A Study of the Security of Unbalanced Oil and Vinegar Signature Schemes

An Braeken, Christopher Wolf, and Bart Preneel
K.U.Leuven, ESAT-COSIC
http://www.esat.kuleuven.ac.be/cosic/
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium
{An.Braeken,Christopher.Wolf,Bart.Preneel}@esat.kuleuven.ac.be

Cryptology ePrint Archive, Report 2004/222
http://eprint.iacr.org/

Current Version: 2005-08-06
First Revision: 2004-10-11
First Version: 2004-09-04

This is the extended version of the article with the same title [BWP05]. The original article has been published in The Cryptographer's Track at RSA Conference 2005*, Lecture Notes in Computer Science, volume 3376. Alfred J. Menezes, editor, Springer, 2005.*

Abstract

The Unbalanced Oil and Vinegar scheme (UOV) is a signature scheme based on multivariate quadratic equations. It uses m equations and n variables. A total of v of these are called "vinegar variables". In this paper, we study its security from several points of view. First, we are able to demonstrate that the constant part of the affine transformation does not contribute to the security of UOV and should therefore be omitted. Second, we show that the case $n \geq 2m$ is particularly vulnerable to Gröbner basis attacks. This is a new result for UOV over fields of odd characteristic. In addition, we investigate a modification proposed by the authors of UOV, namely to chose coefficients from a small subfield. This leads to a smaller public key. But due to the smaller key-space, this modification is insecure and should therefore be avoided. Finally, we demonstrate a new attack which works well for the case of small v. It extends the affine approximation attack from Youssef and Gong against the Imai-Matsumoto Scheme B for odd characteristic and applies it against UOV. This way, we point out serious vulnerabilities in UOV which have to be taken into account when constructing signature schemes based on UOV.

1 Introduction

1.1 Public Key Cryptography in General

Public key cryptography is used in e-commerce systems for authentication (electronic signatures) and secure communication (encryption). In terms of key distribution, public key cryptography has significant advantages over secret key cryptography. Moreover, efficient signature schemes cannot be obtained by secret key schemes. The security of widely used public key algorithms relies on the difficulty of a small set of problems from algebraic number theory. The RSA scheme relies on the difficulty of factoring large integers, while the difficulty of solving discrete logarithms provides the basis for the ElGamal and Elliptic Curve schemes [MvOV96]. Given that the security of these public key schemes rely on such a small number of problems that are *currently* considered hard, research on new schemes that are based on other classes of problems is worthwhile. Such work provides a greater diversity and avoids the risk that the information society joints all its "crypto eggs" in one basket.

In addition, important results on the potential weaknesses of existing public key schemes are emerging. Techniques for factorisation and solving discrete logarithm continually improve. Polynomial time quantum algorithms can be used to solve both problems [Sho97]; fortunately, quantum computers with more than 7 bits are not yet available and it seems unlikely that quantum computers with 100 bits will be available within the next 10–15 years. Nevertheless, this stresses the importance of research into new algorithms for asymmetric encryption and signature schemes that may not be vulnerable to quantum computers.

1.2 Multivariate Cryptography

One way to achieve more variety in asymmetric cryptology are schemes based on the problem of solving $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic equations ($\mathcal{MQ}$-problem), *e.g.*, see [MI88, Pat96a, Pat96b, CGP01, KPG99, Moh99, CGP03, YC04, KS04]. These schemes use the fact that the $\mathcal{MQ}$-problem, *i.e.*, finding a solution $x \in \mathbb{F}^n$ for a given system of m polynomial equations in n variables each

$$\left\{ \begin{array}{rcl} y_1 & = & p_1(x_1, \ldots, x_n) \\ y_2 & = & p_2(x_1, \ldots, x_n) \\ & \vdots & \\ y_m & = & p_m(x_1, \ldots, x_n)\,, \end{array} \right.$$

for given $y_1, \ldots, y_m \in \mathbb{F}$ and unknown $x_1, \ldots, x_n$ is difficult, namely $\mathcal{NP}$-complete (cf [GJ79, p. 251] and [PG97, App.] for a detailed proof)). In the above system of equations, the polynomials p_i have the form

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

for $1 \le i \le m; 1 \le j \le k \le n$ and $\alpha_i, \beta_{i,j}, \gamma_{i,j,k} \in \mathbb{F}$ (constant, linear, and quadratic terms). This polynomial-vector $\mathcal{P} := (p_1, \ldots, p_m)$ forms the public key of these systems. Moreover, the private key consists of the triple $(S, \mathcal{P}', T)$ where $S \in \mathrm{AGL}_n(\mathbb{F}), T \in \mathrm{AGL}_m(\mathbb{F})$ are affine transformations and $\mathcal{P}' \in \mathcal{MQ}_{n,m}$ is a polynomial-vector $\mathcal{P}' := (p'_1, \ldots, p'_m)$ with m components; each component is a polynomial in n variables $x'_1, \ldots, x'_n$. Throughout this paper, we will denote components of this private vector $\mathcal{P}'$ by a prime $'$. In contrast to the public polynomial vector $\mathcal{P} \in \mathcal{MQ}_{n,m}$, the private polynomial vector $\mathcal{P}'$ does allow an efficient computation of $x'_1, \ldots, x'_n$ for given $y'_1, \ldots, y'_m$. At least for secure $\mathcal{MQ}$-schemes, this is not the case if the public key $\mathcal{P}$ alone is given. The main difference between $\mathcal{MQ}$-schemes lies in their special construction of the central equations $\mathcal{P}'$ and consequently the trapdoor they embed into a specific class of $\mathcal{MQ}$-problems. We refer to [KPG03] for an overview of the different proposed schemes. Note that most of them are already broken *e.g.*, [Pat95, KS99, GC00, FJ03, CDF03, WBP04]. We describe in this paper some new results on the cryptanalysis of the Unbalanced Oil and Vinegar scheme which is still considered to be secure for certain choices of parameters.

1.3 Outline and Achievement

We start with an explanation of the Unbalanced Oil and Vinegar scheme (UOV). Second, we outline in Sect. 3.1 why the constant part of the initial affine transformation can be omitted as it does not contribute to the overall security of UOV. In Sect. 3.2, we give a short description of the Shamir and Kipnis attack against the (balanced) oil and vinegar scheme together with its extension on the unbalanced case. Then we show how this attack breaks the scheme proposed in [KPG03, Sect. 14, ex. 4]. Moreover, we show that the case $n \ge 2m$ is particularly vulnerable to Gröbner basis attacks (Sect. 3.3). This way, we improve a result of Courtois *et al.* who were able to defeat the cases $n \ge 4m$ [CGMT02] — and to some extent also $n \ge 3m$. However, for their most efficient attack to work, they need an even characteristic. The attacks demonstrated in this paper do not have this restriction. Finally, we extend the attack from Youssef and Gong [YG01] against the Scheme B from Imai and Matsumoto [IM85] against Unbalanced Oil and Vinegar scheme — both for even and odd characteristic in Sect. 3.4. The algorithm presented in [YG01] only works for the even case. We conclude with Section 4.

2 Oil and Vinegar Signature Schemes

In 1997, Jacques Patarin suggested a scheme called "Oil and Vinegar" for public key cryptography [Pat97]. This scheme uses multivariate quadratic polynomial equations over small finite fields as public key and similar polynomials as the private keys.

In Oil and Vinegar Schemes, the trapdoor is achieved by a special structure of multivariate quadratic polynomials p'_i. Let $o \in \mathbb{N}$ be the number of oil variables and $v \in \mathbb{N}$ the number of vinegar variables. We have $n = o + v$. Moreover, we have $m = o$ and $o = v$ (or also $n = 2m$) for the case of Oil and Vinegar Schemes.[1] The private polynomials p'_i for $1 \le i \le m$ can be represented by

$$\begin{aligned} p'_i(x'_1, \ldots, x'_n) &:= x'_1 \mathit{Lin'}_{i,1}(x'_1, \ldots, x'_n) + \ldots + x'_v \mathit{Lin'}_{i,v}(x'_1, \ldots, x'_n) + \\ &\quad + \mathit{Af'}_i(x'_1, \ldots, x'_n) \\ &= \sum_{\substack{1 \le j \le v \\ 1 \le k \le n}} \gamma'_{i,j,k} x'_j x'_k + \sum_{1 \le k \le n} \beta'_{i,k} x'_k + \alpha'_i \,, \end{aligned}$$

for $\mathit{Lin'}_{i,j}$ linear, $\mathit{Af'}_i$ affine or — more general — for $1 \le i \le m, 1 \le j \le v$ and $1 \le k \le n$ and $\alpha'_i, \beta'_{i,k}, \gamma'_{i,j,k} \in \mathbb{F}$. Here the vinegar variables $x'_1, \ldots, x'_v$ may be quadratically combined while oil variables $x'_{v+1}, \ldots, x'_n$ do not mix with oil variables.

The trapdoor consists of an affine transformation $S \in \mathrm{AGL}_n(\mathbb{F})$ that mixes the oil and vinegar variables, *i.e.*, $(x'_1, \ldots, x'_n) = S(x_1, \ldots, x_n)$ leads to an affine relation between the public variables x_i and the private variables x'_i. In order to obtain a solution for such a system, the legitimate user fixes all vinegar variables to random values. This way, he obtains a (random) linear equation in the oil variables which can be solved with ordinary Gauss elimination.

Generally speaking, the (unbalanced) oil and vinegar scheme is designed for a signature scheme. It is not suitable for encryption because of the

[1] The above notation clearly has some redundancies. The problem in this context is that different papers about these schemes use very different notation. With the above settings, we use a kind of "generalised notation" which suits most of them.

parameter v, which should be chosen too high for an appropriate security level. To sign a message $M \in \mathbb{F}^m$, we perform the following steps:

1. Assign random variables $a_1, \ldots, a_v$ to all the vinegar variables.
2. After substituting the random values, the system $M = \mathcal{P}'(a)$ becomes linear. Solve this linear system for the remaining m variables $a_1, \ldots, a_o$ of a by Gaussian elimination. If the linear system is singular, return to the first step and try with new random vinegar variables.
3. Map the solution a to the signature x by $x = S^{-1}(a)$.

Verifying the signature $x \in \mathbb{F}^n$ is just the evaluation of x by the public system $\mathcal{P}$. An attacker wants to forge signature on a given message $M = (M_1, \ldots, M_m)$, needs to solve the system:

$$\begin{array}{rcl} M_1 & = & p_1(x_1, \ldots, x_n) \\ & \vdots & \\ M_m & = & p_m(x_1, \ldots, x_n) \end{array}$$

In general, this is an $\mathcal{MQ}$-problem and therefore difficult to solve.

As the original Oil and Vinegar scheme was broken in [KS98], Kipnis *et al.* extended it to the so-called "Unbalanced Oil and Vinegar" signature scheme [KPG99] (see also the extended version [KPG03]). For an Unbalanced Oil and Vinegar Scheme (UOV), we have $v > o$ (or equivalently $n > 2m$). According to [KPG99, KPG03], this case is considered to be secure if the number of vinegar variables is not too "close" to the number of oil variables. In symbols: $v \not\approx o$.

3 Cryptanalysis

3.1 Attacking the Constant Part of UOV

We first show that the affine transformation S in the oil and vinegar scheme should be replaced by a linear transformation. This observation can be easily generalised to any multivariate cryptographic system which uses one or two affine transformations to hide an internal structure (*e.g.*, enTTS [YC04]), cf App. A for a more detailed description of the generalisation.

Consider the affine transformation $S \in \mathrm{AGL}_n(\mathbb{F})$, which can be uniquely represented by an invertible matrix $M_S \in \mathbb{F}^{n \times n}$ and a vector $m_s \in \mathbb{F}^n$, *i.e.*, $S(x) = M_S x + m_s$ for all $x \in \mathbb{F}^n$. Moreover, we can uniquely rewrite S

as $S(x) = (x' + m_s) \circ (M_S x)$ where x' denotes the output of $M_S x$ and $\circ$ represents the composition of functions. We now express the public key $\mathcal{P}$ as a composition of the private key $(\mathcal{P}', S)$:

$$\begin{aligned} \mathcal{P} &= \mathcal{P}' \circ S \\ &= \mathcal{P}' \circ [(x' + m_s) \circ (M_S x)] \\ &= [\mathcal{P}' \circ (x' + m_s)] \circ (M_S x) \\ &= \mathcal{P}'' \circ (M_S x) \end{aligned}$$

for some system of equations $\mathcal{P}''$. As $(x' + m_s)$ is a transformation of degree 1, it does not change the overall degree of $\mathcal{P}''$, *i.e.*, as $\mathcal{P}'$ consists of equations of degree 2 at most, so will $\mathcal{P}''$. In addition, due to its construction, $(M_S, \mathcal{P}'')$ forms a private key for the public key $\mathcal{P}$. Moreover, the private key equations $\mathcal{P}'$ were random equations. The transformation $(x' + m_s)$ does not change the internal structure of $\mathcal{P}'$.

Therefore, we can conclude that the use of an affine instead of a linear transformation does not enhance the overall security of the (unbalanced) oil and vinegar schemes. In fact, we can draw a similar conclusion for all such systems — as long as it is possible to replace the equation $\mathcal{P}'$ by an equation of similar shape. This is always the case if $\mathcal{P}'$ allows a constant, non-zero term and also non-zero linear terms. The corresponding observation for HFE has been made by Toli [Tol03].

Remark: It is also possible to extend this observation to other multivariate cryptographic systems, such as enTTS. See App. A for a discussion. In this context, we want to point out that C^* systems such as Sflash [CGP03] are immune against this observation as it is not possible to embed a constant into the private polynomial.

3.2 The Kipnis and Shamir Attack

After this initial observation, we move on to the attack of Kipnis and Shamir against the *Balanced* Oil and Vinegar scheme. The main idea in this attack is to separate the oil and the vinegar variables, which enables the attacker to access an isomorphic copy of the private key. This way, an attacker can forge arbitrary signatures. The attack is very efficient for all $v \leq m$. We describe the attack here for $v = m$ and thus $2m = n$.

We take only the quadratic terms of the private $\mathcal{P}'$ and the public $\mathcal{P}$ equations into account. In odd characteristic, we can uniquely represent the private key equations (resp. public key equations) by $x^t P_i' x$ (resp. $x'^t P_i x'$)

for $0 \leq i \leq m$, where P'_i and P_i are symmetric matrices (here ^{t} denotes transposition). For even characteristic, the unique symmetric matrices $P'_i + P'^t_i$ and $P_i + P^t_i$ where P'_i and P_i are upper-triangular matrices belonging to $\mathbb{F}^{m \times m}$ are considered. For simplicity, we denote these matrices again by P'_i and P_i.

Note that because of the special structure of the private equations $\mathcal{P}'$, the matrices P'_i for $1 \leq i \leq m$ have the form:

$$P'_i = \begin{pmatrix} 0 & A_i \\ B_i & C_i \end{pmatrix},$$

where $0, A_i, B_i, C_i$ are submatrices of dimension $m \times m$. Because $\mathcal{P} = \mathcal{P}' \circ S$, we obtain

$$P_i = M_S \begin{pmatrix} 0 & A_i \\ B_i & C_i \end{pmatrix} M_S^T.$$

It is clear that each P'_i maps the subspace $x_{m+1} = \cdots = x_{2m}$ (oil subspace) to the subspace $x_1 = \cdots = x_m = 0$ (vinegar subspace). If P'_j is invertible, we can then conclude that each $P'_i P'^{-1}_j$ maps the oil subspace to itself. Consequently the image of the oil subspace under S, called the subspace O, is a common eigenspace for each $P_i P_j^{-1}$ with $1 \leq i < j \leq m$. In [KS98, Sect. 4], Shamir and Kipnis describe two very efficient algorithms for computing the common eigenspace O of a set of transformations. Picking a subspace V for which $O + V = \mathbb{F}^m$ allows us to separate the oil and the vinegar variables. This way, we obtain an isomorphic copy of the private key $(\mathcal{P}, S)$.

In [KPG99, Sect. 4], an extension based on a probabilistic approach of the previous attack is described which also works for $v > m$ (or $n > 2m$) with complexity $O(q^{v-m-1}m^4) = O(q^{n-2m-1}m^4)$.

Application against the Parameters from [KPG03, Sect. 14] In order to avoid the birthday paradox, [KPG99, Sect. 8] describes a modification of UOV which fixes the linear terms of the public equations depending on the message M. This way, it is no longer possible to obtain a collision for different messages $M_1 \neq M_2$ and the same public key, as this public key now also depends on the message M. We consider this construction to be secure and therefore refer to [KPG99, Sect. 8] for a detailed description. However, its application in [KPG03, Sect. 14], Example 4 is flawed. In order to derive a smaller public key, the authors use the trick of restricted coefficients (cf [KPG03, Sect. 10]). In a nutshell, all coefficients in the affine transformation S and the system of private polynomials $\mathcal{P}'$ are not chosen from the field

$\mathbb{F}$ but from a strictly smaller subfield $\tilde{\mathbb{F}}$. This way, the public key $\mathcal{P}$ will only have coefficients from $\tilde{\mathbb{F}}$ as $\mathcal{P} = \mathcal{P}' \circ S$ and subfields are closed under addition and multiplication. Thus, we derive a public key which is a factor of $(\log|\tilde{\mathbb{F}}|/\log|\mathbb{F}|)$ smaller than the original key.

In Example 4, the authors of [KPG03] propose $\mathbb{F} = GF(16)$, $\tilde{\mathbb{F}} = GF(2)$, $m = 16$, $v = 32/48$ and obtain a public key with 2.2kB/4 kB — this is 4 times smaller than without this trick. However, we can apply the attack from the [KPG99, Sect. 4] (see above) against the UOV system over $\tilde{\mathbb{F}} = GF(2)$. This is possible as the Kipnis-Shamir attack does not take linear terms into account but only quadratic terms. The crucial point is that the linear terms are from GF(16) while the quadratic terms are from a subfield isomorphic to GF(2). As soon as we derived an isomorphic copy of the private key $(\mathcal{P}, S)$ over GF(2), we can translate it to GF(16) and are now in the same position as a legitimate user. In particular, we can do all computations necessary to translate the linear parts of the public key (over GF(16)) to the corresponding private key (now, also over GF(16)). As we have $q = 2$, the attack complexity is $2^{32-16-1}.16^4 = 2^{32}$ or $2^{48-16-1}.16^4 = 2^{47}$ and therefore far less than the claimed security level of 2^{64}.

Remark: Although the algorithms from [CGMT02] achieve a lower running time, they are not applicable in this case: they are only able to solve a given instance of an $\mathcal{MQ}$-problem. For this attack, we need the fact that we actually derive a valid private key of the UOV-system.

3.3 Attacks using Gröbner Basis Algorithms

The article of Daum, Felke, and Courtois [CDF03] outlines a way of attacking HFE with Gröbner Basis algorithms. The attack works for $m < n$, *i.e.*, less equations than variables. The idea is to add $n-m$ linear equations. This way, the number of variables can be reduced to m. On the other hand, a system with n variables and m equations is expected to have q^{n-m} solutions on average. Therefore, adding a total of $n - m$ linear equations will lead to one solution on average. Repeating this experiment a few times (*e.g.*, 6, cf Fig. 1), we will find at least one solution.

In our experiments, we fixed $n - m$ variables to random values from $\mathbb{F}$ instead of adding $n - m$ linear equations. From a mathematical point of view, both ideas are equivalent, as the transformation S already gives a random system of degree 1 equations. In a first step, we computed the average number of tries for a series of experiments where n takes values from 10 to 24, and v goes from 1 to $n - 1$. Figure 1 shows that we need only a few tries for a given system of equations until we find a solution. In more than 60% of the cases, we obtain a solution with the first random fixing of variables, after that the number of necessary tries converts quickly to zero.

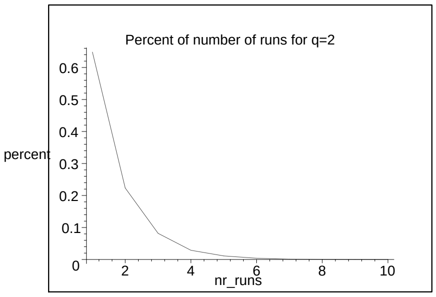

Figure 1: Occurrence of number of runs.

In a second step, we investigated the time complexity of the attack for fixed m and varying v. From experiments, we could conclude that the time complexity increases exponentially with increasing v. This fact can be understood intuitively by the observation that for increasing v, the scheme becomes more random, which makes it more difficult to solve. However, as the number of solutions increases by q^v, *i.e.*, exponentially, the probability of finding one out of these q^v solution becomes higher, too.

In particular, we investigated the logarithmic time complexity (T) for varying the number of equations m for the two values $v = 2m$, $v = 3m$ in characteristics $q = 2, q = 3$ and $q = 16$. The corresponding graphs can be shown in figures 2, 3, and 4. In Table 1, we computed the line that approximately fitted the points from our experiments for the extended Gröbner attack on UOV.

From these experiments, we conclude that the number m of equations should be higher than 38 for characteristic 2 and higher than 24 for characteristic 3 both for $n \geq 2m$ and $n \geq 3m$ in order to obtain a security level greater than 2^{64}. In this paper, we do not predict the behaviour of the curve for $q = 16$ as the graph does not clearly convert to a straight line. To see its behaviour for $m > 8$ — and therefore, to make predictions, we would need to run more experiments. Unfortunately, the current computational power available does not permit this.

These lowerbounds on the minimum number of equations are much higher than the bounds proposed in [KPG03] and later in [CGMT02]. All experiments in this section were carried out with MAGMA and used its

Table 1: Equations representing the time complexity of the extended Gröbner Attack

v	q	Equation	Base
$v = 2m$	$q = 2$	-17.53+1.62m	3.07
$v = 3m$	$q = 2$	-16.66+1.60m	3.03
$v = 2m$	$q = 3$	-23.17+2.74m	6.68
$v = 3m$	$q = 3$	-21.85+2.67m	6.36
$v = 2m$	$q = 16$	-21.14+4.82m	28.20
$v = 3m$	$q = 16$	-21.89+5.03m	32.63

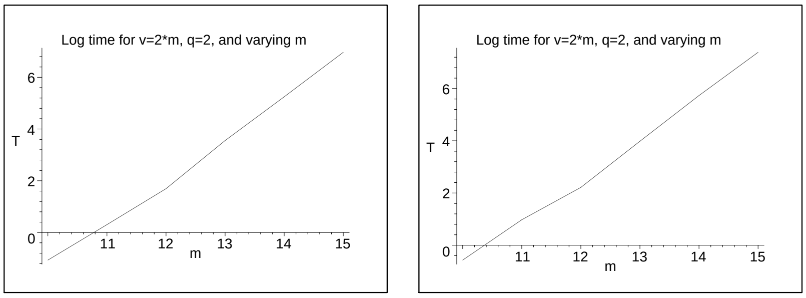

Figure 2: Graphs for logarithmic time in function of m with $v = 2m$, resp. $3m$, and $q = 2$

implementation of Faugere's algorithm F_4 [Fau99]. Given the fact that his algorithm F_5 [Fau02] has a far better running time, we expect the attack to be even more efficient with this method. Unfortunately, we do not have access to an actual implementation of it.

3.4 Exploiting the Existence of Affine Subspaces

This attack extends the attack of Youssef and Gong [YG01] against the Imai and Matsumoto Scheme B [IM85]. It exploits the fact that a cryptosystem can be approximated by several affine equations. The original attack was designed for fields of even characteristic. The attack described in this section is generalised to all characteristics.

In a nutshell, the attack assembles several points belonging to the same affine subspace W. Having w points $x_1, \ldots, x_w \in \mathbb{F}^n$ for which UOV is

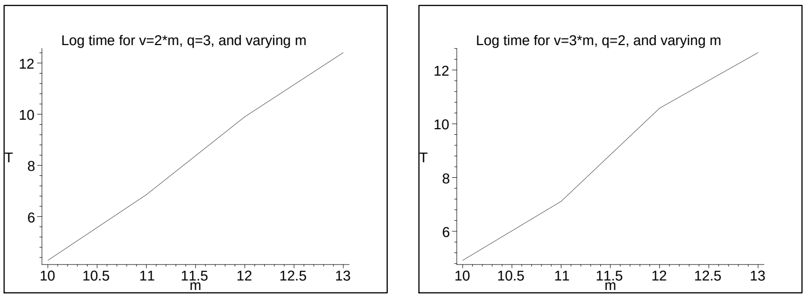

Figure 3: Graphs for logarithmic time in function of m with $v = 2m$, resp. $3m$, and $q = 3$

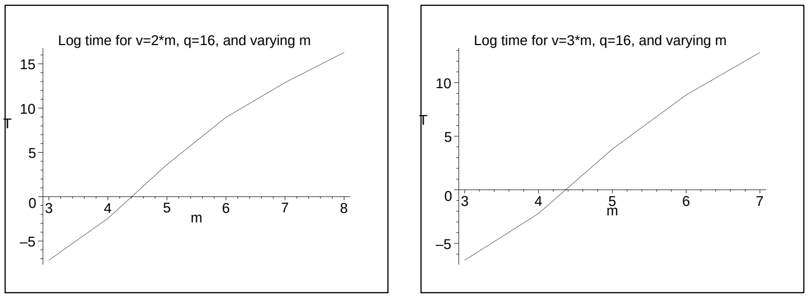

Figure 4: Graphs for logarithmic time in function of m with $v = 2m$, resp. $3m$, and $q = 16$

affine, a function $F(x) = Ax + b$ can be used to describe the output of UOV. To launch the attack, we first compute the corresponding $y_i = UOV(x_i)$ for $1 \leq i \leq w$ and $y_i \in \mathbb{F}^m$. With this knowledge, we can determine for any given y' if it belongs to the subspace W and — if this is the case — compute a vector $a \in \mathbb{F}^w$ with $y' = \sum_{i=1}^{w} a_i y_i$. As the subspace W is affine, we can then determine the corresponding $x' \in \mathbb{F}^n$ as $\sum_{i=1}^{w} a_i x_i$. In the following section, we will present several ways of computing the points x_i, *i.e.*, to determine one or several subspaces W.

For UOV, there exist approx. q^v subspaces of dimension $o = m$ on which UOV is affine. Moreover, all these subspaces are disjunct. If we can find $(o + 1)$ linearly independent points of the same subspace, we completely broke the scheme for this subspace. If we find fewer, *e.g.*, w points, we have at least covered q^w points of the corresponding subspace W. Repeating the search for $(o + 1)$ points q^v times, we break the whole scheme. Note that it is sufficient for the signature forgery of a given $y \in \mathbb{F}^m$ if we know **one** subspace W for which $y \in W$. Therefore, we do not need to know all q^v subspaces but only a small number for forging any given signature $x \in \mathbb{F}^n$ with high probability.

In order to search for points which are in the same subspace, we use the following observation: if the 3 points $R_1, R_2, R_3 \in \mathbb{F}^n$ are in the same affine subspace with respect to UOV, the following condition has to be satisfied:

$$UOV(R_1) - UOV(R_2) - UOV(R_3) + UOV(-R_1 + R_2 + R_3) = 0\,. \quad (1)$$

Using this property, we can determine points of the same affine subspace repeating the heuristic algorithm described in Figure 5 several times. The corresponding algorithm for even characteristic has been described in [YG01].

Input: point R_1, public key $\mathcal{P}$ of UOV
Output: A pair (R_1, R_2) of points which belong to the same affine subspace
repeat
  $pass \leftarrow 0$
  $trials \leftarrow 0$
  $R_2 \leftarrow Random(\mathbb{F}^n)$
  $\delta_x \leftarrow -R_1 + R_2$
  repeat
    $trials \leftarrow trials + 1$
    $R_3 \leftarrow Random(\mathbb{F}^n)$
    $R_4 \leftarrow \delta_x + R_3$
    $\delta_y \leftarrow UOV(R_1) - UOV(R_2) - UOV(R_3) + UOV(R_4)$
    if $(\delta_y = 0)$ **then** $pass \leftarrow pass + 1$
  until $(pass > threshold)$ **or** $(trials > q^v \cdot threshold)$
until $(pass > threshold)$ **or** $(trials > q^v \cdot threshold)$
OUTPUT (R_1, R_2)

Figure 5: Algorithm to find a pair of points in the same affine subspace for which UOV is affine

Repeating this algorithm often enough for a fixed point R_1, we obtain $(o+1)$ linearly independent points of one affine subspace. The complexity of the algorithm will be roughly $O(q^{2v})$, according to the probability that R_1, R_2 and R_3 belong to the same affine subspace.

This attack can be improved using the relation

$$UOV(R_1) + UOV(R_2) - UOV(R_1 + R_2) = b \quad (2)$$

for some fixed $b \in \mathbb{F}^m$. As soon as we find a triple $(R_1, R_2, R_3) \in (\mathbb{F}^n)^3$ of points which yield $\delta_y = 0$ in Algorithm 5, we use (2) to check if all of them yield the same constant b. If this is the case, we can conclude with probability q^{-2m} that all three points belong to the same subspace. At this point, we can change to another algorithm: instead of checking triples, we now check pairs. If the pair (R_1, R') yields the constant b, we found a new candidate belonging to the same subspace as R_1. Using the other points found so far, we can increase the probability that R' is genuine further by q^{-m} with each point we try. We summarise this algorithm:

1. Find a triple $(R_1, R_2, R_3) \in (\mathbb{F}^n)^3$ which satisfies (1).
2. Using this triple and (2), determine the value of the constant $b \in \mathbb{F}^m$.
3. Use (2) to find more points $R' \in \mathbb{F}^n$ in the same subspace.
4. As soon as $(o + 1)$ points $R \in \mathbb{F}^n$ are known, determine the value of the matrix A by Gaussian elimination.

The running time of this algorithm is $O(q^{2v} + (n - v)q^v)$ on average as we chose the points R_2 and R_3 independently from the point R_1 in the first step and R' also independently from R_1. The overall running time to find a total of $(o + 1)$ points in the same subspace becomes therefore $O(q^{2v})$ as $O(oq^v)$ is negligible in comparison to $O(q^{2v})$.

We are able to speed up Algorithm 5 from Section 3.4 if we can spend some memory and also have $m > v$, *i.e.*, we do have "enough" equations in relation to the dimension v of the affine subspaces to be found. This is certainly not true for UOV — here we have typically $m < v$ or even $m < 2v$ (see above). However, for other multivariate quadratic systems, this condition may hold. In particular, it is the case for System B of Matsumoto-Imai, cf [YG01]. We therefore present two ways of speeding up Algorithm 5. We explain it for the example of UOV to simplify the discussion but want to stress that it also works against System B or any other multivariate quadratic system which has affine approximations of small dimension.

3.4.1 Triple-Algorithm

If we can spend $O(kq^{2v})$ of memory for some small k (e.g., $10 \leq k \leq 20$), we can achieve a time/memory-tradeoff for finding **all** subspaces in UOV by using the following technique. In the precomputation phase, we evaluate random pairs $(R_1, R_2) \in_R \mathbb{F}^n \times \mathbb{F}^n$ using (2). The probability for each of these pairs to have points in the same affine subspace is q^{-v} (birthday paradox). Moreover, we know that two points in the same subspace will yield the same constant $b \in \mathbb{F}^m$. On the other hand, two points which are not in the same subspace will yield a random value $v \in \mathbb{F}^m$. The probability for each of these values to occur is q^{-m} with $m > v$. As we were dealing with a total of kq^{2v} pairs, we do not expect two random values $v_1, v_2 \in \mathbb{F}^n$ to occur more often than, say, $\frac{k}{2}$ times. Therefore, all values occurring more often than $\frac{k}{2}$ are constants b with very high probability. Checking the points in the corresponding pairs using (1), we can even distinguish pairs of different subspaces which yield the same constant b. After this precomputation step, we can check for each point $R' \in \mathbb{F}^n$ to which of the q^v subspaces it belongs, using $O(q^v)$ computations on average. After $O(oq^v)$ trials, we have $(o+1)$ points for each subspace and can therefore determine the matrix $A \in \mathbb{F}^{n\times n}$ and the vector b for the affine equation $F(x) = Ax+b$. The above algorithm can be summarised as follows:

1. Use Equation 2 on kq^{2v} random pairs $(R_1, R_2) \in_R \mathbb{F}^n \times \mathbb{F}^n$ and store triples $(b, R_1, R_2) \in \mathbb{F}^m \times (\mathbb{F}^n)^2$
2. Check for each value $b_i \in \mathbb{F}^m$ how often it occurs in the stored list
3. For values b_i which occur at least $\frac{k}{2}$ times, use (1) to check whether the corresponding triples belong to the same affine subspace.
4. Use (2) to determine more points $R' \in \mathbb{F}^n$ for each of these subspaces.

The overall running time of this algorithm is $O(q^{2v})$. However, the drawback is that we need an amount of memory that grows exponentionally with $2v$. Therefore, it seems to be advisable to use the following algorithm $O(q^v)$ times instead. This leads to the same overall running time but requires less memory, namely only $O(q^v)$.

3.4.2 Pair-Algorithm

Using a similar idea, we can also reduce the running time for finding the corresponding subspace W for **one** given point $R_1 \in \mathbb{F}^n$. However, we need $O(kq^v)$ memory for some small k, e.g., $10 \leq k \leq 20$. In this setting, we evaluate pairs (R_1, R_2) for randomly chosen $R_2 \in_R \mathbb{F}^n$ and store the corresponding triples $(b, R_1, R_2) \in \mathbb{F}^m \times (\mathbb{F}^n)^2$. With a similar argument as for the previous algorithm, we expect a random distribution for the values $b_i \in \mathbb{F}^m$ — except if the pair (R_1, R_2) for given R_1, R_2 is in the same vector space W. This event occurs with probability q^{-v}. Therefore, we can assume that the correct value b will occur k times on average and with very high probability at least $\frac{k}{2}$ times. As soon as we have found this value b, we can look for more values R' which satisfy (2). The overall running time of this algorithm is $O(kq^v)$ for the first step and $O(oq^v)$ for the second step, *i.e.*, $O(q^v)$ in total. However, the drawback is that we need an amount of memory that grows exponentionally with v.

Both speed-ups do no longer work for $v, m = \frac{n}{2}$ as the "gap" between q^{-v} and q^{-m} no longer exists. Therefore, we cannot distinguish anymore between values b and random values.

The advantage of the affine approximation attack against UOV is that we know exactly the structure of these affine subspaces. In addition, all these affine subspaces are disjunct. This was not the case for System B from Matsumoto-Imai [IM85]. Theoretical predictions were therefore more difficult.

4 Conclusions

In this paper, we studied the security of the public key signature scheme "Unbalanced Oil and Vinegar" which has been proposed by Kipnis, Patarin, and Goubin in [KPG99] and extended in [KPG03]. We studied its resistance against a modified Gröbner basis attack and concluded that the case $2m < v < 4m$ is particularly vulnerable. In addition, we demonstrated that the choice of parameters in [KPG03, Sect. 14] for Example 4 is insecure under an attack from the previous paper [KPG99]. Moreover, we implemented and simulated an attack using Gröbner bases against the other parameter sets described in [KPG03, Sect. 14]. We conclude that they allow a security-level of 2^{64}, as claimed in the paper. However, as we did not have access to the algorithm F_5 [Fau02], we recommend to be cautious as this algorithm is expected to have a rather small running time, therefore, its effect on UOV should be studied more carefully.

In addition, we showed that the constant part of the affine transformation S does not contribute to the overall security of UOV — at least not for attacks which recover the private key.

Finally, we described a new attack against cryptosystems which have small affine subspaces and applied it against UOV. In particular, parameters with q^v small are shown to be very vulnerable against this type of attack. The attack is very elegant and the occurrence of affine subspaces is a very natural property. We therefore expect it to be efficient against other multivariable cryptographic schemes which have a high number of affine subspaces.

Acknowledgements

We want to thank Jacques Patarin (University of Versailles, France) for fruitful discussions about UOV and pointing out the algorithm from Meier and Tacier [CGMT02] to us. In addition, we want to thank Willi Meier (FH Aargau, Swiss) for answering our questions about the Meier-Tacier algorithm.

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

Disclaimer

References

[BWP05] An Braeken, Christopher Wolf, and Bart Preneel. A study of the security of Unbalanced Oil and Vinegar signature schemes. In *The Cryptographer's Track at RSA Conference 2005*, volume 3376 of *Lecture Notes in Computer Science*. Alfred J. Menezes, editor, Springer, 2005. 13 pages, cf `http://eprint.iacr.org/2004/222/`.

[CDF03] Nicolas T. Courtois, Magnus Daum, and Patrick Felke. On the security of HFE, HFEv- and Quartz. In *Public Key Cryptography — PKC 2003*, volume 2567 of *Lecture Notes in Computer Science*, pages 337–350. Y. Desmedt, editor, Springer, 2002. `http://eprint.iacr.org/2002/138`.

[CGMT02] Nicolas Courtois, Louis Goubin, Willi Meier, and Jean-Daniel Tacier. Solving underdefined systems of multivariate quadratic equations. In *Public Key Cryptography — PKC 2002*, volume 2274 of *Lecture Notes in Computer Science*, pages 211–227. David Naccache and Pascal Paillier, editors, Springer, 2002.

[CGP01] Nicolas Courtois, Louis Goubin, and Jacques Patarin. *Quartz: Primitive specification (second revised version)*, October 2001. `https://www.cosic.esat.kuleuven.ac.be/nessie` Submissions, Quartz, 18 pages.

[CGP03] Nicolas Courtois, Louis Goubin, and Jacques Patarin. *SFlash^{v3}, a fast asymmetric signature scheme — Revised Specificatoin of SFlash, version 3.0*, October 17$^{\text{th}}$ 2003. ePrint Report 2003/211, `http://eprint.iacr.org/`, 14 pages.

[Fau99] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases (F_4). *Journal of Pure and Applied Algebra*, 139:61–88, June 1999.

[Fau02] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (F_5). In *International Symposium on Symbolic and Algebraic Computation — ISSAC 2002*, pages 75–83. ACM Press, July 2002.

[FJ03] Jean-Charles Faugère and Antoine Joux. Algebraic cryptanalysis of Hidden Field Equations (HFE) using gröbner bases. In *Advances in Cryptology — CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 44–60. Dan Boneh, editor, Springer, 2003.

[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.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability — A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979. ISBN 0-7167-1044-7 or 0-7167-1045-5.

[IM85] Hideki Imai and Tsutomu Matsumoto. Algebraic methods for constructing asymmetric cryptosystems. In *Algebraic Algorithms and Error-Correcting Codes, 3rd International Conference, AAECC-3, Grenoble, France, July 15-19, 1985, Proceedings*, volume 229 of *Lecture Notes in Computer Science*, pages 108–119. Jacques Calmet, editor, Springer, 1985.

[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 for realizing ciphtertext of size 100 bit and digital signature scheme. *IEICE Trans. Fundamentals*, E87-A(1):102–109, January 2004. Electronic version: `http://search.ieice.org/2004/files/e000a01.htm\#e87-a,1,102`.

[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/`.

[MI88] Tsutomu Matsumoto and Hideki Imai. Public quadratic polynomial-tuples for efficient signature verification and message-encryption. In *Advances in Cryptology — EUROCRYPT 1988*, volume 330 of *Lecture Notes in Computer Science*, pages 419–545. Christoph G. Günther, editor, Springer, 1988.

[Moh99] T. Moh. A public key system with signature and master key function. *Communications in Algebra*, 27(5):2207–2222, 1999. Electronic version: `http://citeseer/moh99public.html`.

[MvOV96] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 1996. ISBN 0-8493-8523-7, online-version: `http://www.cacr.math.uwaterloo.ca/hac/`.

[Pat95] Jacques Patarin. Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88. In *Advances in Cryptology — CRYPTO 1995*, volume 963 of *Lecture Notes in Computer Science*, pages 248–261. Don Coppersmith, editor, Springer, 1995.

[Pat96a] Jacques Patarin. Asymmetric cryptography with a hidden monomial. In *Advances in Cryptology — CRYPTO 1996*, volume 1109 of *Lecture Notes in Computer Science*, pages 45–60. Neal Koblitz, editor, Springer, 1996.

[Pat96b] Jacques Patarin. Hidden Field Equations (HFE) and Isomorphisms of Polynomials (IP): two new families of asymmetric algorithms. In *Advances in Cryptology — EUROCRYPT 1996*, volume 1070 of *Lecture Notes in Computer Science*, pages 33–48. Ueli Maurer, editor, Springer, 1996. Extended Version: `http://www.minrank.org/hfe.pdf`.

[Pat97] Jacques Patarin. The oil and vinegar signature scheme. presented at the Dagstuhl Workshop on Cryptography, September 1997. transparencies.

[PG97] Jacques Patarin and Louis Goubin. Trapdoor one-way permutations and multivariate polynomials. In *International Conference on Information Security and Cryptology 1997*, volume 1334 of *Lecture Notes in Computer Science*, pages 356–368. International Communications and Information Security Association, Springer, 1997. Extended Version: `http://citeseer.nj.nec.com/patarin97trapdoor.html`.

[Sho97] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, October 1997.

[Tol03] Ilia Toli. Cryptanalysis of HFE, June 2003. arXiv preprint server, `http://arxiv.org/abs/cs.CR/0305034`, 7 pages.

[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*, Lecture Notes in Computer Science, pages 145–151, September 8–10 2004. Extended version: `http://eprint.iacr.org/2004/237`.

[Wol02] Christopher Wolf. *Hidden Field Equations* (HFE) - variations and attacks. Diplomarbeit, Universität Ulm, December 2002. `http://www.christopher-wolf.de/dpl`, 87 pages.

[Wol04] Christopher Wolf. Efficient public key generation for HFE and variations. In *Cryptographic Algorithms and Their Uses 2004*, pages 78–93. Dawson, Klimm, editors, QUT University, 2004.

[WP04] Christopher Wolf and Bart Preneel. Asymmetric cryptography: Hidden Field Equations. In *European Congress on Computational Methods in Applied Sciences and Engineering 2004*. P. Neittaanmäki, T. Rossi, S. Korotov, E. Oñate, J. Périaux, and D. Knörzer, editors, Jyväskylä University, 2004. 20 pages, extended version: `http://eprint.iacr.org/2004/072/`.

[WP05a] 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/`, 12 pages.

[WP05b] Christopher Wolf and Bart Preneel. Superfluous keys in $\mathcal{M}$ultivariate $\mathcal{Q}$uadratic asymmetric systems. In *Public Key Cryptography — PKC 2005*, volume 3386 of *Lecture Notes in Computer Science*, pages 275–287. Serge Vaudenay, editor, Springer, 2005. Extended version `http://eprint.iacr.org/2004/361/`.

[WP05c] Christopher Wolf and Bart Preneel. Taxonomy of public key schemes based on the problem of multivariate quadratic equations. Cryptology ePrint Archive, Report 2005/077, 12th of May 2005. `http://eprint.iacr.org/2005/077/`, 64 pages.

[YC04] Bo-Yin Yang and Jiun-Ming Chen. Rank attacks and defence in Tame-like multivariate PKC's. Cryptology ePrint Archive, Report 2004/061, 23rd March 2004. `http://eprint.iacr.org/`, 21 pages.

[YG01] Amr M. Youssef and Guang Gong. Cryptanalysis of Imai and Matsumoto scheme B asymmetric cryptosystem. In *Progress in Cryptology — INDOCRYPT 2001*, volume 2247 of *Lecture Notes in Computer Science*, pages 214–222. C. Pandu Rangan and Cunsheng Ding, editors, Springer, 2001.

A Attack against Affine Parts in $\mathcal{MQ}$-cryptosystems

In this appendix, we show how the observation from [Tol03] on HFE can be applied against any multivariate cryptographic scheme which allows to embed a constant part into its (quadratic) centre. For the ease of understanding, we demonstrate this attack on a modified UOV which uses two affine transformations S, T rather than only one.

Let $S \in \mathrm{AGL}_n(\mathbb{F})$ and $T \in \mathrm{AGL}_m(\mathbb{F})$. Then there exists a unique, invertible matrix $M_S \in \mathbb{F}^{n\times n}$ (resp. $M_T \in \mathbb{F}^{m\times m}$) and a unique vector $v_s \in \mathbb{F}^n$ (resp. $v_t \in \mathbb{F}^m$) which describes the affine transformation S (resp. T) by $S(x) = M_S x + v_s$ where $x \in \mathbb{F}^n$ is an input vector (resp. $T(x) = M_T x + v_t$ for $x \in \mathbb{F}^m$). Moreover, we can rewrite the affine transformation S as $S(x) = (x' + v_s) \circ (M_S x)$ where x' denotes the output of $M_S x$ and $\circ$ is the composition of functions. In addition, we can rewrite the affine transformation T as $T(x) = (M_T x'') \circ (x + M_T^{-1} v_t)$, where x'' denotes the output of $x + M_T^{-1} v_t$. As M_T is an invertible matrix, the matrix $M_T^{-1} \in \mathbb{F}^{m\times m}$ exists and is unique.

We now express the public key $\mathcal{P}$ as a composition of the private key

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

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

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

for some system of equations $\mathcal{P}''$. As both $(x' + v_s)$ and $(\tilde{x} + M_T^{-1} v_t)$ are transformations of degree 1, they do not change the overall degree of $\mathcal{P}''$, *i.e.*, as $\mathcal{P}'$ consists of equations of degree 2 at most, so will $\mathcal{P}''$. In addition, due to its construction, $(M_S, \mathcal{P}'', M_T)$ form a private key for the public key $\mathcal{P}$. Moreover, the private key equations $\mathcal{P}'$ were random equations. Both

$(x' + v_s)$ and $(\tilde{x} + M_T^{-1} v_t)$ do not change the internal structure of $\mathcal{P}'$. In more detail: let

$$p'_i(x'_1, \ldots, x'_n) = \sum_{\substack{1 \le j \le v \\ 1 \le k \le n}} \gamma'_{i,j,k} x'_j x'_k + \sum_{1 \le k \le n} \beta'_{i,k} x'_k + \alpha'_i$$

be a private polynomial. The transformation T effectively adds linear combinations of polynomials. Therefore, it does not change the fact that the private polynomial p'_i has a constant part α'_i. In particular, this constant part α'_i also depends on the vector v_t of the transformation T. Therefore, we can skip the vector $v_t \in \mathbb{F}^m$ as we already have a random vector $\alpha' \in \mathbb{F}^m$ to determine the constant part of the public key polynomials. To see the effect of the affine transformation S, we have to investigate a little more in detail. Consider

$$\begin{aligned}
& p'_i(x'_1, \ldots, x'_n) \circ S \\
= \;& [\gamma'_{i,1,1} x_1'^2 + \gamma'_{i,1,2} x'_1 x'_2 + \ldots + \gamma'_{i,n,n} x_n'^2 + \beta'_{i,1} x'_1 + \ldots + \beta'_{i,n} x'_n + \alpha_i] \circ S \\
= \;& \gamma_{i,1,1}(s'_{1,1} x_1 + \ldots s_{1,n} x_n + s_{1,0})^2 \\
& + \gamma_{i,1,2}(s'_{1,1} x_1 + \ldots s_{1,n} x_n + s_{1,0})(s_{2,1} x_1 + \ldots s_{2,n} x_n + s_{2,0}) + \ldots \\
& + \gamma'_{i,n,n}(s_{n,1} x_1 + \ldots s_{n,n} x_n + s_{n,0})^2 + \beta'_{i,1}(s_{1,1} x_1 + \ldots s_{1,n} x_n + s_{1,0}) \\
& + \ldots + \beta'_{i,n}(s_{n,1} x_1 + \ldots s_{n,n} x_n + s_{n,0}) + \alpha'_i \\
= \;& \gamma''_{i,1,1} x_1^2 + \gamma''_{i,1,2} x_1 x_2 + \ldots + \gamma''_{i,n,n} x_n^2 + \beta'_{i,1} x_1 + \ldots + \beta''_{i,n} x_n + \alpha''_i
\end{aligned}$$

for the original coefficients $\alpha'_i, \beta'_{i,k}, \gamma'_{i,j,k} \in \mathbb{F}$ (chosen at random). This leads to new coefficients $\alpha''_i, \beta''_{i,k}, \gamma''_{i,j,k} \in \mathbb{F}$ — depending both on $\alpha'_i, \beta'_{i,k}, \gamma'_{i,j,k}$ and S. However, as the coefficient α'_i has been chosen randomly in the first place, there is (from a cryptographic point of view) no difference between $\alpha'_i \in_R \mathbb{F}$ or α''_i. Therefore, we may assume that the affine transformation S is in fact linear and ignore the constant term for cryptanalytic purposes.