix describing the quadratic form of $p^{(k)}$ and $x = (x_1, \ldots, x_n)^\intercal$. Note that we can neglect linear and constant terms as they never mix with quadratic terms and thus have no positive effect on security. In the case of Enhanced TTS those linear terms will even decrease security as we will see later.

The trapdoor is given by a structured central map $\mathcal{F} : \mathbb{F}_q^n \to \mathbb{F}_q^m$ with

$$\mathcal{F} = \begin{pmatrix} f^{(1)}(u_1, \ldots, u_n) \\ \vdots \\ f^{(m)}(u_1, \ldots, u_n) \end{pmatrix}$$

and

$$f^{(k)}(u_1, \ldots, u_n) := \sum_{1 \le i \le j \le n} \gamma_{ij}^{(k)} u_i u_j = u^\intercal \mathfrak{F}^{(k)} u.$$

In order to hide this trapdoor we choose two secret linear transformations S, T and define $\mathcal{P} := T \circ \mathcal{F} \circ S$. See figure 1 for illustration.

$$\begin{tikzcd} \mathbb{F}_q^n \arrow[r, "\mathcal{P}"] \arrow[d, "S"'] & \mathbb{F}_q^m \\ \mathbb{F}_q^n \arrow[r, "\mathcal{F}"'] & \mathbb{F}_q^m \arrow[u, "T"'] \end{tikzcd}$$

Fig. 1. $\mathcal{MQ}$-Scheme.

For the UOV signature scheme the variables u_i with $i \in V := \{1, \ldots, v\}$ are called *vinegar variables* and the remaining variables u_i with $i \in O := \{v+1, \ldots, n\}$ are called *oil variables*. The central map $f^{(k)}$ is given by

$$f^{(k)}(u_1, \ldots, u_n) := \sum_{i \in V, j \in V} \gamma_{ij}^{(k)} u_i u_j + \sum_{i \in V, j \in O} \gamma_{ij}^{(k)} u_i u_j.$$

The corresponding matrix $\mathfrak{F}^{(k)}$ is depicted in figure 2.

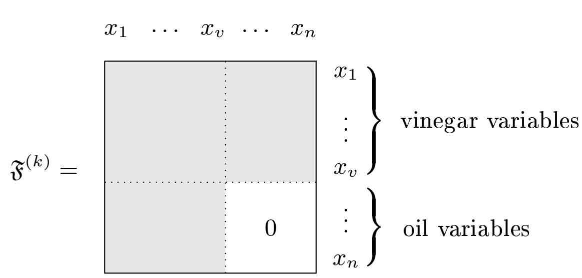


Fig. 2. Central map $\mathfrak{F}$ of UOV. White parts denote zero entries while gray parts denote arbitrary entries.

As we have m equations in $m+v$ variables, fixing v variables will yield a solution with high probability. Due to the structure of $\mathfrak{F}^{(k)}$, *i.e.* there are no quadratic terms of two oil variables, we can randomly fix the vinegar variables to obtain a system of linear equations in the oil variables, which is easy to solve. This procedure is not possible for the public key, as the transformation S of variables fully mixes the variables (like oil and vinegar in a salad). Note that for UOV we can discard the transformation T, as the trapdoor is invariant under this linear transformation of equations.

3 Cryptanalysis of Rainbow

Rainbow was proposed in 2005 [DS05] and is a layer-based variant of the well known multivariate quadratic ($\mathcal{MQ}$) signature scheme Unbalanced Oil and Vinegar (UOV). The downside of UOV is a comparably large signature expansion by a factor of 3 for current parameters ($m = 28, n = 84$) [TW12b]. Rainbow improves this to signatures of length $n = 42$ for messages of length $m = 24$, also for current parameters $(2^8, 18, 12, 12)$ [DYC$^+$08]. In the original paper [DS05] this improvement was even larger, but Billet and Gilbert [BG06] broke the parameter set $(2^8, 6, 6, 5, 5, 11)$ in 2006 using a *MinRank*-Attack. The idea used by Billet and Gilbert was known since 2000 and first proposed in [GC00]. At Crypto 2008 Faugère *et al.* [FdVP08] refined the technique of Billet and Gilbert using Gröbner Bases. Ding *et al.* took this attack into account and proposed new parameters of Rainbow [DYC$^+$08] claimed to be secure against all known attacks. In Algorithm 3 of [DYC$^+$08] the authors also described the Rainbow Band Separation attack, which they discovered in cooperation with Yu-Hua Hu.

Up to now the parameter set $(2^8, 18, 12, 12)$ is still close to secure, even due to two recent developments. Firstly in 2009 Bettale *et al.* published the HybridF_5 approach [BFP09] and thus reduced the complexity of a direct attack on the public key of Rainbow $(2^8, 18, 12, 12)$ to 2^{77}. And secondly in 2011 Faugère *et al.* [FDS11] analyzed systems of bihomogeneous equations and gave an upper bound on the degree of regularity for F_4. This immediately reduced the complexity of MinRank-Attacks on Rainbow $(2^8, 18, 12, 12)$ to $2^{80.8}$. But anyway, neither of these techniques drastically reduced the security of Rainbow. We refer to Petzold *et al.* [PBB10] for a comprehensive comparison of all known attacks on Rainbow and proposals for secure parameters.

Rainbow uses the same idea as UOV but in different layers. A current choice of parameters is given by $(q, v_1, o_1, o_2) = (2^8, 18, 12, 12)$. In particular the field size $q = 2^8$ and the number of layers is two. Note, two layers seems to be the best choice in order to prevent MinRank attacks and preserve short signatures at the same time. The central map $\mathcal{F}$ of Rainbow is divided into two layers $\mathfrak{F}^{(1)}, \ldots, \mathfrak{F}^{(12)}$ and $\mathfrak{F}^{(13)}, \ldots, \mathfrak{F}^{(24)}$ of form given in fig. 3. A formal description is given by the following formula.

$$\begin{aligned}
f^{(k)}(u_1, \ldots, u_n) := & \sum_{i \in V_1, j \in V_1} \gamma_{ij}^{(k)} u_i u_j + \sum_{i \in V_1, j \in O_1} \gamma_{ij}^{(k)} u_i u_j \\
& \text{for } k = 1, \ldots, o_1 \\
f^{(k)}(u_1, \ldots, u_n) := & \sum_{i \in V_1 \cup O_1, j \in V_1 \cup O_1} \gamma_{ij}^{(k)} u_i u_j + \sum_{i \in V_1 \cup O_1, j \in O_2} \gamma_{ij}^{(k)} u_i u_j \\
& \text{for } k = o_1 + 1, \ldots, o_1 + o_2
\end{aligned}$$

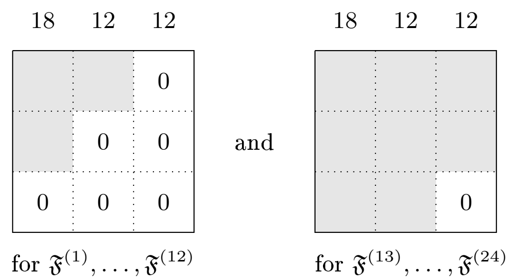


Fig. 3. Central map of Rainbow $(2^8, 18, 12, 12)$. White parts denote zero entries while gray parts denote arbitrary entries.

To use the trapdoor we first solve the small UOV system $\mathfrak{F}^{(1)}, \ldots, \mathfrak{F}^{(12)}$ by randomly fixing the 18 vinegar variables. The solution $u_1, \ldots, u_{30}$ is now used as vinegar variables of the second layer. Solving the obtained linear system yields $u_{31}, \ldots, u_{42}$.

Algebraic Cryptanalysis of Rainbow. Now we investigate what the special structure of $\mathfrak{F}$ tells us about the secret keys S and T. More precisely an algebraic key recovery attack exploits the special structure of $\mathfrak{F}$, *i.e.* zero entries at certain known places, to obtain equations in $\widetilde{T} := T^{-1} =: (\widetilde{t}_{ij})$ and $\widetilde{S} := S^{-1}$ through the following equality, which we obtain from $\mathcal{F} = T^{-1} \circ \mathcal{P} \circ S^{-1}$.

$$\mathfrak{F}^{(i)} = \widetilde{S}^{\intercal} \left(\sum_{j=1}^{m} \widetilde{t}_{ij} \mathfrak{P}^{(j)} \right) \widetilde{S} \tag{1}$$

As $\mathfrak{P}$ is publicly known and we further know that some specified entries of $\mathfrak{F}$ have to be zero, we obtain cubic equations in the elements of $\widetilde{S}$ and $\widetilde{T}$. The key observation is that the equations obtained by the fact that the coefficient of $u_i u_j$ in $f^{(k)}$ is zero are of the form

$$0 = \sum_{x=1}^{n} \sum_{y=1}^{n} \sum_{z=1}^{n} \alpha_{xyz} \widetilde{t}_{kx} \widetilde{s}_{yi} \widetilde{s}_{zj} \tag{2}$$

for some coefficients $\alpha_{xyz} \in \mathbb{F}_q$ that depend on $\mathfrak{P}^{(j)}$ (cf. [PTBW11, Sec. 3] or [TW12b] for an explicit formula). In particular every monomial contains one variable of the i-th column and one variable of the j-th column of $\widetilde{S}$. We will later make heavily use of this fact. But first let us calculate the complexity of a key recovery attack up to this point. Let us define $V_1 := \{u_1, \ldots, u_{v_1}\}$, $O_1 := \{u_{v_1+1}, \ldots, u_{v_1+o_1}\}$, $O_2 := \{u_{v_1+o_1+1}, \ldots, u_{v_1+o_1+o_2}\}$ and $O \times V := \{\{u, v\} \,|\, u \in O, v \in V\}$. The number of equations obtained by (1) equals the number of systematic zeros in all the $f^{(k)}$ and thus is

$$(o_1 + o_2) \cdot |(O_2 \times O_2)| + o_1 \cdot (|(O_2 \times (O_1 \cup V_1))| + |(O_1 \times O_1)|) = 7128.$$

The number of variables in $\widetilde{S}$ and $\widetilde{T}$ is given by $(v_1+o_1+o_2)^2+(o_1+o_2)^2 = 2340$. The complexity of solving such a system of equations using some Gröbner Basis algorithm like F_4 is 2^{3608} (cf. [BFSY05]). In a nutshell, we first have to calculate the degree of regularity d_{reg}. For semi-regular sequences, which generic systems are assumed to be, the degree of regularity is the index of the first non-positive coefficient in the Hilbert series $S_{m,n}$ with

$$S_{m,n} = \frac{\prod_{i=1}^{m}(1-z^{d_i})}{(1-z)^n}, \tag{3}$$

where d_i is the degree of the i-th equation, m is the number of equations and n the number of variables. The complexity of solving a zero-dimensional (semi-regular) system using F_4 is

$$\mathcal{O}\left(\binom{n+d_{reg}}{d_{reg}}^{\alpha}\right),$$

with $2 \le \alpha \le 3$ the linear algebra constant. The internal equations used by F_4 are very sparse and thus $\alpha = 2$ can be used to obtain a lower bound on the complexity. Well, if we really want to break a scheme, we either calculate the correct α or use $\alpha = 2.8$ as upper bound. Note that (3) changes for small fields, *i.e.* if the degree of regularity is larger than the number of elements in the field. Note further that as soon as the equations contain some structure, *e.g.* they are bihomogeneous, the complexity of solving them decrease [FDS11]. As our equations are partly bihomogeneous, the complexity 2^{3608} is just an upper bound. Unfortunately a theoretical complexity analysis of structured $\mathcal{MQ}$-systems is a very important open problem.

A first improvement of this upper bound complexity can be achieved by using *equivalent keys*, a notion introduced by Wolf and Preneel [WP05].

Definition 1 (Equivalent keys for Rainbow (v_1, o_1, o_2)). *Let $S = (s_{ij})_{n,n}$ and $T = (t_{ij})_{m,m}$ be two regular matrices. We label the equations given in (2) by (k, i, j) and define*

$$\begin{aligned}\mathbb{S} := \{(k,i,j) \mid\, & (1 \le k \le o_1 \wedge 1 \le i \le n \wedge \max\{v_1+o_1+1, i\} \le j \le n)\\ & \vee (1 \le k \le o_1 \wedge v_1 < i \le v_1+o_1 \wedge i \le j \le v_1+o_1)\\ & \vee (o_1 < k \le o_1+o_2 \wedge o_1+o_2 < i \le n \wedge i \le j \le n)\end{aligned}$$

the set of all equations obtained by systematic zero coefficients in the central map $\mathcal{F}$. Let S and T be valid solutions of $\mathbb{S}$. We call two regular matrices $S'_{n,n}$ and $T'_{m,m}$ equivalent keys*, if they also fulfill all equations in $\mathbb{S}$.*

Or in other words, if S and T are secret keys for the corresponding central map $\mathcal{F}$ then we call S' and T' *equivalent keys*, if $T \circ \mathcal{F} \circ S = \mathcal{P} = T' \circ \mathcal{F}' \circ S'$ for a valid trapdoor $\mathcal{F}'$. That means S' and T' preserve the structure of $\mathcal{F}$, *i.e.* preserve all systematic zero coefficients. Each equivalent key is sufficient for an attacker to use the trapdoor. Choosing a special representative of the class of equivalent

keys will now allow us to reduce the number of variables in S and T. Lets denote $\widetilde{S} := S^{-1}$ and $\widetilde{T} := T^{-1}$.
We first consider all transformations $\Omega^{-1}u = \Omega^{-1}Sx$, such that

$$x^\intercal S^\intercal \mathfrak{F} S x = x^\intercal S^\intercal (\Omega^{-1})^\intercal \Omega^\intercal \mathfrak{F} \Omega \Omega^{-1} S x$$

and $\Omega^\intercal \mathfrak{F} \Omega$ preserves the special structure of $\mathcal{F}$.

Obviously we are allowed to map $V_1 \mapsto V_1$ as these monomials exist anyway. What we are not allowed is to map $O_1 \cup O_2 \mapsto V_1$ as this would destroy the zero coefficients of monomials in $(O_1 \times O_1)$ and $(O_2 \times O_2)$ in the first layer equations. With the same argument we are allowed to map $V_1 \cup O_1 \mapsto O_1$ and $V_1 \cup O_1 \cup O_2 \mapsto O_2$, *i.e.* $\Omega^{-1}S = \widetilde{S}\Omega$ needs to be of the following form.

$$S' = \widetilde{S}\Omega = \begin{pmatrix} \widetilde{S}^{(1)}_{(v_1 \times v_1)} & \widetilde{S}^{(2)}_{(v_1 \times o_1)} & \widetilde{S}^{(3)}_{(v_1 \times o_2)} \\ \widetilde{S}^{(4)}_{(o_1 \times v_1)} & \widetilde{S}^{(5)}_{(o_1 \times o_1)} & \widetilde{S}^{(6)}_{(o_1 \times o_2)} \\ \widetilde{S}^{(7)}_{(o_2 \times v_1)} & \widetilde{S}^{(8)}_{(o_2 \times o_1)} & \widetilde{S}^{(9)}_{(o_2 \times o_2)} \end{pmatrix} \begin{pmatrix} \Omega^{(1)}_{(v_1 \times v_1)} & 0 & 0 \\ \Omega^{(2)}_{(o_1 \times v_1)} & \Omega^{(3)}_{(o_1 \times o_1)} & 0 \\ \Omega^{(4)}_{(o_2 \times v_1)} & \Omega^{(5)}_{(o_2 \times o_1)} & \Omega^{(6)}_{(o_2 \times o_2)} \end{pmatrix}$$

If $\widetilde{S}^{(9)}$ is regular, which is true with high probability (0.996 for $o_2 = 12$) then there exists $\Omega^{(6)}$ such that $S'^{(9)} = \widetilde{S}^{(9)}\Omega^{(6)} = I$. If $\widetilde{S}^{(9)}$ and $\widetilde{S}^{(5)}$ are regular, which is true with high probability (0.992 for $o_1 = 12$), then $\begin{pmatrix} \widetilde{S}^{(5)} & \widetilde{S}^{(6)} \\ \widetilde{S}^{(8)} & \widetilde{S}^{(9)} \end{pmatrix}$ is regular, too. Thus there exist $\Omega^{(3)}$ and $\Omega^{(5)}$, such that $S'^{(5)} = I$ and $S'^{(8)} = 0$. As we know that $\widetilde{S}$ is regular, it always exist $\Omega^{(1)}$, $\Omega^{(2)}$ and $\Omega^{(3)}$, such that $S'^{(1)} = I$, $S'^{(4)} = 0$ and $S'^{(7)} = 0$. To conclude, with high probability (0.992) there exist an equivalent key S' of the form given in figure 4. Note that we can randomize the algorithm by permuting columns and rows and thus start again, if finding S' fails. The same holds for the transformation of equations T, as we always can add equations within the same layer, as well as equations of the first to the second layer, without destroying the zero coefficients. Thus with overwhelming probability it exists an equivalent key T' of the form given in figure 4.

The total number of variables is now reduced to $v_1(o_1 + o_2) + 2o_1o_2 = 720$. The number of equations stays the same, but as the first v_1 columns of S' does no longer contain any variables, the corresponding $o_1 \cdot |(O_2 \times V_1)|$ equations transform from cubic to quadratic and furthermore are bihomogeneous in s'_{ij} and t'_{ij}. In our case we have 2592 quadratic and 4536 cubic equations. The complexity of solving this system by F_4 is 2^{374} which still is infeasible. To further decrease this complexity we now introduce the notion of *good keys*.
The overall idea is to decrease the number of variables in S' and T' as far as possible while preserving a reasonable amount of equations at the same time. Therefore we generalize the notion of equivalent keys to keys that do not preserve the *whole* structure of $\mathcal{F}$ but just some of it. We call those keys *good keys* if they also reveal some parts of the keys S' and T', respectively.

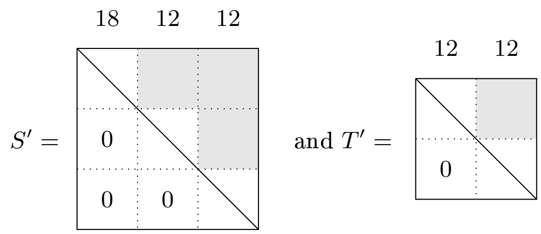


Fig. 4. Equivalent keys for Rainbow $(2^8, 18, 12, 12)$. White parts denote zero entries, gray parts denote arbitrary entries and there are ones at the diagonal.

Definition 2 (Good keys for Rainbow (v_1, o_1, o_2)). *Let $\mathbb{S}$ be the set defined in definition 1, $\mathbb{S}' \subseteq \mathbb{S}$ and S, T equivalent keys. We call two regular matrices $\widehat{S}$ and $\widehat{T}$* good keys*, if they fulfill all equations in $\mathbb{S}'$ and the sets*

$$\{(i,j) \mid s_{ij} = \widehat{s}_{ij} \text{ and } (1 \le i \le v_1 \wedge v_1 < j \le n) \vee (v_1 < i \le v_1 + o_1 \wedge v_1 + o_1 < j \le n)\}$$

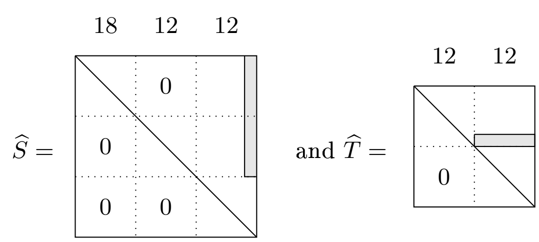
and

$$\{(i,j) \mid t_{ij} = \widehat{t}_{ij} \text{ for } (1 \le i \le o_1 \wedge o_1 < j \le o_1 + o_2)\}$$

are both not empty.

At a first glance it is not clear that good keys even exists. The following lemma proves the existence of good keys and give a special class of them.

Lemma 1. *Let S' and T' be equivalent keys for Rainbow of the form given in figure 4. Then there exist good keys $\widehat{S}$ and $\widehat{T}$, of the following form.*



Only the last column of $\widehat{S}$ contains arbitrary values in the first two blocks, which are equal to the corresponding values in S'. Respectively, only the second block of the o_1-th row of $\widehat{T}$ contains arbitrary values, which are equal to the corresponding values in T'.

Proof. We first show that there exists a unique transformation $S'\Omega = \widehat{S}$, if we assume $\Omega_{n1} = \ldots = \Omega_{n(v_1+o_1)} = 0$. We need those zeros later, to preserve a minimal amount of structure in $\mathcal{F}$.

$$S'\Omega := \begin{pmatrix} I & S'^{(1)}_{(v_1 \times o_1)} & S'^{(2)}_{(v_1 \times o_2)} \\ 0 & I & S'^{(3)}_{(o_1 \times o_2)} \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} \Omega^{(1)}_{(v_1 \times v_1)} & \Omega^{(2)}_{(v_1 \times o_1)} & \Omega^{(3)}_{(v_1 \times o_2 - 1)} || 0 \\ \Omega^{(4)}_{(o_1 \times v_1)} & \Omega^{(5)}_{(o_1 \times o_1)} & \Omega^{(6)}_{(o_1 \times o_2 - 1)} || 0 \\ \Omega^{(7)}_{(o_2 \times v_1)} & \Omega^{(8)}_{(o_2 \times o_1)} & \Omega^{(9)}_{(o_2 \times o_2)} \end{pmatrix} \stackrel{!}{=} \widehat{S}$$

Using linear algebra, we uniquely obtain $\Omega^{(4)} = \Omega^{(7)} = \Omega^{(8)} = 0$, $\Omega^{(1)} = \Omega^{(5)} = \Omega^{(9)} = I$, $\Omega^{(2)} = -S'^{(1)}$ and $\Omega^{(6)}_{(o_1 \times o_2 - 1)} = -S'^{(3)}_{(o_1 \times o_2 - 1)}$ as well as $\Omega^{(3)}_{(v_1 \times o_2 - 1)} = (S'^{(1)}S'^{(3)} - S'^{(2)})_{(v_1 \times o_2 - 1)}$. Obviously the last column of $S'^{(2)}$ and $S'^{(3)}$ are not affected by this transformation. Furthermore omitting the zeros in the last column of Ω would destroy *all* the structure in $\mathcal{F}$ (cf. figure 5).
As soon as we would allow to map u_n to any of the variables in V_1 or O_1 *all* the zero coefficients in $\mathcal{F}$ would vanish and thus no equations would be left to perform an algebraic attack with.
Showing that $\widehat{T}$ is a good key is trivial: If we just want to preserve the structure of $\mathfrak{F}^{(o_1)}$, we can forget everything but the o_1-th row of T'. □

The secret map $\mathfrak{F}' = \Omega^\intercal \mathfrak{F} \Omega$ is of the from given in figure 5.

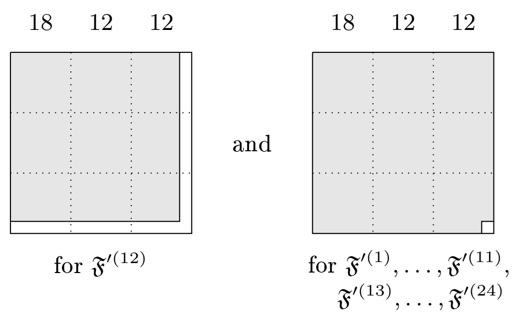


Fig. 5. Central map of Rainbow $(2^8, 18, 12, 12)$ after applying the transformation Ω given by lemma 1. White parts denote zero entries and gray parts denote arbitrary entries.

The total number of variables obtained by good keys chosen as above is $v_1 + o_1 + o_2 = 42$. To count the number of equations, we denote $n := v_1 + o_1 + o_2$ and label every equation obtained by a zero coefficient of $u_i u_j$ in $\mathfrak{F}^{(k)}$ by (i, j, k) (cf. equation (2)). First, (n, n, o_1) provides a cubic equation. Second, (n, n, i) for $i = 1, \ldots, o_1 - 1, o_1 + 1, \ldots, o_1 + o_2$ provides quadratic equations in the variables s_{ij}. Third and most important, (i, n, o_1) for $i = 1, \ldots, n - 1$ provides quadratic, bihomogeneous equations in s_{ij} and t_{ij}. Those equations are the main weakness of all layer based $\mathcal{MQ}$-primitives. Their existence is due to the missing cross-terms $V_1 \times O_2$ and $O_1 \times O_2$ in the first layer of Rainbow. Note that in the case of UOV these equations do not exist. Applying the same approach to UOV, provides m quadratic equations in $2m$ variables, which is infeasible for current parameters of $m = 28$ [TW12b]. For Rainbow $(2^8, 18, 12, 12)$ we end up with 1 cubic, 23 quadratic and, due to the missing cross-terms, 41 bihomogeneous equations. Note that the solution of the cubic equation is independent of t_{ij} as this equation still holds if we use $\widehat{T} = I$. So we actually deal with 24 quadratic equations. Solving this system of equations has a complexity of at least

$\binom{42+10}{10}^2 \approx 2^{67.7}$. Again this complexity estimation assumes generic equations. As our equations contain some special structure, *e.g.* some of them are bihomogeneous, we can hope for a lower complexity in practice. We implemented our attack and compared its running time to those of random systems (cf. table 2). This way we obtained an empirical complexity that is at least 2^{64}.
After we obtained one column of S' and one row of T', all the other parts of S' and T' are revealed by linear equations. More precisely, by equations (i,n,j) for $i = 1,\ldots,n$ and $j = 1,\ldots,o_1-1,o_1+1,\ldots,o_1+o_2$ we obtain $n(o_1+o_2-1)$ linear equations in the remaining $(o_1-1)o_2$ variables of T'. After we recovered T' all the equations (i,j,k) for $i = 1,\ldots,v_1$, $j = v_1+1,\ldots,n$ and $k = 1,\ldots,o_1+o_2$, and even some more, become linear. Solving this system of $v_1(o_1+o_2)^2$ linear equations in $(v_1+o_1-1)o_2+v_1o_1$ variables easily reveals the unique solution of S'.

To recap, we reduced a structured system of many equations and variables, which we could not theoretically analyze, to a less structured small system of few equations and variables, using good keys. Solving this small system somehow contains the hard core difficulty of solving the overall system, as all the other solutions follow by linear equations.

Table 1 gives the theoretical complexity of our attack for several parameters given in [PBB10] which are considered to be secure.

Table 1. Attack complexity for several parameter sets believed to be secure. Note, the parameters for small fields are still valid.

parameter set	field	attack $[\log_2]$
(18,13,14)	$\mathbb{F}_{2^8}$	69.5
(20,14,14)	$\mathbb{F}_{2^8}$	76.1
(17,18,17)	$\mathbb{F}_{31}$	78.3
(21,20,20)	$\mathbb{F}_{2^4}$	88.1

Experimental Results. We have implemented our attack using the software system Magma V2.16-1 [MAG]. All experiments were performed on a Intel Xeon X33502.66GHz (Quadcore) with 8 GB of RAM using only one core. Table 2 gives the results for various parameter sets (v_1, o_1, o_2) of Rainbow. Column 4 and 5 give the number of equations and variables obtained through our attack. Column 6 gives the $\log_2$ value of the theoretical complexity assuming random equations (cf. [BFSY05]). The following three columns show the time in seconds that our attack required over different fields. We guess that $\mathbb{F}_{2^8}$ is implemented more efficiently in Magma and thus it needs longer to solve instances over $\mathbb{F}_{2^4}$ than over $\mathbb{F}_{2^8}$. The last column describes the time it took us to solve a random instance with the same number of variables and equations, assuming that a solution exists.

Comparing these complexities to the ones of our attack, we observe a factor of 32 for $(6,4,4)$ which we are faster over $\mathbb{F}_{2^8}$ than theoretically expected. As this set of parameters is a scaled variant of $(18,12,12)$ and the difference only increases, we conclude that the attack is at least 32 times faster than theoretically evaluated. Thus we end up with an empirical complexity of at least $2^{62.7}$ to break Rainbow $(2^8, 18, 12, 12)$.

Table 2. Running times in seconds of our attack for different sets of parameters, over different fields. In comparison the running time in seconds for random systems is given in the last column, as well as a theoretical complexity in operations in column six.

v_1	o_1	o_2	#eq. m	#var. n	theoretical $[\log_2]$	attack $[s]$ GF(2^8)	attack $[s]$ GF(2^4)	attack $[s]$ GF(31)	random $[s]$ GF(2^8)
5	4	4	20	13	26	0.5	0.7	0.4	6
6	4	4	21	14	26	0.7	1.1	0.6	23
7	4	4	22	15	31	1.4	2.1	1.1	194
8	4	4	23	16	32	4.3	6.9	3.6	641
9	4	4	24	17	33	35	64	29	3328
6	5	5	25	16	28	17	29	15	87
7	5	5	26	17	32	33	58	25	1270
8	5	5	27	18	34	87	159	73	4475
9	5	5	28	19	34	630	1185	527	-
7	6	6	30	19	35	443	821	370	-
8	6	6	31	20	35	877	1765	743	-
9	6	6	32	21	36	3034	6052	2578	-
8	7	7	35	22	41	12567	25311	10730	-

Conclusion. A immediate consequence of our attack is that we should use at least parameters $(22,16,16)$ over $\mathbb{F}_{2^8}$. Further we did not use all the structure for the theoretical analysis of our attack, *i.e.* we neglected that a large portion of the obtained equations is bihomogeneous. Thus we should ask ourselves a very important question: Is the gain in efficiency by transforming UOV to Rainbow larger than the loss of security? If not, Rainbow is superfluous as UOV will always be both, more secure and efficient. This question especially arise because our attack on Rainbow use the missing cross-terms and thus is not applicable to UOV. Unfortunately, a fair comparison of the efficiency/security ratio of UOV and Rainbow is out of the scope of this paper. To even define efficiency in this context is an involved task. Do we only measure the extension factor of the signature or do we take the complexity of the signing algorithm into account, too? Our intuition is that we roughly lose as much security as we gain efficiency in terms of the signature length while transforming UOV to Rainbow. Let us explain this at the following example over $\mathbb{F}_{2^8}$. Using Rainbow $(2^8, 22, 16, 16)$, for which our *key recovery* attack has complexity 2^{84}, we map messages of length 32 to signatures of length 54. For comparison, UOV with parameters $o = 28$ and $v = 56$ is considered to have a security level of 2^{84} against *message recovery* attacks [BFP09,TW12b]. Thus UOV maps a message of length 28 to a signature of length 84. Further we can use that UOV is well parametrized, while Rainbow is built on the edge, *i.e.* in order to prevent key recovery attacks like the one of Kipnis and Shamir [KS98,KPG99] on UOV, we only have to ensure $v - o - 1 \geq 8$. So choosing $v = 2o$ is a little conservative. More precisely $o = 28$ and $v = 37$ is sufficient to prevent this type of key recovery attack. In this case UOV maps a message of length 28 to a signature of length 65. To put security concerns in a nutshell, UOV is based on the $\mathcal{MQ}$- and IP-problem and Rainbow additionally use the difficulty of the MinRank-problem. So everyone have to decide on his own, if obtaining signatures of length 54 instead of 65 is worthwhile to take another class of problems into account.

4 Cryptanalysis of Enhanced STS and all its Variants

Another way to achieve a secret map $\mathcal{F} = \left(f^{(1)}, \dots, f^{(m)}\right)^\intercal$ was given by the *Sequential Solution Method* of Tsujii [STH89,TKI$^+$86]. The idea was somehow similar to the independently proposed schemes of Shamir [Sha93] and Moh [Moh99]. In 2004 Kasahara and Sakai extended this idea to the so-called RSE system [KS04], which later was generalized to the *Stepwise Triangular System* (STS) by Wolf *et al.* [WBP04]. Here the central polynomials $f^{(k)}$ are some random quadratic polynomials in a restricted number of variables. See figure 6 for the stepped structure of the resulting $\mathcal{MQ}$-system. Inverting this map is possible as long as solving r quadratic equations in r variables is practical. Consequently, we need to restrict r to rather small values, *e.g.* $r = 4 \dots 9$.

$$
\begin{array}{ll}
\text{Step } 1 & \left\{ \begin{array}{l} f^{(1)}(u_1, \dots, u_r) \\ \quad\vdots \\ f^{(r)}(u_1, \dots, u_r) \end{array} \right. \\
 & \quad\vdots \\
\text{Step } i & \left\{ \begin{array}{l} f^{((i-1)r+1)}(u_1, \dots, u_{ir}) \\ \quad\vdots \\ f^{(ir)}(u_1, \dots, u_{ir}) \end{array} \right. \\
 & \quad\vdots \\
\text{Step } L & \left\{ \begin{array}{l} f^{((L-1)r+1)}(u_1, \dots, u_m) \\ \quad\vdots \\ f^{(m)}(u_1, \dots, u_m) \end{array} \right.
\end{array}
$$

resp.

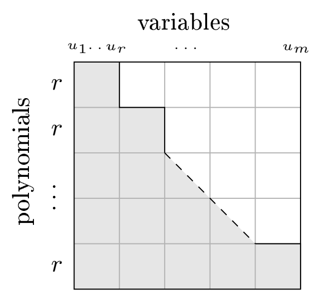


Fig. 6. Central map of STS based signature schemes like RSSE(2)PKC or RSE(2)PKC. The gray parts of the matrix indicate that those variables occur in the corresponding polynomial and white parts indicate that they do not.

In the same year Wolf *et al.* [WBP04] showed how to efficiently break the proposed parameters of the STS schemes RSSE(2)PKC and RSE(2)PKC using a HighRank attack. At PQCrypto 2010 Tsujii *et al.* [TGTF10] tried to fix the scheme by proposing a new variant called *Enhanced STS*, which uses a complementary STS structure (cf. figure 7). Only a few months later they noticed themselves that the scheme is obviously not immune to HighRank attacks, although this was originally a design goal. To fix this problem, they proposed several new variants [GT11,TG10]. We will now shortly repeat the HighRank attack and then give a more efficient algebraic key recovery attack which makes use of *good keys* and missing cross-terms. The latter are quadratic monomials of two variables from different sets, which do not exist in the central map $\mathcal{F}$

by construction. We conclude that it is impossible to find a secure and efficient parameter set of Enhanced STS. We will also break the new variants of STS. To conclude, we discuss (im)possible improvements and show that we either end up with the Rainbow or Oil, Vinegar and Salt signature scheme.

Cryptanalysis of Enhanced STS. To exploit different ranks in plain STS, we use the quadratic form of the polynomials $f^{(k)}$, *i.e.* $f^{(k)} = u^\intercal \mathfrak{F}^{(i)} u$ for $u = (u_1, \ldots, u_m)^\intercal$ and some $(m \times m)$ matrix $\mathfrak{F}^{(i)}$. Note that we have $n = m = Lr$ here. Obviously the rank of these matrices in the i-th step is ir. Now we use that the rank is invariant under the bijective transformation $S^{-1}u = x$ of variables, *i.e.* $\operatorname{rank}(S^\intercal \mathfrak{F}^{(i)} S) = \operatorname{rank}(\mathfrak{F}^{(i)})$ for all i. In addition, the public polynomials $p^{(i)} = x^\intercal \mathfrak{P}^{(i)} x$ are given by some linear combination $\mathfrak{P}^{(i)} = \sum_{j=1}^{m} t_{ij} S^\intercal \mathfrak{F}^{(j)} S = S^\intercal \left(\sum_{j=1}^{m} t_{ij} \mathfrak{F}^{(j)} \right) S$. As the rank is changed by the transformation of equations T, we can use the rank property of the underlying central equations $f^{(k)}$ as a distinguisher to obtain the full transformation T.

Enhanced STS was thought to resist rank attacks. Tsujii *et al.* introduced two sets $U = \{u_1, \ldots, u_m\}$ and $V = \{v_1, \ldots, v_{m-r}\}$ of variables and constructed central polynomials $f^{(k)}$ which all have the *same* rank m. The construction is very similar to figure 6, but every polynomial $f^{(k)}$ depends on m variables. See figure 7 for details.

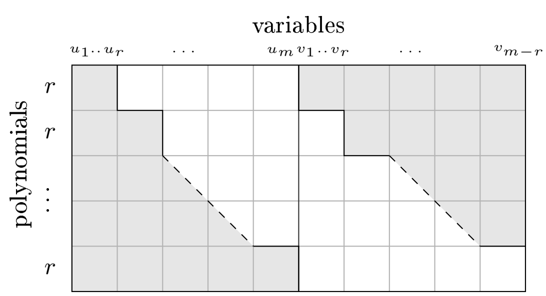


Fig. 7. Central map of Enhanced STS. The gray parts of the matrix indicate that those variables occur in the corresponding polynomial and white parts indicate that they do not.

As the corresponding $\mathcal{MQ}$-system $\mathcal{F}$ has m quadratic equations but $n = 2m - r$ variables, we could fix all variables of V to random values and obtain an $\mathcal{MQ}$-system of r equations and r variables in the first step. Solving this $\mathcal{MQ}$-system, substituting the solution in the next step and so on, allows for a reasonable efficient inversion of $\mathcal{F}$.

Tsujii *et al.* themselves noticed [TG10] that having the same rank m for the central polynomials $f^{(k)}$ does not prevent rank attacks in any way, as the rank of the public polynomials is $2m - r$. The following simple HighRank attack is still applicable. Note that due to the additional variables v_i the minimal rank of the central polynomials is m, for $m \geq 26$ in practice to prevent direct attacks. Thus Enhanced STS is at least secure against MinRank attacks [FdVP08,BG06].

HighRank Attack. In order to reconstruct T we have to search for linear combinations of the public polynomials $\mathfrak{P}^{(i)}$, such that the rank decrease from $2m-r$ to m. Let $\sigma \in S_m$ be a random permutation, which we need for randomization. Then there exist $\lambda_i \in \mathbb{F}_q$ such that the following linear combination has rank $2m - 2r$ and thus the rank drops by r.

$$\mathfrak{P}^{(\sigma(r+1))} + \sum_{i=1}^{r} \lambda_i \mathfrak{P}^{(\sigma(i))} =: \widetilde{\mathfrak{P}}$$

There are 2 different solutions, as we can eliminate the r matrices $\mathfrak{F}^{(1)}, \ldots, \mathfrak{F}^{(r)}$ or $\mathfrak{F}^{(m-r+1)}, \ldots, \mathfrak{F}^{(m)}$ such that $\widetilde{\mathfrak{P}}$ has rank $2m - 2r$. In the first case $\widetilde{\mathfrak{P}}$ is a linear combination of secret polynomials, who do not contain variables $v_1, \ldots, v_r$ respectively $u_{m-r+1}, \ldots, u_m$ in the latter case. Thus brute forcing all λ_i has complexity $q^r/2$. Once we have eliminated all the $\mathfrak{F}^{(i)}$ of one block (*e.g.* $1 \leq i \leq r$) in one polynomial $\widetilde{\mathfrak{P}}$ we easily eliminate those $\mathfrak{F}^{(i)}$ in all the other $m - r$ public polynomials by just determining $\ker(\widetilde{\mathfrak{P}})$. The linear system $\sum_{i=1}^{m} \lambda_i \mathfrak{P}^{(i)} \omega = 0$ with $\omega \in \ker(\widetilde{\mathfrak{P}})$ provides all $m-r$ polynomials of rank $2m-2r$. The complexity of this step is $2(2m - r)^3$. Repeating this whole procedure L times yields r matrices $\widetilde{\mathfrak{P}}^{(i)}$ of rank m. At this point we know the kernel of one of the central blocks of $\mathcal{F}$ and could use this to separate the matrices in the steps before, which are still linear combinations of some $S^{\intercal}\mathfrak{F}^{(i)}S$. Choosing a vector that lies in the kernel of the matrices obtained in the i-th step, but not in the kernel of matrices recovered in step $i+1, \ldots, L$ easily provides T. The overall complexity of this HighRank attack is given by

$$\frac{L}{2} q^r + 2L(2m - r)^3 + \sum_{i=1}^{L-1} (ir)^3 = \mathcal{O}(q^r).$$

Algebraic Key Recovery Attack. We saw that the complexity of the HighRank attack strongly depends on the field size q and the parameter r. Even if r is restricted to small values due to efficiency constraints, it is possible to choose q large enough to obtain a scheme secure against the previously mentioned attack. For example, let $r = 9$ and $q = 2^9$. Now we describe a new key recovery attack that is almost independent of the field size q and thus makes it impossible to find a parameter set that is both efficient and secure. To ease explanation we fix a parameter set of Enhanced STS to illustrate the attack. As there are no parameters given in [TG10], what is by the way not very courteous for cryptanalysis, we choose $m = 27$, $r = 9$ and $q = 2^9$ as this prevents message recovery attacks via Gröbner Bases on the public key as well as HighRank attacks. The number of steps is given by $L = m/r = 3$. The number of variables is $n = 2m - r = |U| + |V| = 27 + 18 = 45$. Note that a legitimate user would need to solve three generic $\mathcal{MQ}$-system with 9 equations and variables over $\mathbb{F}_{2^9}$ to compute a signature. While possible in theory, it is inefficient for practical use. Solving a generic $\mathcal{MQ}$-system with 9 equations and variables over $\mathbb{F}_{2^9}$ using the fastest known method, *i.e.* the hybrid approach [BFP09] by guessing one variable, as well as the very fast F_4 implementation of Magma V2.16-1 [MAG] on a Intel Xeon X33502.66GHz (Quadcore) with 4 GB of RAM using only one core, took us 0.3 seconds. Thus the worst case signing time is $3 \cdot 2^9 \cdot 0.3 \approx 461$ seconds. But despite of choosing such a large r, we now show that the resulting scheme still is not secure.

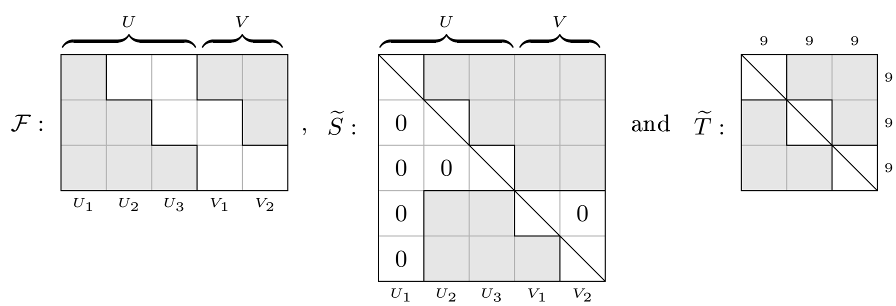


Fig. 8. Central map $\mathcal{F}$ of Enhanced STS and the minimal representative S and T of the class of equivalent keys.

Figure 8 shows the structure of the central map $\mathcal{F}$. The picture describing $\mathcal{F}$ has to be read like figure 7. Every little square denotes a (9×9) array. Moreover, we give the structure of the secret key $\widetilde{S} := S^{-1}$, which is a (45×45) matrix with ones at the diagonal, zeros at the white parts and unknown values at the gray parts. Note that there are many different secret keys S respectively S^{-1} that preserve the structure of $\mathcal{F}$, *i.e.* preserve systematical zero coefficients in the polynomials $f^{(i)}$. We call all them *equivalent keys* and can assume that in every class there is one representative with the structure given in figure 8 with overwhelming probability (cf. definition1). The same holds for $\widetilde{T} := T^{-1}$. We skip the derivation of $\widetilde{S}$ and $\widetilde{T}$ given in figure 8 as it was already known and is very similar to the proof of lemma 1.

An algebraic key recovery attack uses the special structure of $\mathcal{F}$ to obtain equations in $\widetilde{S}$ and $\widetilde{T}$ through the following equality (cf. (1)) derived from $\mathcal{F} = T^{-1} \circ \mathcal{P} \circ S^{-1}$ with $\widetilde{T} := T^{-1} =: (\widetilde{t}_{ij})$ and $\widetilde{S} := S^{-1}$.

$$\mathfrak{F}^{(i)} = \widetilde{S}^{\intercal} \left(\sum_{j=1}^{m} \widetilde{t}_{ij} \mathfrak{P}^{(j)} \right) \widetilde{S}$$

As $\mathfrak{P}$ is publicly known and we further know that some of the entries of $\mathfrak{F}$ are systematically zero, we obtain cubic equations in the elements of $\widetilde{S}$ and $\widetilde{T}$. To ease notation we use $u_{j+m} := v_j$ for $j = 1, \ldots, m-r$. It is interesting to observe that the equations obtained from the coefficients $u_i u_j$ in $f^{(k)}$ are of the form

$$0 = \sum_{x=1}^{n} \sum_{y=1}^{n} \sum_{z=1}^{n} \alpha_{xyz} \widetilde{t}_{kx} \widetilde{s}_{yi} \widetilde{s}_{zj}$$

for some coefficients $\alpha_{xyz} \in \mathbb{F}_q$ that depend on the public key matrices $\mathfrak{P}^{(j)}$ (cf. [PTBW11, Sec. 3] or [TW12b] for an explicit formula). Due to the special form of $\widetilde{S}$ this immediately implies that all equations obtained by zero monomials $u_i u_j$ with $u_i \in U_1 := \{u_1, \ldots, u_9\}$ and $u_j \in U_2 \cup U_3 := \{u_{10}, \ldots, u_{18}\} \cup \{u_{19}, \ldots, u_{27}\}$, as well as $u_i v_j$ with $u_i \in U_1$ and $v_j \in V_1 \cup V_2 := \{v_1, \ldots, v_9\} \cup \{v_{10}, \ldots, v_{18}\}$ become quadratic instead of cubic. This change hence greatly improves the overall attack complexity. Defining $U \times V := \{\{u, v\} \mid u \in U, v \in V\}$ the total amount of equations obtained by systematical zeros in $\mathcal{F}$ is

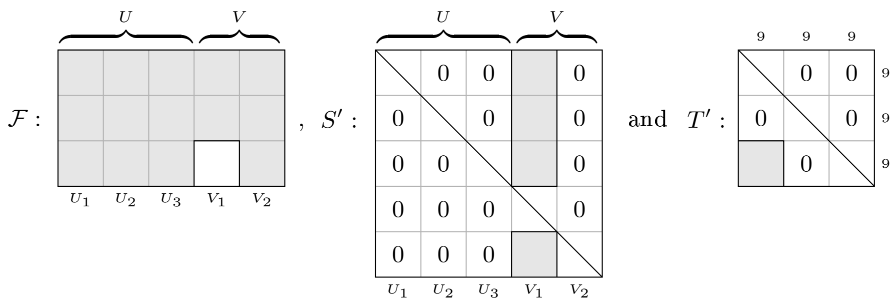
$$\begin{aligned}
& 9 \cdot (|(U_2 \cup U_3) \times (U_2 \cup U_3)| + |(U_2 \cup U_3) \times (V_1 \cup V_2)|) \\
+ & 9 \cdot (|(U_3 \cup V_1) \times (U_3 \cup V_1)| + |(U_3 \cup V_1) \times (U_2 \cup V_2)|) \\
+ & 9 \cdot (|(V_1 \cup V_2) \times (V_1 \cup V_2)| + |(V_1 \cup V_2) \times (U_2 \cup U_3)|) \\
= & 9 \cdot 3 \cdot ((18 \cdot 19)/2 + 18 \cdot 18) \\
= & 27 \cdot (171 + 324) = 13,365 \text{ cubic equations and}
\end{aligned}$$

$$\begin{aligned}
& 9 \cdot |(U_2 \cup U_3) \times U_1| + 9 \cdot |(U_3 \cup V_1) \times U_1| + 9 \cdot |(V_1 \cup V_2) \times U_1| \\
= & 27 \cdot 162 = 4374 \text{ quadratic equations.}
\end{aligned}$$

Solving this system of equations in 486 variables $\widetilde{t}_{ij}$ and 1134 variables $\widetilde{s}_{ij}$ with a common Gröbner basis algorithm like F_4 has a total complexity of 2^{877} (cf. [BFS04,BFSY05]). This huge complexity is due to the large number of variables and the fact that the complexity estimation assumes *generic equations* and thus does not take the structure of the equations into account. In order to decrease the complexity, we have to break down the problem into smaller pieces. This can be done if we further decrease the number of variables in $\widetilde{S}$ and $\widetilde{T}$. To achieve this goal we use good keys again (cf. definition 2).

Lemma 2. *Let $\widetilde{S}$ and $\widetilde{T}$ be equivalent keys for Enhanced STS of the form given in figure 8. Then there exist good keys S' and T', of the following form.*
S' is all zero except the gray parts, which are equal to the corresponding values in $\widetilde{S}$ and the diagonal, which contains only ones. Similarly, the gray parts of T' equals the corresponding values in $\widetilde{T}$.

Proof. To preserve the structure of $\mathcal{F}$ given in lemma 2 we are allowed to map variables $U_1 \cup U_2 \cup U_3 \cup V_2 \mapsto U_1 \cup U_2 \cup U_3 \cup V_2$ as well as $V_1 \mapsto V_1$. As soon as we were to map variables from V_1 to any other set of variables, all polynomials would contain variables from V_1 and thus the whole structure of $\mathcal{F}$ would be



destroyed. Now we show that using such a transformation Ω of variables, we can uniquely map $\widetilde{S}$ to S' by $\widetilde{S}\Omega = S'$.

$$\widetilde{S}\Omega := \begin{pmatrix} I & \widetilde{S}^{(1)} & \widetilde{S}^{(2)} & \widetilde{S}^{(3)} & \widetilde{S}^{(4)} \\ 0 & I & \widetilde{S}^{(5)} & \widetilde{S}^{(6)} & \widetilde{S}^{(7)} \\ 0 & 0 & I & \widetilde{S}^{(8)} & \widetilde{S}^{(9)} \\ 0 & \widetilde{S}^{(10)} & \widetilde{S}^{(11)} & I & 0 \\ 0 & \widetilde{S}^{(12)} & \widetilde{S}^{(13)} & \widetilde{S}^{(14)} & I \end{pmatrix} \begin{pmatrix} \Omega^{(1)} & \Omega^{(2)} & \Omega^{(3)} & 0 & \Omega^{(4)} \\ \Omega^{(5)} & \Omega^{(6)} & \Omega^{(7)} & 0 & \Omega^{(8)} \\ \Omega^{(9)} & \Omega^{(10)} & \Omega^{(11)} & 0 & \Omega^{(12)} \\ \Omega^{(13)} & \Omega^{(14)} & \Omega^{(15)} & \Omega^{(16)} & \Omega^{(17)} \\ \Omega^{(18)} & \Omega^{(19)} & \Omega^{(20)} & 0 & \Omega^{(21)} \end{pmatrix} \stackrel{!}{=} S'$$

Obviously $\Omega^{(16)} = I$ and thus $\widetilde{S}^{(3)}, \widetilde{S}^{(6)}$, $\widetilde{S}^{(8)}$ and $\widetilde{S}^{(14)}$ remain unchanged. As $\widetilde{S}$ is regular, all other $\Omega^{(i)}$ are uniquely determined by $\widetilde{S}^{-1}S'$. Showing that T' is a good key is trivial: If we only want to $f_{2r+1}, \ldots, f_{3r}$ to contain no V_1 variables, we are allowed to map all polynomials except $f_1, \ldots, f_r$ to one another. □

Using the good keys of lemma 2 we end up with 405 cubic equations, 2916 quadratic equations and 405 variables. The complexity of solving such a system using F$_4$ is still 2^{151}. To bring this game to an end, we only need to assure that f_{30} do not contain the variable v_1. Analogous to lemma 2 we obtain $|(U \cup V_2 \cup V_1 \backslash \{v_1\}) \times \{v_1\}| = 44$ quadratic equations and one cubic equation. Using good keys analogous to lemma 2 we obtain 9 variables t_{27j} for $1 \leq j \leq 9$ as well as 36 variables s_{i28} for $1 \leq i \leq 36$. Applying the generic complexity analysis as before still provides the same, and hence infeasible complexity of 2^{151}. The reason is that now the number of equations equals the number of variables, so the overall complexity does not decrease. To obtain a better attack complexity we somehow have to use the fact that all quadratic equations are bihomogeneous, *i.e.* of the form $\sum_{i=1}^{36} \sum_{j=1}^{9} \alpha_{ij} t_{27j} s_{i28}$ for some $\alpha_{ij} \in \mathbb{F}_q$. In [FDS11] Faugère *et al.* analyzed systems of such a special structure and gave an upper bound on the degree of regularity for F$_4$. To use their results we first have to guess one variable t_{ij} such that we obtain a system of 44 bihomogeneous equations in 44 variables. According to their results we now obtain a degree of regularity of 9 and a complexity of $2^9 \binom{44+9}{9}^2 \approx 2^{73}$. In general the degree of regularity is r, as we have $r-1$ variables t_{ij} after guessing and thus the complexity of our attack for arbitrary parameters is given by

$$q\binom{2m-1}{r}^2.$$

Once we obtained a single row/column of $\widetilde{S}$ and $\widetilde{T}$, the whole system breaks down as all other elements are now determined through linear equations. Therefore let us label every equation obtained by a zero coefficient of $u_i u_j$ in $f^{(k)}$ by (u_i, u_j, k) (cf. (2)). Now, (u_i, v_1, k) and (v_j, v_1, k) with $i = 1, \ldots, 27$, $j = 1, \ldots, 18$ and $k = 19, \ldots, 26$ provide linear equations in t_{ij} with $i = 19, \ldots, 26$ and $j = 1, \ldots, 9$. Next we can apply the same approach using good keys as above for v_1 to v_i, $i = 2, \ldots, 9$. As we already know the coefficients t_{ij} of the appropriate good key, all bihomogeneous equations become linear in s_{ij}. We now can determine the next blocks in T through linear equations only. We repeat the process until all secret coefficients are recovered.

To summarize our new attack, we first used the fact that cross-terms from $(U \cup V_2) \times V_1$ do not exist and thus obtained quadratic instead of cubic equations in the key recovery attack. Second, we reduced the number of variables through good keys. And third, we used the special bihomogeneous structure of the equations to lower the attack complexity. In order to protect the scheme against this attack we either have to increase m or r. But as the complexity of the signing algorithm is $3q\binom{r-1+d_{reg}}{r-1}^2$, *i.e.* in the same order of magnitude of our attack, Enhanced STS cannot be efficient and secure at the same time. In general it do not seem to be a good idea to use an exponential time signing algorithm.

Cryptanalysis of Check Equation Enhanced STS. The original Enhanced STS scheme contains m quadratic equations in $2m - r$ variables in the public key and thus have q^{m-r} possible valid signatures to one message. Even if current algorithms cannot take much advantage of underdetermined $\mathcal{MQ}$-systems [TW12b], Tsujii *et al.* [TG10] suggested to strength their signature by adding $m - r$ check equations and thus fix one unique signature. From a message recovery point of view, the attacker now would have to solve a $\mathcal{MQ}$-system of $2m - r$ (public key) equations and variables. Before he had to solve a system of m equations and variables after just guessing the additional $m - r$ variables.

However, the check equations do not affect the algebraic key recover attack we just described. Moreover, if the check equations are not chosen purely random and thus introducing new structure, the attack may even benefit.

Cryptanalysis of Hidden Pair of Bijection. The overall idea of this variant is very general. Take a pair $F^{(1)}, F^{(2)} : \mathbb{F}_q^m \to \mathbb{F}_q^m$ of bijections with a disjoint set of variables, *i.e.* $u = (u_1, \ldots, u_m)$ and $v = (v_1, \ldots, v_m)$ and connect them with a function H containing all the cross-terms of u and v. The central polynomial $f^{(k)}$ is given by

$$f^{(k)}(u, v) := F_1(u) + F_2(v) + H(u, v) \text{ for some } H(u, v) := \sum_{j=1}^{m} \sum_{i=1}^{m} \alpha_{ij} u_i v_j.$$

If $F^{(1)}$ and $F^{(2)}$ contain some trapdoor and we assign u or v zero, we can invert the central map. An instantiation of this scheme using the STS trapdoor is depicted in figure 9.

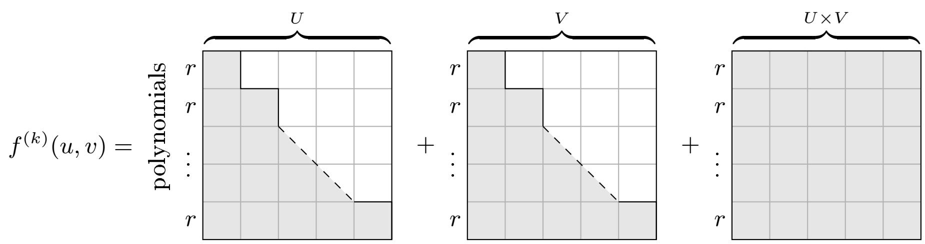


Fig. 9. Secret map $\mathcal{F}$ of Hidden Pair of Bijection using STS trapdoor.

The first observation is that due to the cross-terms in H all the secret matrices $\mathfrak{F}^{(i)}$ have full rank $2m$ and thus rank attacks are not trivially applicable. But there is a smart way in applying rank attacks to the scheme. The weak point is the signing algorithm proposed by Tsujii *et al.*, which first chooses u or v to be zero. They claimed that this would not help an attacker, as his chance to guess the right choice is $\frac{1}{2}$. Well, if we collect $4m-1$ valid signatures $x_1, \ldots, x_{4m-1}$ to arbitrary massages, which are all signed using the same secret S, we can built an efficient distinguisher. We know $X := (x_1^\intercal, \ldots, x_{2m-1}^\intercal)$ is (up to column permutations) of the following form

$$X = \; S \cdot \begin{array}{|c|c|} \hline & 0 \\ \hline 0 & \\ \hline \end{array}$$

The probability of matrix X to have rank $2m-1$ is $(1/2)^{2m-1} 2\binom{2m-1}{m}$ which is sufficiently large—for example choosing $m = 30$ this equals 0.205. We want to thank Masahito Gotaishi, who pointed out that this probability also depends on the field size q and thus the exact formula is

$$\left(\prod_{i=0}^{m-2} (q^m - q^i) \right)^2 \cdot 2(q^m - q^{m-1}) \binom{2m-1}{m} \cdot (2q^m)^{-(2m-1)}.$$

Note that both formulas equal for large q—for example choosing $m = 30$ and $q = 256$ we obtain 0.204. Once we found a collection of signatures $x_1, \ldots, x_{2m-1}$, such that $\operatorname{rank}(X) = 2m - 1$ we obtained an efficient distinguisher. If $X || x_j$ for $j \geq 2m$ still has rank $2m-1$ we add x_j to the set A. If the rank increase by one we add x_j to the set B. Masahito Gotaishi noticed that with probability $1/q$ a signature is added to the set A even if it is of kind B. That means A only contains signatures of the same kind with probability $\left(1 - \frac{1}{2q}\right)^m$. If so, we easily

can recover the part of S which separates the U or the V space, by linear algebra. Once we achieved this we can easily distinguish both kinds of signatures, collect m linear independent signatures of kind B and recover the rest of S. Masahito Gotaishi even proposed a more efficient way to get rid of wrong signatures in A. For every of those signatures he saved the vectors of our distinguisher used in the linear combination of the zero vector. At the end he did a majority vote and only used those signatures assigned to vectors which occurred in most of the linear combinations.
After fixing one of the both sets of variables we obtain a plain STS scheme and can apply the HighRank or the Key Recovery attack from above.

In order to prevent this attack we would have to assign arbitrary values to u respectively v instead of all zeros. This immediately invalidate the trapdoor and makes the scheme unusable. In every step we would have to solve a quadratic underdetermined system of equations without destroying possible solutions through guessing variables.

Conclusions or: Where do we take it from here? In summary, we have introduced a new attack on Enhanced STS that makes use of the heavily structured central map in terms of missing cross-terms. We rate it very unlikely that Enhanced STS or its variants can be repaired while providing an efficient signing algorithm. So the question at hand is if non-linearity could help in any way to improve UOV or Rainbow.

One answer was already given by Kipnis *et al.* in the paper that proposed UOV [KPG03]. One of their possible variants to repair the *balanced* Oil and Vinegar scheme and thus to avoid the attack of Kipnis and Shamir [KS98] was called *Oil, Vinegar and Salt* signature scheme. Here the variables are divided into three sets O, V and S. The central map $\mathcal{F}$ is constructed such that there are no monomials $u_i u_j$ with $u_i \in O$ and $u_j \in V \cup S$. After fixing the vinegar variables we obtain a system linear in the O variables and quadratic in the S variables. The best known way to solve such a system is to brute-force the S variables and then solve the remaining linear system. This way we loose a factor of $q^{|S|}$ in terms of efficiency. As it turned out later, a modified version of the Kipnis and Shamir attack actually *can* be applied to the Oil, Vinegar and Salt scheme. Ironically, the factor we gain in terms of security compared to the original scheme is exactly the factor we loose in terms of efficiency. But as the (positive) effect of non-linearity to the public key size is negligible compared to the (negative) effect to the efficiency of the scheme, the best trade-off is to just skip the salt variables and hence use the original UOV scheme.
STS can be seen as a layer-based version of Oil, Vinegar and Salt. So we can rephrase the question between UOV and UOV+S in this setting. In particular, we have to ask ourselves if the layered structure of STS allows for a better trade-off between efficiency and security than UOV. Unfortunately, we have to leave the final answer as an open question. However, we incline to the negative. To

illustrate this, we want to elaborate some thoughts on this matter. One the one hand, it is not clear even for UOV if the ratio between efficiency and security increases for the layer-based scheme Rainbow. Especially the attack of section 3, which is not applicable to UOV, challenges this hope. On the other hand, the attack of Kipnis and Shamir [KS98] is not practical for layer-based schemes like Rainbow. So the question remains, if and how much security we can gain at all by introducing some non-linearity in each layer. Our intuition is that the loss of efficiency is always greater or equal than the gain of security in these cases and hence of no avail in practice. The reason is that on the one hand the signing algorithm becomes exponential instead of polynomial, as soon as we introduce non-linear parts. In comparison, the attack stays exponential in *both* cases, *i.e.* there is no gap between the legitimate user and the attacker.

The only exception from this rule seem to be Gröbner bases that are used as a trapdoor. Clearly we have to use Vinegar variables in that case, as otherwise MinRank attacks are applicable. But we found no way to fuse this into a working scheme—but got the impression that this is not possible at all. Hence, we leave it as an open problem, how to embed a Gröbner Basis into a scheme using Vinegar variables and to derive a both secure *and* efficient scheme.

5 Cryptanalysis of STS based on Prime Factorization

In section 4 we showed, how to exploit the different rank properties of several variants of the STS signature scheme, to recover the secret keys S and T. At SCC 2012 Tsujii *et al.* proposed yet another variant of STS [TTGF12a,TTGF12b], which relies on the difficulty of prime factorization. At a first glance rank attacks are not applicable, as all the secret polynomials are of full rank. Nevertheless we will now show how to apply a MinRank attack and thus recover the factorization in polynomial time.

The new variant of STS uses the common construction of $\mathcal{MQ}$-schemes, *i.e.* a multivariate quadratic public map $\mathcal{P}$, two secret linear transformations S and T and a secret multivariate quadratic map $\mathcal{F}$ (cf. section 2). In order to invert $\mathcal{F}$ efficiently, we have to embed some special structure. Therefore the authors of [TTGF12b] defined $N = pq$ the product of two secret primes p and q and $\mathcal{F} : \mathbb{Z}_N^n \to \mathbb{Z}_N^n : (x_1, \ldots, x_n)^\intercal \mapsto (f_1(x), \ldots, f_n(x))^\intercal$ as follows.

$$\begin{aligned}
f_1(x_1,\ldots,x_n) &:= p \cdot g_1(x_1) && + q \cdot h_1(x_1,\ldots,x_n)\\
f_2(x_1,\ldots,x_n) &:= p \cdot g_2(x_1,x_2) && + q \cdot h_2(x_2,\ldots,x_n)\\
&\;\;\vdots && \;\;\vdots\\
f_n(x_1,\ldots,x_n) &:= p \cdot g_n(x_1,\ldots,x_n) && + q \cdot h_n(x_n)
\end{aligned}$$

Where g_i and h_i are random quadratic polynomials in the according set of variables. Inverting this map is efficient, because we efficiently find solutions mod q and mod p by solving equation by equation and substituting the solution in the next equation. Using the Chinese Reminder Theorem we easily obtain a valid solution mod N.
If we write $f_i(x) = x^\intercal \mathfrak{F}^{(i)} x$ in its quadratic form, obviously $\mathfrak{F}^{(i)}$ is of full rank and thus rank attacks did not seem to work. Well, thanks to a smart idea of Gottfried Herold (Bochum) we will now show how to find a linear transformation of equations and thus an equivalent key for T such that the structure of the central map $\mathcal{F}$ is given as follows.

$$\begin{aligned}
f_1(x_1,\ldots,x_n) &:= p \cdot g_1(x_1) && + q \cdot h_n(x_n)\\
f_2(x_1,\ldots,x_n) &:= p \cdot g_2(x_1,x_2) && + q \cdot h_2(x_{n-1},x_n)\\
&\;\;\vdots && \;\;\vdots\\
f_n(x_1,\ldots,x_n) &:= p \cdot g_n(x_1,\ldots,x_n) && + q \cdot h_1(x_1,\ldots,x_n)
\end{aligned}$$

Obviously $I \circ \mathcal{F}$ mod q, with I the identity mapping, provides the required form of $\mathcal{F}$ mod q. Furthermore $\Pi \circ \mathcal{F}$ mod p provides the required form of $\mathcal{F}$ mod p, for Π a permutation of the following form.

$$\Pi = \begin{bmatrix} & & & 1\\ & & 1 & \\ & \iddots & & \\ 1 & & & \end{bmatrix}$$

Now we easily obtain the searched transformation X by solving

$$X \equiv I \mod q$$
$$X \equiv \Pi \mod p$$

using the Chinese Reminder Theorem. In fact, the solution is given through $X = Ipp_q^{-1} + \Pi q q_p^{-1} \mod N$, with q_p^{-1} the multiplicative inverse of q mod p and vice versa.
With $X \cdot T$ there exist a linear transformation of equations, such that one central polynomial is of rank 2. Recovering this polynomial by a MinRank attack using Gröbner Bases has complexity $\binom{n+3}{3}^2$ (cf. [FDS11,FdVP08]). Note that we never have to determine X to run our attack, it is sufficient that $X \cdot T$ exists. To finally obtain the factorization of N, we first have to partly recover the transformation of variables S. We again use equivalent keys to achieve this goal. More precisely, to preserve the zero coefficients in $p \cdot g_1(x_1) + q \cdot h_n(x_n)$ we are allowed to map all variables, except x_1 and x_n to sums of all variables. Further we are allowed to map x_1 to multiples of x_1 and x_n to multiples of x_n. The according equivalent key has the following form.

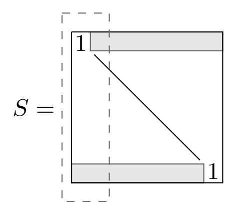


Fig. 10. Equivalent key S with ones at the diagonal, arbitrary values at the gray parts and zeros at the white parts.

In order to recover the first 3 columns of S, *i.e.* the dashed area in figure 10 or more precisely $s_{12}, s_{13}, s_{n1}, s_{n2}$ and s_{n3}, we have to solve the quadratic equations obtained by the zero coefficients of $x_1x_2, x_1x_3, x_2^2, x_2x_3$ and x_3^2. Solving this system of 5 quadratic equations in 5 variables with some Gröbner Basis algorithm like F_4 has complexity $\binom{5+6}{6}^2 \approx 2^{18}$ as the degree of regularity will be 6 (cf. [BFS04]). Note that $\mathbb{Z}_N$ is not a field and thus the Gröbner Basis computation might fail. But as soon as an error occur, we immediately learn the factorization of N.
After learning the first column of S and applying the transformation, the coefficient γ_{11} of x_1^2 in the public polynomial is a pure multiple of p. This is due to the fact that x_n is no longer mapped to x_1. This way we can easily factor N by calculating $\gcd(\gamma_{11}, N)$.

For a reasonable choice of parameters, *i.e.* $n = 30$, our attack has complexity at most $\binom{33}{3}^2 \approx 2^{25}$.

6 Cryptanalysis of Enhanced TTS

Enhanced TTS was proposed by Yang and Chen in 2005 [YC05]. The overall idea of the scheme was to use several layers of UOV trapdoors and to make them as sparse as possible. In contrast to UOV this would prevent the Kipnis and Shamir attack [KS98] without increasing the number of vinegar variables. In fact, while we have a signature extension of factor 3 for UOV, enTTS improves this figure to 1.3. As enTTS was designed for high speed implementation it uses as few monomials as possible.
There are two different scalable central maps given in [YC05], one is called *even* sequence and the other *odd* sequence. The following equations show the even sequence.

$$
\begin{aligned}
f^{(i)} &= u_i + \sum_{j=1}^{2\ell-5} \gamma_{ij} u_j u_{2\ell-4+(i+j+1 \bmod 2\ell-2)} \qquad \text{for } 2\ell-4 \le i \le 4\ell-7,\\
f^{(i)} &= u_i + \sum_{j=1}^{\ell-4} \gamma_{ij} u_{i+j-(4\ell-6)} u_{i-j-2\ell-1} + \sum_{j=\ell-3}^{2\ell-5} \gamma_{ij} u_{i+j-3\ell+5} u_{i-j+\ell-4}\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for } 4\ell-6 \le i \le 4\ell-3,\\
f^{(i)} &= u_i + \gamma_{i0} u_{i-2\ell+2} u_{i-2\ell-2} + \sum_{j=i+1}^{6\ell-5} \gamma_{i,j-(4\ell-3)} u_{4\ell-3+i-j} u_j\\
&\quad + \gamma_{i,i-4\ell+3} u_0 u_i + \sum_{j=4\ell-2}^{i-1} \gamma_{i,j-(4\ell-3)} u_{2(i-j)-(i \bmod 2)} u_j + \gamma_{i,i-4\ell+2} u_0 u_i\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for } 4\ell-2 \le i \le 6\ell-5.
\end{aligned}
$$

The number of equations and variables is $m = 4\ell$ and $n = 6\ell-4$, respectively, for some parameter ℓ. The first observation is that the number of equations obtained by (2) is very large, as only $2\ell-3$ monomials per equation are non-zero. The second observation is that the linear terms provide an enormous amount of new equations, as their coefficients are not chosen at random but fixed. Considering only the linear parts of the public polynomials $p^{(j)}$ we obtain the following equation analogously to (1)

$$
e_{i+2\ell-5} = \widetilde{S}\left(\sum_{j=1}^{m} \widetilde{t}_{ij}(\gamma_1^{(j)}, \ldots, \gamma_n^{(j)})^\intercal\right) \text{ for } 1 \le i \le m, \tag{4}
$$

where e_i denote the all-zero vector with a single 1 in the i-th entry and $\gamma_i^{(j)}$ is the coefficient of x_i in $p^{(j)}$. We obtain a total amount of $4\ell(6\ell-4)$ bihomogeneous equations in the $(4\ell)^2$ variables of $\widetilde{T}$ and in the $(6\ell-4)^2$ variables of $\widetilde{S}$. Note that the number of variables would increase if we choose S to be affine instead of linear. But despite of this large amount of equations a theoretical complexity analysis of solving those equations provide infeasible large results, due to the

large amount of variables. Note that in practice the solving algorithm may seriously benefit of the equations internal structure. We leave it as an open problem to implement this attack and run experiments to determine the real complexity of attacking enTTS this way.
In the sequel we once again focus on reducing the number of variables. Note that most of the equations (4) vanish as soon as we use equivalent keys. This is also true for a large amount of zero-coefficients in the quadratic part. Thus we generalize the scheme by adding *more* monomials. In particular, we adapt the definition of enTTS as follows: As soon as a monomial $x_i x_j$ with $x_i \in U$ and $x_j \in V$ occurs in the original enTTS polynomial $f^{(k)}$, we just assume that all monomials $x_i x_j$ with $x_i \in U$ and $x_j \in V$ occur as well. This way we easily see that enTTS is a very special case of the Rainbow signature scheme, neglecting the linear parts. We chose the parameter set $(n, m) = (32, 24)$ and thus $\ell = 6$ given in [YC05], as this provides a security level of 2^{88}. See figure 11 for an illustration.

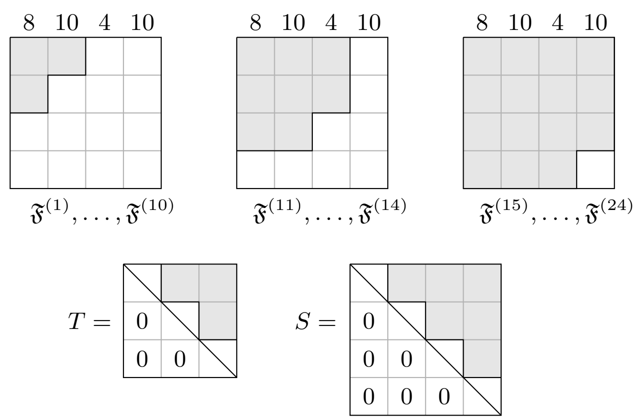


Fig. 11. Secret map $\mathcal{F}$ of TTS $(32, 24)$ and equivalent keys T and S.

The attack is similar to the one described in section 4. Suppose we just want do preserve zero coefficients of $x_{32} x_i$ in polynomial $u^\intercal \mathfrak{F}^{(14)} u$. This leads to the good keys given in figure 12 and thus to 31 bihomogeneous equations in 10 variables t_{14i} with $i = 15, \ldots, 24$ and 22 variables s_{j32} with $j = 1, \ldots, 22$. Analogous to section 4 we first have to guess one variable t_{ij}. Solving the remaining system of 31 bihomogeneous equations in 31 variables has complexity $2^8 \binom{31+10}{10}^2 \approx 2^{68}$ (cf. [FDS11]).

Remark 1. Using the good key T' of figure 12 gives arbitrary values for the first $4\ell - 2$ entries in e_i of (4). Only the last $2\ell - 2$ entries are invariant under the transformation Ω. But due to the good key S' these entries become arbitrary as well, except the last one. Thus we obtain one more bihomogeneous equation from

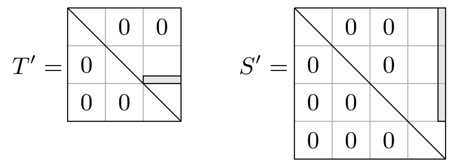


Fig. 12. Good Keys T' and S' for enTTS $(32, 24)$.

(4) using good keys. Now we can apply [FDS11] without guessing one variable beforehand and obtain an overall complexity of $\binom{32+11}{11}^2 \approx 2^{65}$.

But due to the special structure of enTTS we can do even better. Applying the transformation of variables Ω analogous to lemma 1, we see that the monomial $u_{32}u_{32}$ do not occur in *any* of the secret polynomials. This way we additionally obtain 23 quadratic equations in s_{ij}. The complexity of solving a generic system of $23 + 32$ quadratic and 1 cubic equation in 32 variables is $2^{47.7}$. Note that this complexity is just an upper bound as we assumed generic equations and thus did not use the special bihomogeneous structure.

7 Cryptanalysis of MFE Based on Diophantine Equations

The MFE encryption scheme was published at CT-RSA 2006 [WYHL06] and broken at PKC 2007 by Ding et Al. [DHN+07]. The variant using Diophantine equations was published at Designs, Codes and Cryptography in 2011 [GH11]. Clearly the security goals of MFE are out of date, as even a direct attack on the public key using F_4 or XL is efficient due to the small number of equations and variables. Therefore we will not give another attack on MFE, but concentrate on the more secure variant proposed in [GH11]. Note that our attack also applies to the original MFE scheme and very likely would be as efficient as the high order linearization attack of [DHN+07].

MFE Encryption Scheme. We briefly describe the main idea of MFE. For a detailed description please refer to [WYHL06].

The central map $\mathcal{F} : \mathbb{F}_{2^k}^{12} \to \mathbb{F}_{2^k}^{15} : (x_1, \dots, x_{12}) \mapsto (y_1, \dots, y_{15})$ is defined by

$$
\begin{array}{ll}
y_1 = x_1 + \phi(x_1) + \psi_1 & \\
y_2 = x_2 + \phi(x_1, x_2) + \psi_2 & \\
y_3 = x_3 + \phi(x_1, x_2, x_3) + \psi_3 & \\
y_4 = x_1x_5 + x_2x_7 & y_{10} = x_3x_9 + x_4x_{11} \\
y_5 = x_1x_6 + x_2x_8 & y_{11} = x_3x_{10} + x_4x_{12} \\
y_6 = x_3x_5 + x_4x_7 & y_{12} = x_5x_9 + x_7x_{11} \\
y_7 = x_3x_6 + x_4x_8 & y_{13} = x_5x_{10} + x_7x_{12} \\
y_8 = x_1x_9 + x_2x_{11} & y_{14} = x_6x_9 + x_8x_{11} \\
y_9 = x_1x_{10} + x_2x_{12} & y_{15} = x_6x_{10} + x_8x_{12}
\end{array}
$$

where ϕ_1, ϕ_2 and ϕ_2 are random quadratic polynomials and ψ_1, ψ_2 and ψ_3 are polynomials in $y_4, \dots, y_{15}$ obtained by a special determinant relation. On a high level view the central map is a mix of two different principles. First y_1, y_2 and y_3 are composed of a stepwise triangular structure (cf. STS in section 4) and a masking ψ_1, ψ_2, ψ_3 which hides this structure. To decrypt, we can easily calculate the values of ψ_i, as they only depend on $y_4, \dots, y_{15}$ and unmask y_1, y_2 and y_3. Consecutively solving these equations yields x_1, x_2 and x_3. Second $y_4, \dots, y_{15}$ are partitioned in 3 blocks of oil and vinegar structure (cf. UOV section 2), *i.e.* plugging in x_1, x_2 and x_3 provide linear equations and so on. The public map $\mathcal{P}$ is obtained as usual by $\mathcal{P} = T \circ \mathcal{F} \circ S$.

MFE Encryption Scheme Based on Diophantine Equations. The variant of [GH11] generalize the idea of MFE to another class of Diophantine equations. In particular they use a Diophantine equation of the form

$$\psi_1\psi_2 = f_1f_2 + f_3f_4 + f_5f_6 + f_7f_8 + f_9f_{10}$$

where $f_1, \dots, f_{10}$ are quadratic polynomials with oil and vinegar structure and ψ_1, ψ_2 are the polynomials used for masking later on. To find an instantiation of ψ_i and f_i the authors used the polynomial ring

$$R = \mathbb{F}_{2^k}[z_1, z_2, z_3, z_4, w_1, w_2, w_3, w_4, u_1, u_2, u_3, u_4, v_1, v_2, v_3, v_4] \tag{5}$$

and Plücker coordinates (cf. definition 3), which are know to satisfy the following identity

$$\begin{aligned} 0 = &(p^{12}_{zw} + p^{12}_{uv})p^{34}(z,w,u,v) + (p^{13}_{zw} + p^{13}_{uv})p^{24}(z,w,u,v) + \\ &(p^{14}_{zw} + p^{14}_{uv})p^{23}(z,w,u,v) + (p^{23}_{zw} + p^{23}_{uv})p^{14}(z,w,u,v) + \\ &(p^{24}_{zw} + p^{24}_{uv})p^{13}(z,w,u,v) + (p^{34}_{zw} + p^{34}_{uv})p^{12}(z,w,u,v). \end{aligned} \tag{6}$$

Definition 3 (Plücker coordinates). *Given the polynomial ring defined in (5), the Plücker coordinates are defined by*

$$\begin{aligned} p^{ij}_{zw} &:= z_i w_j - z_j w_i = z_i y_j + w_j y_i, \\ p^{ij}(z,w,u,v) &:= p^{ij}_{zu} + p^{ij}_{wu} + p^{ij}_{wv}. \end{aligned}$$

To transform the 5 last terms of the sum (6) in oil and vinegar form, the authors used the isomorphism

$$\begin{aligned} \rho : R &\to \mathbb{F}_{2^k}[x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8] \\ &: (z_1, z_2, z_3, z_4, w_1, w_2, w_3, w_4, u_1, u_2, u_3, u_4, v_1, v_2, v_3, v_4) \mapsto \\ &\quad (x_1, x_3, y_1 + y_5, y_3 + y_7, x_4, x_2, y_5, y_7, x_5, x_7, y_4, y_2, x_8, x_6, y_8, y_6) \end{aligned}$$

Note that there were two typos in the definition of ρ in [GH11] (confirmed by [Gao12]).

The central map $\mathcal{F} : \mathbb{F}^{56}_{2^k} \to \mathbb{F}^{74}_{2^k} : (x_1, \ldots, x_{24}, y_1, \ldots, y_{32}) \mapsto (z_1, \ldots, z_{74})$ is defined by

$$\begin{array}{lll}
z_1 &= x_1 + \phi_1(x_1) \qquad\qquad + \psi_{1,1}(x_1, \ldots, x_8) & \\
z_2 &= x_2 + \phi_2(x_1, x_2) \qquad + \psi_{1,2}(y_1, \ldots, y_8) & \\
z_3 &= x_3 + \phi_3(x_1, \ldots, x_3) + \psi_{2,2}(y_9, \ldots, y_{16}) & \\
z_4 &= x_4 + \phi_4(x_1, \ldots, x_4) + \psi_{3,2}(y_{17}, \ldots, y_{24}) & \\
z_5 &= x_5 + \phi_5(x_1, \ldots, x_5) + \psi_{1,1}(x_9, \ldots, x_{16}) & \\
z_6 &= x_6 + \phi_6(x_1, \ldots, x_6) + \psi_{1,1}(x_{17}, \ldots, x_{24}) & \\
z_7 &= x_7 + \phi_7(x_1, \ldots, x_7) + \psi_{4,2}(y_{25}, \ldots, y_{32}) & \\
z_{7+i} &= f_{1,i}(x_1, \ldots, x_8, y_1, \ldots, y_8) & 1 \le i \le 10 \\
z_{17+i} &= f_{2,i}(x_1, \ldots, x_8, y_9, \ldots, y_{16}) & 1 \le i \le 10 \\
z_{27+i} &= f_{2,i}(y_1, \ldots, y_8, y_9, \ldots, y_{16}) & 1 \le i \le 8 \\
z_{36} &= f_{2,10}(y_1, \ldots, y_8, y_9, \ldots, y_{16}) & \\
z_{36+i} &= f_{3,i}(x_1, \ldots, x_8, y_{17}, \ldots, y_{24}) & 1 \le i \le 10 \\
z_{46+i} &= f_{2,i}(x_9, \ldots, x_{16}, y_9, \ldots, y_{16}) & 1 \le i \le 8 \\
z_{55} &= f_{2,10}(x_9, \ldots, x_{16}, y_9, \ldots, y_{16}) & \\
z_{56+i} &= f_{3,i}(x_{17}, \ldots, x_{24}, y_{17}, \ldots, y_{24}) & 1 \le i \le 8 \\
z_{64} &= f_{3,10}(x_{17}, \ldots, x_{24}, y_{17}, \ldots, y_{24}) & 1 \le i \le 8 \\
z_{56+i} &= f_{3,i}(x_{17}, \ldots, x_{24}, y_{17}, \ldots, y_{24}) & \\
z_{64+i} &= f_{4,i}(x_9, \ldots, x_{16}, y_{25}, \ldots, y_{32}) & 1 \le i \le 10
\end{array}$$

where $\phi_1, \ldots, \phi_7$ are random quadratic polynomials and $f_{i,j} := f_{1,j}$ for $i = 2, 3, 4$ and $j = 1, 3, 5, 7, 9$. Further we define

$$
\begin{array}{lll}
\psi_{2,2} := \rho(p^{34}(z,w,v,u)) & \psi_{3,2} := \rho(p^{34}(w,z,u,v)) & \psi_{4,2} := \rho(p^{34}(w,z,v,u)) \\
f_{2,2} := \rho(p^{24}(z,w,v,u)) & f_{3,2} := \rho(p^{24}(w,z,u,v)) & f_{4,2} := \rho(p^{24}(w,z,v,u)) \\
f_{2,4} := \rho(p^{23}(z,w,v,u)) & f_{3,4} := \rho(p^{23}(w,z,u,v)) & f_{4,4} := \rho(p^{23}(w,z,v,u)) \\
f_{2,6} := \rho(p^{14}(z,w,v,u)) & f_{3,6} := \rho(p^{14}(w,z,u,v)) & f_{4,6} := \rho(p^{14}(w,z,v,u)) \\
f_{2,8} := \rho(p^{13}(z,w,v,u)) & f_{3,8} := \rho(p^{13}(w,z,u,v)) & f_{4,8} := \rho(p^{13}(w,z,v,u)) \\
f_{2,10} := \rho(p^{12}(z,w,v,u)) & f_{3,10} := \rho(p^{12}(w,z,u,v)) & f_{4,10} := \rho(p^{12}(w,z,v,u))
\end{array}
$$

To use the structure of $\mathcal{F}$ for an algebraic key recovery attack, *e.g.* missing cross-terms, we need to look at the equations explicitly:

$$
\begin{aligned}
z_1 &= x_1 + \phi_1(x_1) + x_1x_2 + x_3x_4 + x_5x_6 + x_7x_8 \\
z_2 &= x_2 + \phi_2(x_1, x_2) + y_1y_2 + y_3y_4 + y_5y_6 + y_7y_8 \\
z_3 &= x_3 + \phi_3(x_1, \ldots, x_3) + y_9y_{14} + y_{10}y_{13} + y_{11}y_{16} + y_{12}y_{15} \\
z_4 &= x_4 + \phi_4(x_1, \ldots, x_4) + y_{17}y_{18} + y_{17}y_{22} + y_{19}y_{20} + y_{19}y_{24} + y_{21}y_{22} + y_{23}y_{24} \\
z_5 &= x_5 + \phi_5(x_1, \ldots, x_5) + x_9x_{10} + x_{11}x_{12} + x_{13}x_{14} + x_{15}x_{16} \\
z_6 &= x_6 + \phi_6(x_1, \ldots, x_6) + x_{17}x_{18} + x_{19}x_{20} + x_{21}x_{22} + x_{23}x_{24} \\
z_7 &= x_7 + \phi_7(x_1, \ldots, x_7) + y_{25}y_{26} + y_{25}y_{30} + y_{26}y_{29} + y_{27}y_{28} + y_{27}y_{32} + y_{28}y_{31} \\
\hline
z_8 &= (x_1 + x_4)y_5 + x_4y_1 + x_5y_8 + x_8y_4 \\
z_9 &= (x_2 + x_3)y_2 + x_2y_6 + x_6y_7 + x_7y_3 \\
z_{10} &= (x_1 + x_4)y_7 + x_4y_3 + x_5y_6 + x_8y_2 \\
z_{11} &= (x_2 + x_3)y_4 + x_2y_8 + x_6y_5 + x_7y_1 \\
z_{12} &= (x_2 + x_3)y_5 + x_2y_1 + x_6y_4 + x_7y_8 \\
z_{13} &= (x_1 + x_4)y_2 + x_4y_6 + x_5y_3 + x_8y_7 \\
z_{14} &= (x_2 + x_3)y_7 + x_2y_3 + x_6y_2 + x_7y_6 \\
z_{15} &= (x_1 + x_4)y_4 + x_4y_8 + x_5y_1 + x_8y_5 \\
z_{16} &= y_1y_7 + y_2y_8 + y_3y_5 + y_4y_6 \\
z_{17} &= (x_1 + x_4)x_7 + (x_2 + x_3)x_5 + x_2x_8 + x_4x_6 \\
\hline
z_{18} &= (x_1 + x_4)y_{13} + x_4y_9 + x_5y_{16} + x_8y_{12} \\
z_{19} &= (x_2 + x_3)y_{14} + x_2y_{10} + x_6y_{11} + x_7y_{15} \\
z_{20} &= (x_1 + x_4)y_{15} + x_4y_{11} + x_5y_{14} + x_8y_{10} \\
z_{21} &= (x_2 + x_3)y_{16} + x_2y_{12} + x_6y_9 + x_7y_{13} \\
z_{22} &= (x_2 + x_3)y_{13} + x_2y_9 + x_6y_{12} + x_7y_{16} \\
z_{23} &= (x_1 + x_4)y_{14} + x_4y_{10} + x_5y_{15} + x_8y_{11} \\
z_{24} &= (x_2 + x_3)y_{15} + x_2y_{11} + x_6y_{10} + x_7y_{14} \\
z_{25} &= (x_1 + x_4)y_{16} + x_4y_{12} + x_5y_{13} + x_8y_9 \\
z_{26} &= y_9y_{15} + y_{10}y_{16} + y_{11}y_{13} + y_{12}y_{14} \\
z_{27} &= (x_1 + x_4)x_6 + x_2x_5 + (x_2 + x_3)x_8 + x_4x_7 \\
\hline
z_{28} &= (y_1 + y_4)y_{13} + y_4y_9 + y_5y_{16} + y_8y_{12}
\end{aligned}
$$

$$z_{29} = (y_2 + y_3)y_{14} + y_2y_{10} + y_6y_{11} + y_7y_{15}$$
$$z_{30} = (y_1 + y_4)y_{15} + y_4y_{11} + y_5y_{14} + y_8y_{10}$$
$$z_{31} = (y_2 + y_3)y_{16} + y_2y_{12} + y_6y_9 + y_7y_{13}$$
$$z_{32} = (y_2 + y_3)y_{13} + y_2y_9 + y_6y_{12} + y_7y_{16}$$
$$z_{33} = (y_1 + y_4)y_{14} + y_4y_{10} + y_5y_{15} + y_8y_{11}$$
$$z_{34} = (y_2 + y_3)y_{15} + y_2y_{11} + y_6y_{10} + y_7y_{14}$$
$$z_{35} = (y_1 + y_4)y_{16} + y_4y_{12} + y_5y_{13} + y_8y_9$$

$$z_{36} = (y_1 + y_4)y_6 + y_2y_5 + (y_2 + y_3)y_8 + y_4y_7$$

$$z_{37} = (x_1 + x_4)y_{21} + x_4y_{17} + x_5y_{24} + x_8y_{20}$$
$$z_{38} = (x_2 + x_3)y_{18} + x_3y_{22} + (x_6 + x_7)y_{19} + x_6y_{23}$$
$$z_{39} = (x_1 + x_4)y_{23} + x_4y_{19} + x_5y_{22} + x_8y_{18}$$
$$z_{40} = (x_2 + x_3)y_{20} + x_3y_{24} + (x_6 + x_7)y_{17} + x_6y_{21}$$
$$z_{41} = (x_2 + x_3)y_{21} + x_2y_{17} + x_6y_{20} + x_7y_{24}$$
$$z_{42} = (x_1 + x_4)y_{18} + x_1y_{22} + (x_5 + x_8)y_{19} + x_8y_{23}$$
$$z_{43} = (x_2 + x_3)y_{23} + x_2y_{19} + x_6y_{18} + x_7y_{22}$$
$$z_{44} = (x_1 + x_4)y_{20} + x_1y_{24} + (x_5 + x_8)y_{17} + x_8y_{21}$$
$$z_{45} = y_{17}y_{23} + y_{18}y_{24} + y_{19}y_{21} + y_{20}y_{22}$$
$$z_{46} = (x_1 + x_4)x_7 + x_1x_6 + (x_2 + x_3)x_5 + x_3x_8$$

$$z_{47} = (x_9 + x_{12})y_{13} + x_{12}y_9 + x_{13}y_{16} + x_{16}y_{12}$$
$$z_{48} = (x_{10} + x_{11})y_{14} + x_{10}y_{10} + x_{14}y_{11} + x_{15}y_{15}$$
$$z_{49} = (x_9 + x_{12})y_{15} + x_{12}y_{11} + x_{13}y_{14} + x_{16}y_{10}$$
$$z_{50} = (x_{10} + x_{11})y_{16} + x_{10}y_{12} + x_{14}y_9 + x_{15}y_{13}$$
$$z_{51} = (x_{10} + x_{11})y_{13} + x_{10}y_9 + x_{14}y_{12} + x_{15}y_{16}$$
$$z_{52} = (x_9 + x_{12})y_{14} + x_{12}y_{10} + x_{13}y_{15} + x_{16}y_{11}$$
$$z_{53} = (x_{10} + x_{11})y_{15} + x_{10}y_{11} + x_{14}y_{10} + x_{15}y_{14}$$
$$z_{54} = (x_9 + x_{12})y_{16} + x_{12}y_{12} + x_{13}y_{13} + x_{16}y_9$$

$$z_{55} = (x_9 + x_{12})x_{14} + x_{10}x_{13} + (x_{10} + x_{11})x_{16} + x_{12}x_{15}$$

$$z_{56} = (x_{17} + x_{20})y_{21} + x_{20}y_{17} + x_{21}y_{24} + x_{24}y_{20}$$
$$z_{57} = (x_{18} + x_{19})y_{18} + x_{19}y_{22} + (x_{22} + x_{23})y_{19} + x_{22}y_{23}$$
$$z_{58} = (x_{17} + x_{20})y_{23} + x_{20}y_{19} + x_{21}y_{22} + x_{24}y_{18}$$
$$z_{59} = (x_{18} + x_{19})y_{20} + x_{19}y_{24} + (x_{22} + x_{23})y_{17} + x_{22}y_{21}$$
$$z_{60} = (x_{18} + x_{19})y_{21} + x_{18}y_{17} + x_{22}y_{20} + x_{23}y_{24}$$
$$z_{61} = (x_{17} + x_{20})y_{18} + x_{17}y_{22} + (x_{21} + x_{24})y_{19} + x_{24}y_{23}$$
$$z_{62} = (x_{18} + x_{19})y_{23} + x_{18}y_{19} + x_{22}y_{18} + x_{23}y_{22}$$
$$z_{63} = (x_{17} + x_{20})y_{20} + x_{17}y_{24} + (x_{21} + x_{24})y_{17} + x_{24}y_{21}$$

$$z_{64} = x_{17}x_{22} + (x_{17} + x_{20})x_{23} + (x_{18} + x_{19})x_{21} + x_{19}x_{24}$$

$$z_{65} = (x_9 + x_{12})y_{29} + x_{12}y_{25} + x_{13}y_{32} + x_{16}y_{28}$$

$$
\begin{aligned}
z_{66} &= (x_{10} + x_{11})y_{30} + x_{11}y_{26} + (x_{14} + x_{15})y_{27} + x_{15}y_{31} \\
z_{67} &= (x_9 + x_{12})y_{31} + x_{12}y_{27} + x_{13}y_{30} + x_{16}y_{26} \\
z_{68} &= (x_{10} + x_{11})y_{32} + x_{11}y_{28} + (x_{14} + x_{15})y_{25} + x_{15}y_{29} \\
z_{69} &= (x_{10} + x_{11})y_{29} + x_{10}y_{25} + x_{14}y_{28} + x_{15}y_{32} \\
z_{70} &= (x_9 + x_{12})y_{30} + x_9y_{26} + (x_{13} + x_{16})y_{27} + x_{13}y_{31} \\
z_{71} &= (x_{10} + x_{11})y_{31} + x_{10}y_{27} + x_{14}y_{26} + x_{15}y_{30} \\
z_{72} &= (x_9 + x_{12})y_{32} + x_9y_{28} + (x_{13} + x_{16})y_{25} + x_{13}y_{29} \\
z_{73} &= y_{25}y_{31} + y_{26}y_{32} + y_{27}y_{29} + y_{28}y_{30} \\
z_{74} &= (x_9 + x_{12})x_{14} + x_9x_{15} + (x_{10} + x_{11})x_{16} + x_{11}x_{13}
\end{aligned}
$$

Let $\mathfrak{Z}^{(i)}$ be the matrix describing the quadratic form of the central polynomial z_i, *i.e.* $z_i(x) = x^\intercal \mathfrak{Z}^{(i)} x$ with $x := (x_1, \ldots, x_{24}, y_1, \ldots, y_{32})$. Due to $\mathcal{P} = T \circ \mathcal{F} \circ S$, we know that every public polynomial $p^{(i)}$ is of the form

$$
\mathfrak{P}^{(i)} = S^\intercal \underbrace{\left(\sum_{j=1}^{74} t_{ij} \mathfrak{Z}^{(j)} \right)}_{=:\widetilde{\mathfrak{Z}}} S.
$$

For arbitrary chosen T the matrix $\widetilde{\mathfrak{Z}}$ is of form given in figure 13. All the white values denote coefficients that are systematical zero and thus can be used to recover S without recovering T at the same time.

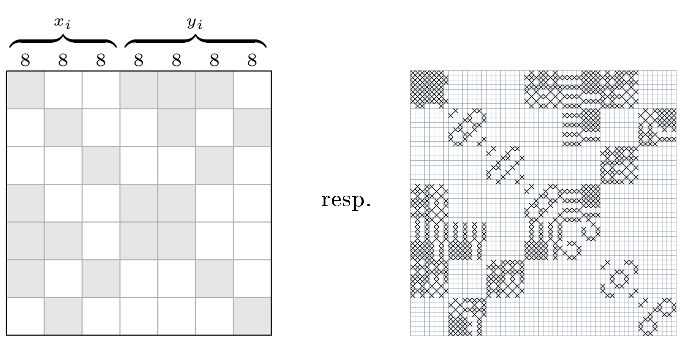


Fig. 13. Matrix $\widetilde{\mathfrak{Z}}$, where gray parts denote arbitrary values of the corresponding coefficients and white parts denote zeros, respectively. The left matrix is a generalized version of the detailed right matrix.

At this stage an algebraic key recovery attack fails due to the large number of variables s_{ij}. To be precise, we derive $74 \cdot 15 \cdot 8^2 = 71040$ quadratic equations in $6 \cdot 7 \cdot 8^2 = 2688$ variables s_{ij}. The complexity of solving a generic system of this size using F_4 or XL would be 2^{320} and thus infeasible. To reduce this complexity we have to use the special structure of the central polynomials z_i and find good keys minimizing the number of variables while maximizing the preserved structure of the central map. The first observation is that variables $y_{25}, y_{28}, y_{29}, y_{32}$ only occur in the six polynomials $z_7, z_{65}, z_{68}, z_{69}, z_{72}, z_{73}$. Thus, with high probability, there exist a linear combination

$$\mathfrak{P}^{(7)} + \sum_{i=1}^{6} \widetilde{t}_i \mathfrak{P}^{(i)} = S^{\intercal} \left(\sum_{j\in I} t_j \mathfrak{Z}^{(j)} \right) S$$

with $I := \{1, \ldots, 74\} \backslash \{7, 65, 68, 69, 72, 73\}$. Now we can use a linear transformation Ω that maps every variable except y_{32} to every of the other variables. We obtain the good key S' shown in figure 14. Furthermore Ω preserves all zero coefficients of monomials $x_i y_{32}$ and $y_i y_{32}$.

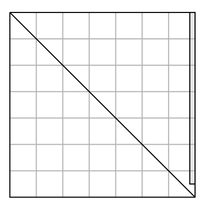
$S' =$ 

Fig. 14. Good Key S' for MFE based on Diophantine equations, where white parts denote zeros, gray parts denote arbitrary values and ones at the diagonal.

We end up with 55 bihomogeneous quadratic (from $x_i y_{32}$ and $y_i y_{32}$ with $i \neq 32$) and one cubic equation (from $y_{32} y_{32}$) in 52 variables s_{ij} and 6 variables t_i. Unfortunately the number of bihomogeneous equations is less than the number of variables and thus we cannot directly apply the results of [FDS11]. But after guessing 3 variables t_i we can use their formula and obtain an attack complexity of $q^3 \binom{59+4}{4}^2 \approx 2^{86}$. Well this already beats the claimed security of 2^{113}, but we can do even better.

A first simple optimization is to use 4 instead of 1 rows of T and thus obtain 4 central polynomials with the structure described above. We end up with $4 \cdot 55 = 220$ bihomogeneous quadratic and 4 cubic equations in $52 + 4 \cdot 6 = 76$ variables. As it is an oben problem to determine the complexity of solving such block-wise bihomogeneous equations we only can assume generic equations and thus obtain a very bad upper bound of 2^{71} to solve the system using F_4.

But we can do even better by ignoring the transformation T and just using the structure given in figure 13.

Let $J := \{x_{14}, x_{16}, x_{21}, x_{23}, y_6, y_{13}, y_{14}, y_{15}, y_{16}, y_{18}, y_{20}, y_{21}, y_{23}, y_{29}, y_{30}, y_{31}, y_{32}\}$ and $K := \{x_1, \ldots, x_{24}, y_1, \ldots, y_{36}\} \backslash J$. The crucial observation is that non of the central polynomials z_i contains monomials $J \times J$. In order to preserve the zero coefficient of y_{32}^2 we are thus allowed to map every variable to variables of J and every variable except y_{32} to variables of K. Let us label columns and rows of S by $(x_1, \ldots, x_{24}, y_1, \ldots, y_{32})$, *i.e.* $s_{x_2,y_{32}}$ is the element of S in the 2nd row and 56th column. The good key S, which only preserves the zero coefficients of y_{32}^2 only consists of $56 - 17 = 39$ variables $s_{i,y_{32}}$ for $i \in K$ in the last column. We omit a formal proof, as it is the same like for lemma 1 and 2. In total we obtain 74 quadratic equations (the coefficient of y_{32}^2 has to be zero in *every* public polynomial independently of T) in 39 variables $s_{i,y_{32}}$. Solving this system has complexity 2^{56}.

Now we can repeat this progress for y_{31}^2 and obtain $s_{i,y_{31}}$ for $i \in K$ with complexity 2^{56} again. At this point we can determine $s_{i,y_{32}}$ for $i \in J$ using that the coefficients of $y_{31}y_{32}$ has to be zero. Solving those 74 equations in 17 variables has complexity 2^{20}. Next we obtain $3 \cdot 74$ equations through $y_{30}^2, y_{30}y_{31}, y_{30}y_{32}$ and can determine variables $s_{i,y_{30}}$ for $i \in K$ and $s_{i,y_{31}}$ for $i \in J$ at once. Solving this system of 222 equations in 56 variables has complexity 2^{45}. Note that from now on more and more equations become available in every step, until we obtained all columns of S labeled by J. To determine the remaining columns of S, we use that non of the elements of K is connected to more than 9 out of 17 elements of J in all the central equations z_i. Thus we obtain at least $8 \cdot 74$ equations to determine the 56 variables of column $j \in K$ of S. This has complexity 2^{30}. Note that if we proceed sequential we can also use zero coefficients of $K \times K$ and thus obtain much more equations. As soon as all the columns of S labeled with all the monomials occurring in z_i are determined we obtain the i-th row of the secret key T through linear equations.
To summarize, a key recovery attack on MFE based on Diophantine equations has complexity at least $2 \cdot 2^{56} = 2^{57}$.

8 Cryptanalysis of the MQQ Encryption Scheme

The original variant of the multivariate quadratic quasigroup (MQQ) scheme was proposed by Gligoroski *et al.* in 2008 [GMK08]. The underlying idea was to use the bijective operation of a quasigroup, *e.g.* the left parastrophe, which can be described through a quadratic map over the underlying field, as trapdoor to build a $\mathcal{MQ}$-scheme. Unfortunately this trapdoor provided a lot of structure such that the MQQ scheme was broken by a direct attack on the public key. Faugère *et al.* showed in [FØPG10] that the degree of regularity, and thus the complexity of a direct attack, can be bounded from above by a small constant. In [FGJ$^+$11] Faugère *et al.* proposed a signature scheme, called MQQ-Sig, which is based on the same idea. They made heavily use of the minus modifier, known from HFE-, to built a scheme which is considered secure up to now. At SCC 2012 Gligoroski and Samardjiska [GS12] proposed an enhanced variant of the MQQ encryption scheme, called MQQ-Enc.
Up to now, almost all unbroken $\mathcal{MQ}$-schemes are signature schemes (cf. UOV and Rainbow in section 2 and 3). Moreover at Crypto 2011 Sakumoto *et al.* [SSH11] proposed the first $\mathcal{MQ}$ identification scheme that is proven to be secure. Via the Fiat-Shamir construction this provides the first $\mathcal{MQ}$ signature scheme that is reasonable efficient and secure in the random oracle model.
On the other hand, constructing efficient and secure $\mathcal{MQ}$ encryption schemes failed (cf. section 7 or [Her12,KS99,DH11]). We will now show that this rule also applies to MQQ-Enc: Moreover our attack also applies to MQQ-Sig and thus breaks both schemes very efficiently.

MQQ Encryption Scheme. Let $\mathcal{P} : \mathbb{F}_{p^k}^n \to \mathbb{F}_{p^k}^n$ be the public key and S and T two secret affine transformations (cf. section 2). In the sequel we neglect linear terms, as we do not use them for our attack and they also never interfere with the coefficients of quadratic monomials. Thus we assume S and T to be linear transformations. Note that using coefficients of linear terms could only speed up the attack, as soon as they are not all chosen uniformly at random.
The main idea of constructing the trapdoor map $\mathcal{F}$ is to make use of the quadratic map q (cf. definition 4), which is derived by the left parastrophe operation in the quasigroup $\mathbb{F}_{p^k}^d$. This way we obtain a bijective map if we fixing the first d variables, which assures correct decryption later on. Furthermore the authors used a stepwise triangular structure in order to scale the scheme in the number of variables.

Definition 4. *The map* $q = (q^{(1)}, \dots, q^{(d)}) : \mathbb{F}_{p^k}^{2d} \to \mathbb{F}_{p^k}^d$ *is defined by*

$$q^{(s)}(x_1, \dots, x_d, y_1, \dots, y_d) = f^{(s)}(y_s) + \sum_{1 \le i,j \le d} \alpha_{ij}^{(s)} x_i x_j + \sum_{s < i,j \le d} \beta_{ij}^{(s)} y_i y_j + \\ + \sum_{\substack{1 \le i \le d \\ s < j \le d}} \gamma_{ij}^{(s)} x_i y_j + \sum_{1 \le i \le d} \delta_i^{(s)} x_i + \sum_{s < i \le d} \varepsilon_i^{(s)} y_i + \eta^{(s)},$$

for $1 \leq s \leq d$ *and* $f^{(s)}(y_s) = ay_s$, $a \neq 0$ *over fields of odd characteristic and* $f^{(s)}(y_s) = ay_s^2$, $a \neq 0$ *over fields of even characteristic.*

Writing the quadratic part of $q^{(s)}$ in its quadratic form $X^\intercal QX$ with $X = (x_1, \ldots, x_d, y_1, \ldots, y_d)^\intercal$, we can illustrate the matrix Q by figure 15. Note that in the sequel we always assume $f^{(s)}(y_s) = ay_s^2$, as this will yield the worst case attack complexity.

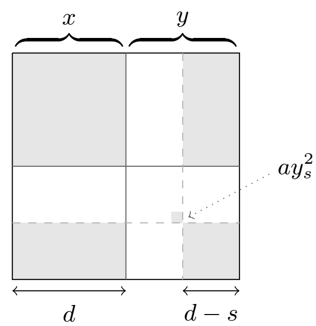


Fig. 15. Quadratic form Q of $q^{(s)}$. Gray parts denote arbitrary values, whereas white parts denote systematic zeros.

The authors of [GS12] fixed the degree of the quasigroup to $d = 8$. Let $\overline{x}_1, \ldots, \overline{x}_{\frac{n}{8}}$ and $\overline{y}_1, \ldots, \overline{y}_{\frac{n}{8}}$ be elements of the quasigroup $\mathbb{F}_{p^k}^d$. Now the central map $\mathcal{F}$: $(x_1, \ldots, x_n) \mapsto (y_1, \ldots, y_n)$ is defined as follows.

$$\begin{array}{llll}
\overline{y}_1 & = (y_1, \ldots, y_8) & := \overline{x}_1 & = (x_1, \ldots, x_8) \\
\overline{y}_2 & = (y_9, \ldots, y_{16}) & := q(\overline{x}_1, \overline{x}_2) & = q(x_1, \ldots, x_{16}) \\
\vdots & = \quad \vdots & := \quad \vdots & = \quad \vdots \\
\overline{y}_{\frac{n}{8}} & = (y_{n-7}, \ldots, y_n) & := q(\overline{x}_{\frac{n}{8}-1}, \overline{x}_{\frac{n}{8}}) & = q(x_{n-15}, \ldots, x_n)
\end{array}$$

Remark 2. The central map $\mathcal{F}$ pictured in figure 16 is a slight simplification of the original central map. More precisely the authors did not use $q(\overline{x}_1, \overline{x}_2)$ but $\widetilde{q}(\overline{x}_1, \overline{x}_2) := D_1 \cdot q(\overline{x}_1, D_2 \cdot \overline{x}_2 + c_2) + c_1$ for some random regular $d \times d$ matrices D_1, D_2 and random vectors c_1, c_2 of dimension d. As we are only considering quadratic coefficients later on, we can safely ignore c_1 and c_2. Further the linear transformation of equations D_1 can be absorbed by T, *i.e.* instead of using $\widetilde{q}$ and recovering the original T, we work with q and recover $T \cdot (I_{\frac{n}{d}} \otimes D_1)$, with $\otimes$ the matrix tensor product of the $\frac{n}{d}$ dimensional identity matrix and D_1. The same holds for the transformation of variables S. Instead of working with $\widetilde{q}$ and recovering the original transformation S, we recover $(I_{\frac{n}{d}} \otimes D_2^{-1}) \cdot S$ and thus work with $\widehat{q}(\overline{x}_1, \overline{x}_2) := q(D_2^{-1}\overline{x}_1, \overline{x}_2)$. As there is no structure hidden in the first component of q, all the systematical zeros in $\widehat{q}$ and q are equal and thus we can assume a central map as defined by $\mathcal{F}$.

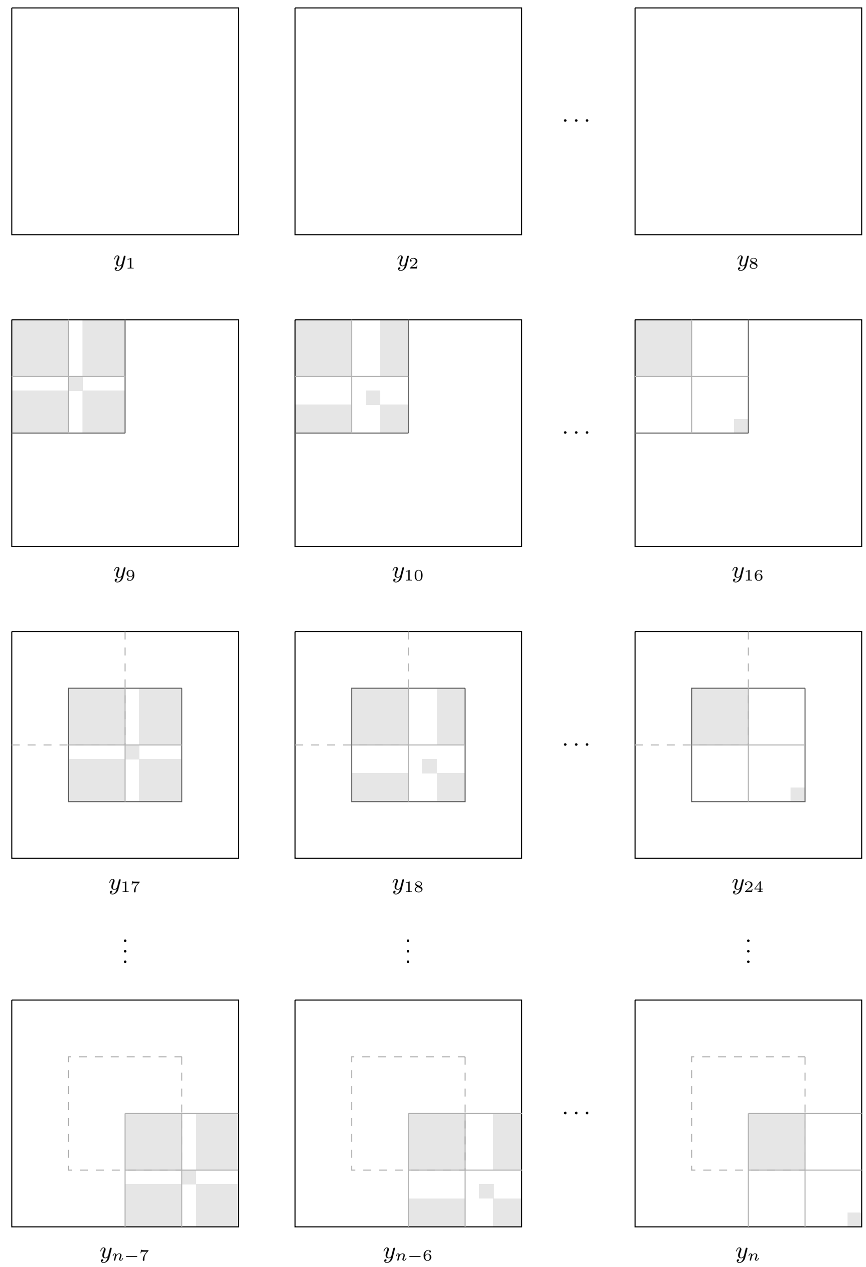


Fig. 16. Matrices of the quadratic form of the central map $\mathcal{F}$ of MQQ-Enc. Gray parts denote some arbitrary values, whereas white parts denote systematical zeros.

Remark 3. In [GS12] the authors did not choose S and T purely at random but as a combination of two circulant matrices. This structure was meant to reduce the key size and speed up the decryption process. We did not use this special structure to speed up our attack yet. As we are recovering $(I_{\frac{n}{d}} \otimes D_2^{-1}) \cdot S$ instead of S and $T \cdot (I_{\frac{n}{d}} \otimes D_1)$ instead of T, for some randomly chosen D_1 and D_2, we lost most of the structure anyway. Therefore we assume to recover some random matrix in the sequel. Note that this gives a worst case complexity of our attack again.

MinRank Attack. Looking out for weaknesses of MQQ-Enc, we are first attracted by the maps $y_1, \ldots, y_8$, as they are not even quadratic and thus can easily be recovery by a MinRank attack [BG06,FdVP08,TW11]. In a nutshell we have to find a linear combination of public polynomials $p^{(i)} = x^\intercal \mathfrak{P}^{(i)} x$ with rank zero. This can be done by solving

$$\sum_{i=1}^{n} \lambda_i \mathfrak{P}^{(i)} \omega = 0 \tag{7}$$

for some $\omega \in \ker(S^\intercal \mathfrak{F}^{(i)} S)$, $1 \leq i \leq 8$ (cf. [FdVP08]). As the rank of $\mathfrak{F}^{(1)}, \ldots, \mathfrak{F}^{(8)}$ is zero, every element ω lies in the kernel and thus (7) is easily solvable by Gaussian elimination. This way we can find a total of 8 linearly independent linear combinations of the public key polynomials with rank zero. Next we exploit that the linear terms of those polynomials only consist of $x_1, \ldots, x_8$. The remaining $n-8$ coefficients are systematical zero. Mapping the public coefficients of the linear terms to those zeros, we obtain $8(n-8)$ linear equations in the elements of S.

At that point this attack does not give us further information, as the quadratic coefficients are also zero in the linear combination of the public key polynomials. So we are not able to learn more about S. Well, the authors of [GS12] prevented this attack by removing r polynomials in the public key anyway. So, they applied the minus modifier known from HFE- to enhance security against direct attacks. This immediately destroys the bijectivity of $\mathcal{P}$ and thus the decryption process is slowed down by the factor of p^{kr} brute-force steps. Therefore kr must be chosen small in practice. A parameter set chosen by the authors is $k = 4$, $r = 2$, $p = 2$, $d = 8$ and $n = 64$. Note that $r \geq d$ must hold in order to prevent the MinRank attack above.

Note that all the central maps have ranks between 9 and 16 (cf. figure 16). To extend the MinRank attack above and recover the full transformation S, we would have to find a linear combination of public polynomials with rank 9. This is also possible, if we remove $r \leq d$ polynomials in the public key. If we remove $d + \ell$ polynomials, we are only able to find a matrix with rank $9 + \ell$. Due to the results of [FDS11,FdVP08] the complexity of solving this MinRank problem is $\binom{n+10}{10}^2$, which is 2^{79} for the parameter set given above.

HighRank Attack. As the complexities of MinRank attacks are quite high, we now investigate HighRank attacks and show that they are as efficient as the original decryption algorithm. The crucial point to start our HighRank attack is that the variable x_{n-7} only occurs in exactly one central map y_{n-7} (cf. figure 16). Remember that every public polynomial can be written as

$$\mathfrak{P}^{(i)} = S^{\mathsf{T}} \left(\sum_{j=1}^{n} t_{ij} \mathfrak{F}^{(j)} \right) S.$$

Thus with probability $1/p^k$, with p^k the number of field elements, we can find a linear combination of two public polynomials $\mathfrak{P}^{(i)} + \lambda \mathfrak{P}^{(j)}$, $i \neq j$ such that $\mathfrak{F}^{(n-7)}$ vanishes. We even can check if we found the right linear combination by the rank property. This approach is also known as HighRank attack, *i.e.* we brute force all linear combinations until we found one with rank $n-1$. This has complexity p^k at most. Now no term $x_i x_{n-7}$ occurs in the sum of underlying central maps $\mathfrak{F}$, *i.e.* due to the missing cross-terms we can apply equivalent keys for S. More precisely we are allowed to map every variable to an arbitrary sum of all variables except x_{n-7} in order to preserve all the zero coefficients of $x_i x_{n-7}$. As x_{n-7} does not occur at all, we are also allowed to map x_{n-7} to x_{n-7}. This way we obtain the equivalent key illustrated in figure 17, *i.e.* it is sufficient to recover column $n-7$ of S.

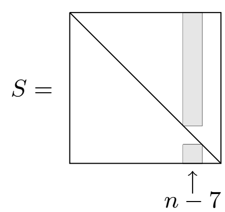


Fig. 17. Equivalent key S with ones at the diagonal, arbitrary values at the gray parts and zeros at the white parts.

We obtain $n-1$ linear equations due to the coefficients $x_i x_{n-7}$, $i = 1, \ldots, n-8, n-6, \ldots, n$ and one quadratic equation due to x_{n-7}^2 in $n-1$ variables $s_{i(n-7)}$. Solving the first $n-1$ linear equations by Gaussian elimination has complexity $(n-1)^3$. We can check the correctness of the result by the quadratic equation, which we did not use so far. The overall complexity of this first step is $p^k + (n-1)^3$. Now that we know the $(n-7)$-th column of S we can apply this transformation to the public key. This way the coefficients of x_{n-7} are plain, *i.e.* a sum of the coefficients of the according central maps without any transformation of variables. This way we can add $\mathfrak{P}^{(1)}$ to all the other polynomials, such that x_{n-7}^2 vanishes. This also deletes $\mathfrak{F}^{(n-7)}$ in all $\mathfrak{P}^{(i)}$, $2 \leq i \leq n$. Now we can repeat this process, as x_{n-6}^2 only occurs in one of the remaining central polynomials. Note that the rank difference is always one and thus the brute force complexity is bounded by p^k. Due to the minus modifier on the public key, this process stops after recovering columns $9, \ldots, n$ of S and ending up with one public polynomial, which is a linear combination of $\mathfrak{F}^{(1)}, \ldots, \mathfrak{F}^{(9)}$. Note that up to this point we did not recover T, *i.e.* all the intermediate polynomials are still linear combinations of some central polynomials. But with the knowledge of large parts of S we can easily calculate backwards and thus separate the original transformation T. We are now in the position of the legitimate user, *i.e.* in order to decrypt a message we have to brute force variables $x_1, \ldots, x_8$.

Our HighRank attack has complexity $(n-r)(p^k + (n-1)^3)$ whereas the original decryption algorithm has complexity roughly $p^{kr}nd^2$. For the parameters proposed in [GS12] this leads to the complexities given in table 3.

Table 3. Complexities of MQQ-Enc.

p^k	k	n	r	d	Decryption	Key Recovery
2	1	256	8	8	2^{22}	2^{32}
4	2	128	4	8	2^{21}	2^{28}
16	4	64	2	8	2^{20}	2^{24}
32	8	32	1	8	2^{19}	2^{19}

We want to thank Simona Samardjiska for implementing the attack and pointing out some subtleties. So an actual implementation need some re-randomization steps, as equivalent keys of some special form only exists with high probability. Further we want to notice that the attack analogously applies to MQQ-Sig [FGJ+11], as the only difference is a more extensive use of the minus modifier (half of the public polynomials are discarded). This does not affect the HighRank attack at all.

Acknowledgments

I want to thank Christopher Wolf (Bochum) for various contributions, especially in the sections about Enhanced STS and Enhanced TTS, Gottfried Herold (Bochum) for his contribution in section 5, Alexander Meurer (Bochum) for correcting minor errors and Peter Czypek (Bochum) for fruitful discussions and helpful remarks on Enhanced TTS. Furthermore I thank the reviewers of [TW12a] for helpful comments.
The author was supported by the German Science Foundation (DFG) through an Emmy Noether grant. Furthermore the author was in part supported by the European Commission through the IST Programme under contract *ICT-2007-216676 Ecrypt II.*

References

[BBD09] Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. *Post-Quantum Cryptography.* Springer, 2009. ISBN 978-3-540-88701-0.

[BFP09] Luk Bettale, Jean-Charles Faugère, and Ludovic Perret. Hybrid approach for solving multivariate systems over finite fields. *In Journal of Mathematical Cryptology*, 3:177–197, 2009.

[BFS04] M. Bardet, J.-C. Faugère, and B. Salvy. On the complexity of gröbner basis computation of semi-regular overdetermined algebraic equations. In *Proceedings of the International Conference on Polynomial System Solving*, pages 71–74, 2004.

[BFSY05] M. Bardet, J.-C. Faugère, B. Salvy, and B.-Y. Yang. Asymptotic expansion of the degree of regularity for semi-regular systems of equations. In P. Gianni, editor, *MEGA 2005 Sardinia (Italy)*, 2005.

[BG06] Olivier Billet and Henri Gilbert. Cryptanalysis of rainbow. In *SCN*, pages 336–347, 2006.

[DH11] Jintai Ding and Timothy J. Hodges. Inverting hfe systems is quasi-polynomial for all fields. In *CRYPTO*, volume 6841 of *Lecture Notes in Computer Science*, pages 724–742. Springer, 2011.

[DHN+07] Jintai Ding, Lei Hu, Xuyun Nie, Jianyu Li, and John Wagner. High order linearization equation (hole) attack on multivariate public key cryptosystems. In *Public Key Cryptography*, volume 4450 of *Lecture Notes in Computer Science*, pages 233–248. Springer, 2007.

[DS05] Jintai Ding and Dieter Schmidt. Rainbow, a new multivariable polynomial signature scheme. In *Conference on Applied Cryptography and Network Security — ACNS 2005*, volume 3531 of *Lecture Notes in Computer Science*, pages 164–175. Springer, 2005.

[DYC+08] Jintai Ding, Bo-Yin Yang, Chia-Hsin Owen Chen, Ming-Shing Chen, and Chen-Mou Cheng. New differential-algebraic attacks and reparametrization of Rainbow. In *Proceedings of the 6th international conference on Applied cryptography and network security*, ACNS'08, pages 242–257, Berlin, Heidelberg, 2008. Springer-Verlag.

[FDS11] Jean-Charles Faugère, Mohab Safey El Din, and Pierre-Jean Spaenlehauer. Gröbner bases of bihomogeneous ideals generated by polynomials of bidegree (1, 1): Algorithms and complexity. *J. Symb. Comput.*, 46(4):406–437, 2011.

[FdVP08] Jean-Charles Faugère, Françoise Levy dit Vehel, and Ludovic Perret. Cryptanalysis of MinRank. In *CRYPTO*, pages 280–296, 2008.

[FGJ+11] Jean-Charles Faugère, Danilo Gligoroski, Erlend Jensen, Rune Odegard, Ludovic Perret, Svein Johan Knapskog, and Smile Markovski. An ultra-fast and provably cma resistant digital signature scheme. In Moti Yung, Liqun C., and Liehuang Z., editors, *The Third International Conference on Trusted Systems - INTRUST 2011*, Lecture Notes in Computer Science, pages 1–10. Springer Verlag, 2011.

[FØPG10] Jean-Charles Faugère, Rune Steinsmo Ødegård, Ludovic Perret, and Danilo Gligoroski. Analysis of the mqq public key cryptosystem. In *CANS*, volume 6467 of *Lecture Notes in Computer Science*, pages 169–183. Springer, 2010.

[Gao12] Shuhong Gao. Private communication, April 2012.

[GC00] 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.

[GH11] Shuhong Gao and Raymond Heindl. Multivariate public key cryptosystems from diophantine equations. *Designs, Codes and Cryptography*, pages 1–18, 2011.

[GMK08] Danilo Gligoroski, Smile Markovski, and Svein J. Knapskog. Multivariate quadratic trapdoor functions based on multivariate quadratic quasigroups. In *Proceedings of the American Conference on Applied Mathematics (MATH)*, pages 44–49, 2008.

[GS12] Danilo Gligoroski and Simona Samardjiska. The multivariate probabilistic enryption scheme mqq-enc. In *SCC*, 2012.

[GT11] Masahito Gotaishi and Shigeo Tsujii. Hidden pair of bijection signature scheme. *IACR Cryptology ePrint Archive*, 2011. http://eprint.iacr.org/2011/353.

[Her12] Gottfried Herold. Polly cracker, revisited, revisited. In *Public Key Cryptography*, volume 7293 of *Lecture Notes in Computer Science*, pages 17–33. Springer, 2012.

[KPG99] 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.

[KPG03] Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced Oil and Vinegar signature schemes — extended version, 2003. 17 pages, citeseer/231623.html, 2003-06-11.

[KS98] Aviad Kipnis and Adi Shamir. Cryptanalysis of the Oil and Vinegar signature scheme. In *Advances in Cryptology — CRYPTO 1998*, volume 1462 of *Lecture Notes in Computer Science*, pages 257–266. Hugo Krawczyk, editor, Springer, 1998.

[KS99] Aviad Kipnis and Adi Shamir. Cryptanalysis of the HFE public key cryptosystem. In *Advances in Cryptology — CRYPTO 1999*, volume 1666 of *Lecture Notes in Computer Science*, pages 19–30. Michael Wiener, editor, Springer, 1999. http://www.minrank.org/hfesubreg.ps or http://citeseer.nj.nec.com/kipnis99cryptanalysis.html.

[KS04] 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.

[MAG] Computational Algebra Group, University of Sydney. *The MAGMA Computational Algebra System for Algebra, Number Theory and Geometry.* http://magma.maths.usyd.edu.au/magma/.

[Moh99] T. Moh. A public key system with signature and master key functions, 1999.

[PBB10] Albrecht Petzoldt, Stanislav Bulygin, and Johannes Buchmann. Selecting parameters for the Rainbow signature scheme. In *PQCrypto*, pages 218–240, 2010.

[PTBW11] Albrecht Petzoldt, Enrico Thomae, Stanislav Bulygin, and Christopher Wolf. Small public keys and fast verification for multivariate quadratic public key systems. In *CHES*, pages 475–490, 2011.

[Sha93] Adi Shamir. Efficient signature schemes based on birational permutations. In *Advances in Cryptology — CRYPTO 1993*, volume 773 of *Lecture Notes in Computer Science*, pages 1–12. Douglas R. Stinson, editor, Springer, 1993.

[SSH11] Koichi Sakumoto, Taizo Shirai, and Harunaga Hiwatari. Public-key identification schemes based on multivariate quadratic polynomials. In *CRYPTO*, volume 6841 of *Lecture Notes in Computer Science*, pages 706–723. Springer, 2011.

[STH89] A. Fujioka S. Tsujii and Y. Hirayama. Generalization of the public-key cryptosystem based on the difficulty of solving non-linear equations. *The Transactions of the Institute of electronics and communication Engineers of Japan*, 1989.

[TG10] Shigeo Tsujii and Masahito Gotaishi. Enhanced sts using check equation - extended version of the signature scheme proposed in the pqcrypt2010. *IACR Cryptology ePrint Archive*, 2010. http://eprint.iacr.org/2010/480.

[TGTF10] Shigeo Tsujii, Masahito Gotaishi, Kohtaro Tadaki, and Ryou Fujita. Proposal of a signature scheme based on sts trapdoor. In *PQCrypto*, pages 201–217, 2010.

[TKI+86] S. Tsujii, K. Kurosawa, T. Itho, A. Fujioka, and T. Matsumoto. A public-key cryptosystem based on the difficulty of solving a system of non-linear equations. *The Transactions of the Institute of electronics and communication Engineers of Japan*, 1986.

[TTGF12a] Shigeo Tsujii, Kohtaro Tadaki, Masahito Gotaishi, and Ryou Fujita. Construction of the tsujii-shamir-kasahara (tsk) type multivariate public key cryptosystem, which relies on the difficulty of prime factorization. IACR Cryptology ePrint Archive, 2012. http://eprint.iacr.org/2012/145.

[TTGF12b] Shigeo Tsujii, Kohtaro Tadaki, Masahito Gotaishi, and Ryou Fujita. Construction of the tsujii-shamir-kasahara (tsk) type multivariate public key cryptosystem, which relies on the difficulty of prime factorization. In *SCC*, 2012.

[TW11] Enrico Thomae and Christopher Wolf. Roots of square: Cryptanalysis of double-layer square and square+. In *Post-Quantum Cryptography (PQCrypto 2011)*, pages 83–97. Springer-Verlag, 2011.

[TW12a] Enrico Thomae and Christopher Wolf. Cryptanalysis of enhanced tts, sts and all its variants, or: Why cross-terms are important. In *AFRICACRYPT*, Lecture Notes in Computer Science. Springer, 2012.

[TW12b] Enrico Thomae and Christopher Wolf. Solving underdetermined systems of multivariate quadratic equations revisited. In *Practice and Theory in Public Key Cryptography (PKC 2012)*. Springer-Verlag, 2012.

[WBP04] Christopher Wolf, An Braeken, and Bart Preneel. 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*, pages 294–309. Springer, September 8–10 2004. Extended version: http://eprint.iacr.org/2004/237.

[WP05] Christopher Wolf and Bart Preneel. Equivalent keys in HFE, C^*, and variations. In *Proceedings of Mycrypt 2005*, volume 3715 of *Lecture Notes in Computer Science*, pages 33–49. Serge Vaudenay, editor, Springer, 2005. Extended version http://eprint.iacr.org/2004/360/, 15 pages.

[WYHL06] Lih-Chung Wang, Bo-Yin Yang, Yuh-Hua Hu, and Feipei Lai. A "medium-field" multivariate public-key encryption scheme. In *CT-RSA*, volume 3860 of *Lecture Notes in Computer Science*, pages 132–149. Springer, 2006.

[YC05] Bo-Yin Yang and Jiun-Ming Chen. Building secure tame-like multivariate public-key cryptosystems: The new TTS. In *ACISP 2005*, volume 3574 of *LNCS*, pages 518–531. Springer, July 2005.