Practical key-recovery attack on MQ-Sign

Thomas Aulbach[1], Simona Samardjiska[2], and Monika Trimoska[2]

[1] University of Regensburg, Regensburg, Germany
[2] Radboud Universiteit, Nijmegen, The Netherlands
thomas.aulbach@ur.de,{simonas,mtrimoska}@cs.ru.nl

Abstract. This note describes a polynomial-time key-recovery attack on the UOV-based signature scheme called MQ-Sign. The scheme is a first-round candidate in the Korean Post-Quantum Cryptography Competition. Our attack exploits the sparsity of the secret central polynomials in combination with the specific structure of the secret linear map S. We provide a verification script that recovers the secret key in less than seven seconds for security level V.

1 Introduction

The lack of diversity of hardness assumptions motivated NIST's announcement of reopening the call for post-quantum digital signature proposals, specifying the need for shorter signatures with fast verification. Multivariate cryptography is a contender in this ongoing search for post-quantum digital signature schemes. Since many schemes in multivariate cryptography make use only of quadratic polynomials, public key cryptosystems in the multivariate family are often referred to as $\mathcal{MQ}$ public key cryptosystems. However, the security of some of the $\mathcal{MQ}$ systems does not rely directly on the hardness of the Multivariate Quadratic polynomial ($\mathcal{MQ}$) problem, but rather on the (non)possibility of exploiting a planted trapdoor. The trapdoor usually consists of the knowledge of a so-called central map $\mathcal{F}$ that is easy to invert and a linear or affine transformation $\mathcal{S}$.

One of the most studied trapdoor-based systems is the Unbalanced Oil and Vinegar (UOV) signature scheme [2]. The UOV construction results in short signatures and signing time, but large public and secret keys. Several recent efforts are focused on developing a UOV-based signature scheme with additional structure that results in smaller keys, but without compromising the security. One of those efforts is the MQ-Sign [6] signature scheme submitted to the Korean Post-Quantum Cryptography Competition[3]. The main idea behind MQ-Sign is to have a sparse central map in order to reduce the secret key. In this work, we show how the property of using sparse polynomials can be exploited to develop a polynomial time key-recovery attack. Our attack focuses on recovering the linear transformation $\mathcal{S}$, which allows to subsequently compute the central map $\mathcal{F}$.

[3] www.kpqc.or.kr

2 Preliminaries

Throughout the text, $\mathbb{F}_q$ will denote the finite field of q elements, and $\mathrm{GL}_n(\mathbb{F}_q)$ and $\mathrm{AGL}_n(\mathbb{F}_q)$ will denote respectively the general linear group and the general affine group of degree n over $\mathbb{F}_q$. We will also use the notation $\mathbf{x} = (x_1, \ldots, x_n)^\intercal$ for the vector $(x_1, \ldots, x_n) \in \mathbb{F}_q^n$. Similarly, the entries of a matrix A are denoted by $A_{[ij]}$.

2.1 Multivariate signatures

First, we recall the general principle of $\mathcal{MQ}$ public key cryptosystems.

A typical $\mathcal{MQ}$ public key cryptosystem relies on the knowledge of a trapdoor for a particular system of polynomials over the field $\mathbb{F}_q$. The public key of the cryptosystem is usually given by a multivariate quadratic map $\mathcal{P} = (\mathcal{P}^{(1)}, \ldots, \mathcal{P}^{(m)}) : \mathbb{F}_q^n \to \mathbb{F}_q^m$, where $\mathcal{P}^{(k)}(x_1, \ldots, x_n) = \sum\limits_{1 \le i \le j \le n} \gamma_{ij}^{(k)} x_i x_j + \sum_{i=1}^n \beta_i^{(k)} x_i + \alpha^{(k)}$ for some coefficients $\gamma_{ij}^{(k)}, \beta_i^{(k)}, \alpha^{(k)} \in \mathbb{F}_q$. It is obtained by obfuscating a structured central map

$$\mathcal{F} : (x_1, \ldots, x_n) \in \mathbb{F}_q^n \to \big(\mathcal{F}^{(1)}(x_1, \ldots, x_n), \ldots, \mathcal{F}^{(m)}(x_1, \ldots, x_n)\big) \in \mathbb{F}_q^m,$$

using two bijective affine mappings $\mathcal{S}, \mathcal{T} \in \mathrm{AGL}_n(q)(\mathbb{F}_q)$ that serve as a sort of mask to hide the structure of $\mathcal{F}$. The public key is defined as

$$\mathcal{P} = \mathcal{T} \circ \mathcal{F} \circ \mathcal{S}.$$

The mappings $\mathcal{S}$ and $\mathcal{T}$ are part of the private key s. Besides them, the private key may also contain other secret parameters that allow creation, but also easy inversion of the transformation $\mathcal{F}$. Without loss of generality, we can assume that the private key is $s = (\mathcal{F}, \mathcal{S}, \mathcal{T})$.

Signature Generation. To generate a signature for a message d, the signer uses a hash function $\mathcal{H} : \{0,1\}^\star \to \mathbb{F}^m$ to compute the hash value $\mathbf{w} = \mathcal{H}(d) \in \mathbb{F}^m$ and computes recursively $\mathbf{x} = \mathcal{T}^{-1}(\mathbf{w}) \in \mathbb{F}^m$, $\mathbf{y} = \mathcal{F}^{-1}(\mathbf{x}) \in \mathbb{F}^n$, and $\mathbf{z} = \mathcal{S}^{-1}(\mathbf{y})$. The signature of the message d is $\mathbf{z} \in \mathbb{F}^n$. Here, $\mathcal{F}^{-1}(\mathbf{x})$ means finding one (of possibly many) preimages of $\mathbf{x}$ under the central map $\mathcal{F}$.

Verification. To check if $\mathbf{z} \in \mathbb{F}^n$ is indeed a valid signature for a message d, one computes $\mathbf{w} = \mathcal{H}(d)$ and $\mathbf{w}' = \mathcal{P}(\mathbf{z}) \in \mathbb{F}^m$. If $\mathbf{w}' = \mathbf{w}$ holds, the signature is accepted, otherwise it is rejected.

The standard signature generation and verification process of a multivariate signature scheme works as shown in Figure 1.

2.2 Unbalanced Oil and Vinegar

The Unbalanced Oil and Vinegar signature scheme is one of the oldest multivariate signature schemes. It was proposed by Kipnis and Patarin at EUROCRYPT'99 [2] as a modification of the oil and vinegar scheme of Patarin [4] that was broken by Kipnis and Shamir in 1998 [3].

Signature Generation

$$\begin{tikzcd}
\mathbf{w} \in \mathbb{F}^m \arrow[r, "\mathcal{T}^{-1}"] & \mathbf{x} \in \mathbb{F}^m \arrow[r, "\mathcal{F}^{-1}"] & \mathbf{y} \in \mathbb{F}^n \arrow[r, "\mathcal{S}^{-1}"] & \mathbf{z} \in \mathbb{F}^n \arrow[lll, bend left=30, "\mathcal{P}"]
\end{tikzcd}$$

Signature Verification

Fig. 1. General workflow of multivariate signature schemes.

The characteristic of the oil and vinegar construction is in the special structure of the central map in which the variables are divided in two distinct sets, vinegar variables and oil variables. The vinegar variables are combined quadratically with all of the variables, while the oil variables are only combined quadratically with vinegar variables and not with other oil variables. Formally, the central map is defined as $\mathcal{F} : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^m$, with central polynomials

$$\mathcal{F}^{(k)}(x_1, \ldots, x_n) = \sum_{i \in V, j \in V} \gamma_{ij}^{(k)} x_i x_j + \sum_{i \in V, j \in O} \gamma_{ij}^{(k)} x_i x_j + \sum_{i=1}^{n} \beta_i^{(k)} x_i + \alpha^{(k)} \quad (1)$$

where $n = v + m$, and $V = \{1, \ldots, v\}$ and $O = \{v+1, \ldots, n\}$ denote the index sets of the vinegar and oil variables, respectively.

It can be shown that if an oil an vinegar central map is used in the standard $\mathcal{MQ}$ construction the affine mapping $\mathcal{T}$ does not add to the security of the scheme and is therefore not necessary. Hence the secret key consists of a linear transformation $\mathcal{S}$ and central map $\mathcal{F}$, while the public key is defined as $\mathcal{P} = \mathcal{F} \circ \mathcal{S}$. In order to sign a message, we need to find a preimage of $\mathcal{F}$. This can be done by simply fixing the vinegar variables to some random values. In this way, we obtain a system of m linear equations in m variables, which has a solution with probability around $1 - 1/q$. If the obtained system does not have a solution, we repeat the procedure with different values for the vinegar variables.

Key Generation. It was shown in [5] that for any instance of a UOV secret key $(\mathcal{F}, \mathcal{S})$, there exists an equivalent secret key $(S, \mathcal{F})$ with

$$S = \begin{pmatrix} I_{v \times v} & S_1 \\ 0_{m \times v} & I_{m \times m} \end{pmatrix}. \quad (2)$$

Furthermore, the quadratic polynomials of the central map $\mathcal{F} : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^m$ can be represented using upper triangular matrices $F^{(1)}, \ldots, F^{(m)} \in \mathbb{F}_q^{n \times n}$ where each nonzero coefficient (i, j) in $F^{(k)}$ corresponds to the nonzero coefficient of

$x_i x_j$ in $\mathcal{F}^{(k)}$. Note that the $m \times m$ block on the bottom right of these matrices is empty, since the polynomials of the central map have no quadratic oil terms. Thus, these matrices contain an upper triangular block $F_1^{(k)} \in \mathbb{F}_q^{v \times v}$ and a block $F_2^{(k)} \in \mathbb{F}_q^{v \times m}$ on the top right. In other words, the matrices are of the form:

$$F^{(k)} = \begin{pmatrix} F_1^{(k)} & F_2^{(k)} \\ 0 & 0 \end{pmatrix}.$$

Thus, in order to obtain a key pair, it suffices to first randomly generate $(S_1, F^{(1)}, \ldots, F^{(m)})$ and then compute $(P^{(1)}, \ldots, P^{(m)})$ by evaluating $P^{(k)} = S^\top F^{(k)} S$ and bringing the resulting matrices to upper triangular form.

2.3 MQ-Sign

MQ-Sign is a signature scheme based on UOV. The scheme uses inhomogenous polynomials and each polynomial of the central map can be written as

$$\mathcal{F}^{(k)} = \mathcal{F}_V^{(k)} + \mathcal{F}_{OV}^{(k)} + \mathcal{F}_{L,C}^{(k)}$$

where $\mathcal{F}_V^{(k)} = \sum_{i \in V, j \in V} \gamma_{ij}^{(k)} x_i x_j$ and $\mathcal{F}_{OV}^{(k)} = \sum_{i \in V, j \in O} \gamma_{ij}^{(k)} x_i x_j$. These can alternatively be referred to as the vinegar-vinegar quadratic part and the vinegar-oil quadratic part. Finally, $\mathcal{F}_{L,C}^{(k)}$ refers to the linear and constant part of the polynomials. In the following, we ignore the linear and constant parts, since our attack does not use them. The goal of MQ-Sign is to reduce the size of the secret key compared to traditional UOV. This is achieved by using sparse polynomials for the quadratic part of the central map. The quadratic homogenous part of the sparse polynomials is defined as $\mathcal{F}_V^{(k)} + \mathcal{F}_{OV}^{(k)}$ such that

$$\begin{aligned} \mathcal{F}_V^{(k)} &= \sum_{i=1}^{v} \alpha_i^k x_i x_{(i+k-1(\bmod v))+1} \\ \mathcal{F}_{OV}^{(k)} &= \sum_{i=1}^{v} \beta_i^k x_i x_{(i+k-2(\bmod m))+v+1}. \end{aligned} \tag{3}$$

The size of the secret key is thus reduced to $2vm$ field elements.

The MQ-Sign proposal provides a parameter selection for four variations of the scheme: MQ-Sign-SS, MQ-Sign-RS, MQ-Sign-SR and MQ-Sign-RR. The first S/R in the suffix specifies whether $\mathcal{F}_V$ is defined with sparse or random polynomials. The second S/R refers to the same property, but for $\mathcal{F}_{OV}$. Note that the variation MQ-Sign-RR corresponds to the standard UOV scheme defined with inhomogenous polynomials.

3 An algebraic attack

The attack described in this section consists in solving the Extended Isomorphism of Polynomials (EIP) problem as defined in [6]. We recall here its definition.

$\mathsf{EIP}(n, m, \mathcal{P}, \mathcal{C})$:
Input: an m-tuple of multivariate polynomials $\mathcal{P} = (\mathcal{P}^{(1)}, \mathcal{P}^{(2)}, \ldots, \mathcal{P}^{(m)}) \in \mathbb{F}_q[x_1, \ldots, x_n]^m$ and a special class of multivariate polynomial systems $\mathcal{C} \subseteq \mathbb{F}_q[x_1, \ldots, x_n]^m$.
Question: Find – if any – $S \in \mathrm{GL}_n(q)$ and $\mathcal{F} = (\mathcal{F}^{(1)}, \mathcal{F}^{(2)}, \ldots, \mathcal{F}^{(m)}) \in \mathcal{C}$ such that $\mathcal{P} = \mathcal{F} \circ S$.

In the following, we consider $\mathcal{C}$ to be the class of polynomials defined by $\mathcal{F}_V + \mathcal{F}_{OV}$ where $\mathcal{F}_{OV}$ is defined as in (3). Hence, this algebraic attack applies to MQ-Sign-SS and MQ-Sign-RS.

The computation of the public key for UOV-like signatures schemes can be written in matrix form as

$$\begin{pmatrix} P_1^{(k)} & P_2^{(k)} \\ 0 & P_4^{(k)} \end{pmatrix} = \begin{pmatrix} I & 0 \\ S_1^\top & I \end{pmatrix} \begin{pmatrix} F_1^{(k)} & F_2^{(k)} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} I & S_1 \\ 0 & I \end{pmatrix}.$$

From this we deduce

$$\begin{pmatrix} P_1^{(k)} & P_2^{(k)} \\ 0 & P_4^{(k)} \end{pmatrix} = \begin{pmatrix} F_1^{(k)} & (F_1^{(k)} + F_1^{(k)\top})S_1 + F_2^{(k)} \\ 0 & \mathsf{Upper}(S_1^\top F_1^{(k)} S_1 + S_1^\top F_2^{(k)}) \end{pmatrix}.$$

From the two upper blocks we obtain the following two equations

$$\begin{aligned} P_1^{(k)} &= F_1^{(k)} \\ P_2^{(k)} &= (F_1^{(k)} + F_1^{(k)\top})S_1 + F_2^{(k)}. \end{aligned}$$

From these, we infer that

$$P_2^{(k)} = (P_1^{(k)} + P_1^{(k)\top})S_1 + F_2^{(k)}. \tag{4}$$

The matrices $F_2^{(k)}$ are part of the secret key, but we know that they are sparse. From the description of $\mathcal{F}_{OV}$ in (3) we can see that the value of $F_2^{(k)}$ is known on $(vm - v)$ entries. Since $F_2^{(k)}$ appears linearly in (4), we can extract constraints from the entries where the value of $F_2^{(k)}$ is zero. Let $\widetilde{P}_1 = P_1^{(k)} + P_1^{(k)\top}$. We obtain the following system of equations.

$$\sum_{1 \leqslant p \leqslant v} \widetilde{P}_{1[ip]}^{(k)} S_{1[pj]} - P_{2[ij]}^{(k)} = 0, \quad \forall (i, j, k) \text{ s.t. } F_{2[ij]}^{(k)} = 0. \tag{5}$$

This is a linear system in vm variables. The number of equations that we can obtain if we use all of the m quadratic maps from the public key is $mv(v-1)$. Hence, the system has vm linearly independent equations with overwhelming probability. As such, it can be solved efficiently through Gaussian Elimination.

Note from (5) that each equation in the system contains variables from only one column of S_1. This observation allows us to further optimize the attack by solving for one column at a time. Instead of solving one linear system in

vm variables, we solve m linear systems in v variables. The complexity of this approach is $\mathcal{O}(mv^{\omega})$, where ω is the linear algebra constant.

Our attack relies on two key properties. Firstly, we exploit the sparseness property of the vinegar-oil quadratic part. Secondly, we use the specific structure of the linear transformation S, as per the *equivalent keys* key generation technique. This technique is used in most modern UOV-based signature schemes, including MQ-Sign. As future work, we envision exploring the hardness of the EIP problem when one of these two conditions is not met. In other words, we will look at the MQ-Sign-SR variant, or the MQ-Sign-{S/R}S variant with a random linear transformation S.

3.1 Implementation.

We provide a verification script in MAGMA [1] where we implement the key generation of MQ-Sign and then run the main algorithm for recovering the secret key from the public key as input. The running time of the attack on a laptop is 0.6 seconds for the proposed parameters for security level I, 2.3 seconds for security level III and 6.9 seconds for security level V. We also provide an equivalent SageMath [7] script that is slower. The verification scripts can be found at

`https://github.com/mtrimoska/MQ-Sign-attack`.

References

[1] W. Bosma, J. Cannon, and C. Playoust. The Magma Algebra System. I. The User Language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. Computational algebra and number theory (London, 1993).

[2] A. Kipnis, J. Patarin, and L. Goubin. Unbalanced Oil and Vinegar Signature Schemes. In J. Stern, editor, *Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding*, volume 1592 of *Lecture Notes in Computer Science*, pages 206–222. Springer, 1999.

[3] A. Kipnis and A. Shamir. Cryptanalysis of the Oil & Vinegar Signature Scheme. In H. Krawczyk, editor, *CRYPTO '98*, volume 1462 of *Lecture Notes in Computer Science*, pages 257–266. Springer, 1998.

[4] J. Patarin. The oil and vinegar signature scheme, 1997.

[5] A. Petzoldt. *Selecting and reducing key sizes for multivariate cryptography.* PhD thesis, Darmstadt University of Technology, Germany, 2013.

[6] K.-A. Shim1, J. Kim1, and Y. An. MQ-Sign: A New Post-Quantum Signature Scheme based on Multivariate Quadratic Equations: Shorter and Faster. https://www.kpqc.or.kr/images/pdf/MQ-Sign.pdf, 2022.

[7] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 9.5)*, 2022. `https://www.sagemath.org`.