IPRainbow

Ryann Cartor[1], Max Cartor[2], Mark Lewis[2], and Daniel Smith-Tone[2,3]

[1]School of Mathematical and Statistical Sciences, Clemson University,
Clemson, South Carolina, USA
[2]Department of Mathematics, University of Louisville,
Louisville, Kentucky, USA
[3]National Institute of Standards and Technology,
Gaithersburg, Maryland, USA

rcartor@clemson.edu, maxwell.cartor@louisville.edu,
mark.lewis.2@louisville.edu, daniel.smith@nist.gov

Abstract. The Rainbow signature scheme is the only multivariate scheme listed as a finalist in round 3 of the NIST post-quantum standardization process. A few recent attacks, including the intersection attack, rectangular MinRank attacks, and the "simple attack," have changed this landscape; leaving questions about the viability of this scheme for future application.

The purpose of this paper is to analyze the possibility of repairing Rainbow by adding an internal perturbation modifier and to compare its performance with that of UOV at the same security level. While the costly internal perturbation modifier was originally designed with encryption in mind, the use of schemes with performance characteristics similar to Rainbow is most interesting for applications in which short signatures or fast verification is a necessity, while signing can be done offline. We find that Rainbow can be made secure while achieving smaller keys, shorter signatures and faster verification times than UOV, but this advantage comes at significant cost in terms of signing time.

Keywords: Multivariate Cryptography, Rainbow, MinRank.

1 Introduction

As the world marches toward a future of widespread quantum computing, the need for secure post-quantum cryptosystems is imperative. One branch of post-quantum cryptography is multivariate cryptography. Multivariate cryptosystems are based on the MQ problem, which is the problem of solving a system of nonlinear equations over a finite field. The Rainbow signature scheme is the only multivariate cryptosystem among the round 3 finalists of the National Institute for Standards and Technology (NIST) Post Quantum Standardization process [18].

This work was partially supported by a grant from the Simons Foundation (712530, DCST).

The first massively multivariate cryptosystem published in the west was C^*, introduced in 1988 by Matsumoto and Imai [17]. This encryption scheme is an example of a big-field scheme, which makes use of computations in both a base field and an extension field. Given a base field $\mathbb{F}_q$ and an extension field $\mathbb{K}$, C^* will have an $\mathbb{F}_q$-quadratic central map $F : \mathbb{K} \to \mathbb{K}$, whose structure is hidden by function composition. C^* was broken by Patarin in 1995 [19] with the introduction of linearization equations, which exploits a linear relationship between plain text and ciphertext vectors. Many modifiers were introduced after the break of C^* in the hopes of repairing the scheme, including minus, projection and internal perturbation modifiers, see [22,21,9]. The security of this family of modifiers is discussed in [6].

Another avenue of study is to consider small-field cryptosystems, which are multivariate schemes that work over only one field, $\mathbb{F}_q$. Patarin introduced the small field scheme Oil and Vinegar [20] as a new possible multivariate signature scheme. The Oil and Vinegar scheme consists of two different types of variables, specifically oil variables and vinegar variables. In the original presentation of the scheme, the number of oil variables was equal to the number of vinegar variables. Cryptanlysis from Kipnis and Shamir [16] showed this parameterization to be insecure, which lead to the Unbalanced Oil and Vinegar scheme (UOV) which necessitates that the number of vinegar variables is much larger than the number of oil variables.

The Rainbow Signature scheme [11] is an extension of the UOV signature scheme that consists of layers of UOV central maps. Despite the relatively large size of public keys associated with the Rainbow scheme, its short signatures and high degree of computational efficiency in verification make it an attractive choice for many applications, such as verified/secure boot and certificate transparency.

Following the support minors advance in MinRank methodology, see [1], new attacks in [3], and more significantly [4], have reduced the security of Rainbow below their claimed NIST security levels, rendering the scheme significantly less efficient. The critical insight of these attacks is that information about the secret key can be encoded in equations in the public variable set and combined with the public equations, resulting in a significant enhancement of a direct algebraic attack targeting a hidden subspace.

In this paper, we introduce the variant “IPRainbow,” which adds an internal perturbation modifier to the Rainbow central map. This perturbation of the private key disrupts the above attacks by decoupling the new relations from the public equations; specifically, the public equations are satisfied by a vector in the secret subspace with low probability, corrupting the attack mechanism. We analyze the security and efficiency of this new scheme in comparison with UOV. We show that it is still possible for Rainbow to outperform UOV in terms of verification speed, signature size and public key size; however, these enhancements come at a significant cost in signing time.

2 UOV and Rainbow

2.1 Oil and Vinegar

The Oil and Vinegar signature scheme was introduced in [20] as a response to Patarin's linearization equations in [19], which broke the first multivariate cryptosystem C^*. The scheme consists of two types of variables over a finite field $\mathbb{F}_q$, namely oil and vinegar variables. Furthermore, the number of oil variables and the number of vinegar variables were equal in the original parameterization. Kipnis and Shamir broke this balanced Oil and Vinegar scheme in [16], so we now only consider the case of Unbalanced Oil and Vinegar (UOV), where the number of vinegar variables is sufficiently large enough that the statistical attacks of [15] and the intersection attack from [3] are infeasible.

Let $\mathbf{x} = (x_1, \ldots, x_v, x_{v+1}, \ldots, x_n) \in \mathbb{F}_q^n$. We will call $x_1, \ldots, x_v$ the vinegar variables whereas $x_{v+1}, \ldots, x_n$ will denote the oil variables. We define the following central map $F = (f_1, \ldots, f_{v+1})$, where each f is of the form:

$$f(\mathbf{x}) = \sum_{i=1}^{v} \sum_{j=i}^{v} \alpha_{ij} x_i x_j + \sum_{i=1}^{v} \sum_{j=v+1}^{n} \beta_{ij} x_i x_j + \sum_{i=1}^{n} \gamma_i x_i + \delta$$

To create the public key equations P we compose F with an invertible affine map $T : \mathbb{F}_q^n \to \mathbb{F}_q^n$ to get $P = F \circ T$. Notice that although F is a quadratic map, F is linear on the oil variables. Therefore, inversion of the central map is completed by choosing random values in $\mathbb{F}_q$ for each of the vinegar variables. Each equation is then set equal to zero and Gaussian Elimination is used to solve for the remaining oil variables. If no solution is found, choose different values for the vinegar variables. Repeat this process until a solution is found.

2.2 Rainbow

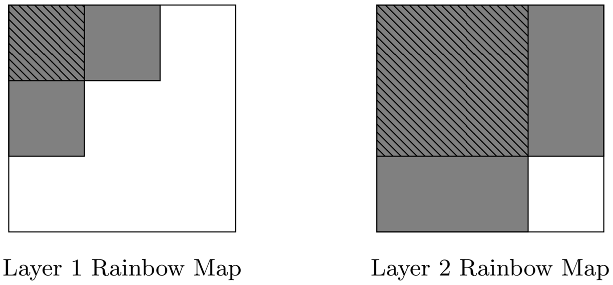
The Rainbow signature scheme was first introduced in [11]. Rainbow can be thought of a banded construction of UOV, where Rainbow consists of L different UOV layers. Rainbow is the only multivariate signature scheme to make it into the finalists of the third round of the NIST standardization process [18], but the scheme has recently faced substantial attacks from [3] and [4].

To create a Rainbow signature scheme, we will still consider input vectors of the form $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$, but now each layer of Rainbow will contain a different number of vinegar variables. Consider a sequence of integer values $0 < v_1 < v_2 < \ldots < v_L < n$, and corresponding sets of variables $V_1 = \{x_1, \ldots, x_{v_1}\}, V_2 = \{x_1 \ldots, x_{v_1}, \ldots, x_{v_2}\}, \ldots, V_L = \{x_1, \ldots, x_{v_L}\}$ that contain the vinegar variables for the 1st, 2nd, …, and Lth layers, respectively. Note that the oil variables in layer ℓ will contain $O_\ell = \{x_{v_\ell+1}, \ldots, x_n\}$. Furthermore $V_1 \subset V_2 \subset \cdots \subset V_L$, whereas $O_L \subset \cdots \subset O_2 \subset O_1$.

Each layer ℓ will be composed of $n - v_\ell$ equations, which is also the number of oil variables in that layer. A polynomial in the ℓth layer will have the form:

$$f_\ell(\mathbf{x}) = \sum_{i=1}^{v_\ell} \sum_{j=1}^{v_\ell} \alpha_{ij\ell} x_i x_j + \sum_{i=1}^{v_\ell} \sum_{j=v_\ell+1}^{n} \beta_{ij\ell} x_i x_j + \sum_{i=1}^{n} \gamma_{i\ell} x_i + \delta_\ell$$

The public key is then formed by composing the central map with two affine maps, $P = U \circ F \circ T$. The Rainbow parameterization proposed in the current submission [10] to NIST's standardization process utilizes $L = 2$ layers, as is historically typical. Also, in order to speed up key generation, by convention we consider only homogeneous polynomials f_i.



Layer 1 Rainbow Map

Layer 2 Rainbow Map

Fig. 1. These diagrams represent the matrices corresponding to the central map of a Rainbow scheme with two layers. White areas represent entries of the matrix that are zero, whereas gray areas correspond to possibly nonzero entries. The lined gray areas correspond to coefficients on the quadratic vinegar terms, and solid gray areas correspond to mixed vinegar and oil coefficients.

To invert the central map $F = f^{(1)}, \dots, f^{(n)}$ we choose values for the first layer vinegar variables $x_1, \dots, x_{v_1}$ and substitute these values into the first layer maps $f^{(1)}, \dots, f^{(o_1)}$. Then we solve the resulting linear system in the first layer oil variables $x_{v_1+1}, \dots, x_{v_2}$. Next we substitute the values of these variables into the central maps $f^{(v+1)}, \dots, f^{(n)}$ and solve similarly for the remaining variables, $x_{v_2+1}, \dots, x_n$.

3 Known Attacks of Rainbow

3.1 Background

MinRank attacks have proven to be highly effective against multivariate schemes. We can define the MinRank problem as follows:

Problem 1 (MinRank Problem) *Given matrices $A_1, \dots, A_k \in \mathbb{F}_q^{n \times m}$ and $r \in \mathbb{N}$, decide if there exists a linear combination $y_1, \dots, y_k \in \mathbb{F}_q$ (not all zero) such that*

$$\mathit{rank}\left(\sum_{i=1}^{k} y_i A_i\right) \le r.$$

The MinRank attack was first introduced in [14] as the first effective attack on the multivariate scheme HFE. This first iteration of the MinRank attack is commonly called the Kipnis-Shamir (KS) attack. Other methods have since followed, including minors modeling and support minors modeling [13,2]. The goal of MinRank attacks is to try to find linear combinations of the public matrices that result in a matrix with low rank. This is useful against schemes like HFE and C^* as the central map has low rank, thus the attacker can find an equivalent key. The MinRank attack is also applicable to Rainbow, since the first layer maps exhibit a rank defect.

The complexity of MinRank attacks are tied to the complexity of polynomial solvers, such as the XL algorithm of [8]. These algorithms create a larger generating set by generating higher degree equations through monomial multiplication. The first degree fall of the XL-style algorithm should occur at the degree corresponding to the first non-positive coefficient of the corresponding Hilbert Series.

We briefly explain the idea of the support minors modeling of [2], see [2] for the details. The support minors system from [2] involves two variable sets, the so-called "minor" variables, whereas the above variables are given the moniker "linear." As mentioned in [24] with more details following in [23], the additivity of Hilbert Series can be generalized to a multi-series respecting disparate variable sets. Due to the large number of the minor variables, we may restrict ourselves to consider the algebra of degree one in the minor variables and graded with respect to the degree of the linear variables. In this way, we can "forget" the minor variables and recover a univariate series.

In [2], the coefficients of this series for degree b where m' columns are used is derived. Specifically, the degree b coefficient is given by

$$\sum_{i=0}^{b}(-1)^i\binom{m'}{o_2+i}\binom{n+i-1}{i}\binom{n+b-i-1}{b-i}.$$

Note that we must include all n matrices. Thus we obtain the series

$$G(t)=\sum_{b=0}^{\infty}\sum_{i=0}^{b}(-1)^i\binom{m'}{o_2+i}\binom{n+i-1}{i}\binom{n+b-i-1}{b-i}t^b.$$

Given that the solving bi-degree is $(1,b)$, it follows that the support-minors algorithm solves a MinRank instance of k many $n\times m$ matrices with a target rank r with an estimated cost of

$$3(k-1)(r+1)\binom{m'}{r}^2\binom{k+b-2}{b}^2$$

field multiplications. Note that it is sometimes more efficient to increase b if it is possible to use a smaller m'.

3.2 Rectangular MinRank Attack

In this section, we describe the attack presented in [3]. The public key of a multivariate cryptosystem is a set of m nonlinear equations in n variables. We can consider the quadratic form of each equation f_i, which will be an $n \times n$ matrix $\mathbf{F}_i$ of the form:

$$f_i(\mathbf{x}) = \mathbf{x}\mathbf{F_i}\mathbf{x}^\top.$$

It is often useful to consider the public or private key of a multivariate scheme with m equations in n variables as a single 3-tensor. In this vein, consider the Rainbow public and private keys as 3-tensors of dimension $n \times n \times m$. In particular, consider Figure 2, where the white represents zero coordinates and the gray represents nonzero coordinates. Given a vector from O_2, the multiplication of the public key with this oil vector will result in a matrix with nonzero elements only in the upper $(v + o_1) \times o_2$ coordinates.

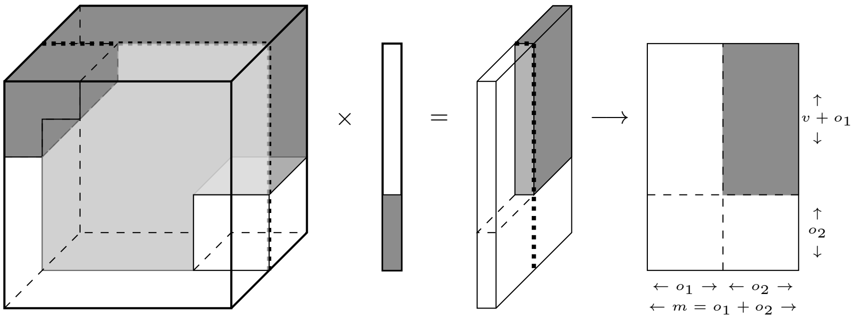


Fig. 2. Multiplication of a Rainbow public key by a vector in O_2.

Thus, if we can find a linear combination of the public key equations such that

$$\operatorname{rank}\left(\sum_{i=1}^{n-o_2+1} y_i \mathbf{P}_i\right) \leq o_2,$$

then it is probable that $\mathbf{y} \in O_2$. This instance of the MinRank problem requires $n - o_2 + 1$ different $n \times m$ matrices with a target rank of o_2.

3.3 Simple Attack

The Simple attack of [4] breaks the Rainbow I parameters quite efficiently. The technique can also be used in conjunction with the Rectangular MinRank attack to significantly impact security for the higher security parameters, Rainbow III and Rainbow V as well. The attack introduces a new strategy to find vectors in O_2, which then can be used to remove the outer layer of the Rainbow public key, leaving us with a small instance of UOV.

The Simple attack will make use of the discrete differential of the public key, defined as

$$P'(\mathbf{x}, \mathbf{y}) = P(\mathbf{x} + \mathbf{y}) - P(\mathbf{x}) - P(\mathbf{y}).$$

We denote $W := P(O_1)$, $\dim(O_2) = \dim(W) = o_2$. From analysis in [3], we know that for $\mathbf{y} \in O_2$, $P'(\mathbf{x}, \mathbf{y}) \in W$ for any $\mathbf{x} \in \mathbb{F}_q^n$. This structure is illustrated in Figure 3.

$$\begin{tikzcd}
O_2 \arrow[r, phantom, "\subset"] \arrow[d, "P"'] \arrow[dr, "P'(x,\cdot)"'] & O_1 \arrow[r, phantom, "\subset"] \arrow[d, "P"] & \mathbb{F}_q^n \arrow[d, "P"] \\
\{0\} \arrow[r, phantom, "\subset"] & W \arrow[r, phantom, "\subset"] & \mathbb{F}_q^m
\end{tikzcd}$$

Fig. 3. Structure of nested subspaces.

By fixing a random $\mathbf{x} \in \mathbb{F}_q^n$, we can define

$$D_{\mathbf{x}}(\mathbf{y}) = P(\mathbf{x} + \mathbf{y}) - P(\mathbf{x}) - P(\mathbf{y}),$$

where for any nonzero $\mathbf{x}$ and $\mathbf{y} \in O_2$, $D_{\mathbf{x}}(\mathbf{y}) \in W$. So, if we restrict our domain to O_2, we see that $D_{\mathbf{x}}|_{O_2}$ is a linear map from O_2 to W. Therefore, for any choice of basis, we can express $D_{\mathbf{x}}|_{O_2}$ as an $o_2 \times o_2$ matrix. For a fixed $\mathbf{x}$, the probability that there exists a nontrivial kernel vector $\mathbf{y} \in O_2$ such that $D_{\mathbf{x}}(\mathbf{y}) = 0$ is the same as the probability that a random $o_2 \times o_2$ matrix will be singular. This is a well known problem and gives us the probability

$$1 - \prod_{i=0}^{o_2-1} (1 - q^{i-o_2}),$$

which for large q is approximately q^{-1}. This leads to the strategy of guessing a random vector $\mathbf{x}$ and trying to find a solution to the system of equations

$$\begin{cases} D_{\mathbf{x}}(\mathbf{y}) = 0 \\ P(\mathbf{y}) = 0. \end{cases}$$

If we can find such a $\mathbf{y}$, then it is likely that $\mathbf{y} \in O_2$. If we cannot find such a $\mathbf{y}$, choose a different $\mathbf{x}$ and repeat the process.

Once we have a vector $\mathbf{y} \in O_2$, we can generate a subspace of W by computing

$$\langle P'(\mathbf{e}_1, \mathbf{y}), \cdots, P'(\mathbf{e}_n, \mathbf{y}) \rangle \subseteq W.$$

Analysis from [4] shows that with high probability the generated space will be equal to W. This gives us access to a subspace of, if not the entirety of, the secret space W. Given this information, we can create a map V that allows us to find the secret space O_2. We define V to be the change of variables such that

$$V \circ P(\mathbf{x}) = \begin{cases} P_1(\mathbf{x}) \\ P_2(\mathbf{x}) \end{cases}$$

where $P_1 : \mathbb{F}_q^n \to \mathbb{F}_q^{m-o_2}$ and $P_2 : \mathbb{F}_q^n \to \mathbb{F}_q^{o_2}$. From here, we can find the kernel of the map

$$\mathbf{x} \mapsto \begin{pmatrix} P'(\mathbf{e}_1, \mathbf{x}) \\ \vdots \\ P'(\mathbf{e}_n, \mathbf{x}) \end{pmatrix}.$$

With high probability, the kernel of this map will be O_2. Beullens completes the attack using another change of variable map. Let $U : \mathbb{F}_q^n \to \mathbb{F}_q^n$ send the last o_2 coordinates to O_2 and then consider

$$V \circ P \circ U(\mathbf{x}) = \begin{cases} F_1(\mathbf{x}) \\ F_2(\mathbf{x}) \end{cases}.$$

It is shown in [4] that finding a preimage P is equivalent to finding a preimage of F, and finding a preimage of F_1 gives a preimage of F. F_1 is a system of $m-o_2$ equations that has the structure of a UOV public key with $n-o_2$ variables and an oil space of dimension $m-o_2$. Given this smaller UOV system, the remainder of the attack is to solve a system of m equations in $n-m$ unknowns. Under the assumption that this system is semi-regular, it can be solved with an XL-style algorithm at degree

$$d_{sr} = \min_d \left\{ [t^d] \frac{(1-t^2)^m}{(1-t)^{n-m}} \leq 0 \right\}.$$

In such a case, the complexity of the attack is dominated by the cost of the block Wiedemann [7] step in the XL algorithm. This cost is well known to be

$$3\binom{n-m-1+d_{sr}}{d_{sr}}\binom{n-m+1}{2},$$

where $\binom{n-m-1+d_{sr}}{d_{sr}}$ is the number of monomials (i.e., the dimension of the square Macaulay submatrix), and $\binom{n-m+1}{2}$ is the number of nonzero entries in each row of the Macaulay matrix.

The Simple attack of [4] can be combined with the Rectangular MinRank attack of [3]. We may construct a Hilbert series in this case by pasting together the rectangular MinRank support minors system with the two systems

$$D_{\mathbf{x}}(\mathbf{y}) = \mathbf{0}, \text{ and}$$
$$P(\mathbf{y}) = \mathbf{0}.$$

The latter two systems involve the same variable set, thus we obtain the Hilbert series

$$\frac{(1-t)^m(1-t^2)^m}{(1-t)^n} = \frac{(1-t^2)^m}{(1-t)^{n-m}}.$$

To obtain the Hilbert series for the entire system, we merely add the relations in the already present variables. Under the assumption of semi-regularity of the resulting system, we obtain the series

$$(1-t)^m(1-t^2)^m G(t),$$

where $G(t)$ is as described in Section 3.1. This is a similar result to what was observed in [3].

4 IPRainbow

4.1 Description of IPRainbow

We will consider the internal perturbation (IP) modifier, see [9], applied to the Rainbow scheme. The IP modifier can be described as follows. Let $Q : \mathbb{F}_q^n \to \mathbb{F}_q^m$ be a set of m quadratic equations where q_i denotes the i^{th} equation. Given a public key $P : \mathbb{F}_q^n \to \mathbb{F}_q^m$ whose i^{th} equation is denoted p_i we create the internally perturbed public key $\tilde{P}(\mathbf{x})$ by defining

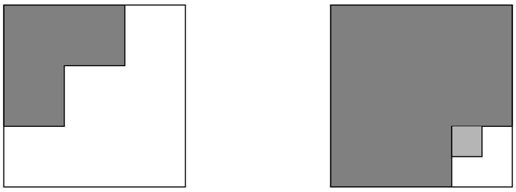
$$\tilde{p}_i(\mathbf{x}) = p_i(\mathbf{x}) + q_i(\mathbf{x})$$

for each $0 < i \leq m$. The support dimension of IP will be denoted as s.

To define IPRainbow, we will keep the layer 1 central maps the same and internally perturb the 2nd layer maps. Specifically, we will consider an internally perturbed 2nd layer homogeneous equation of the form:

$$f(\mathbf{x}) = \sum_{i=1}^{v_2}\sum_{j=1}^{v_2} \alpha_{ij}x_ix_j + \sum_{i=1}^{v_2}\sum_{j=v_2+1}^{n} \beta_{ij}x_ix_j + \sum_{i=v_2+1}^{v_2+s}\sum_{j=v_2+1}^{v_2+s} \mu_{ij}x_ix_j,$$

see Algorithm 1 in Appendix A. The matrix representations of the central maps are illustrated in Figure 4.



Layer 1 Rainbow Map.

Layer 2 Rainbow Map.

Fig. 4. The first layer maps remain the same as the unmodified Rainbow first layer maps. Now we consider a $s \times s$ submatrix of the oil times oil section of the second layer map that is nonzero and denoted as light gray.

Given an unmodified Rainbow public key, we know that for any $\mathbf{x} \in \mathbb{F}_q^n$ such that $T\mathbf{x} \in O_2$, $P(\mathbf{x}) = 0$. Now, given the IPModifier, the O_2 space is not a subspace of the kernel. Indeed, an O_2 vector is in the kernel of Q with probability approximately q^{-s}.

Inversion is similar to the process of inversion for Rainbow. One randomly assigns values to the first layer vinegar variables, $x_1, \ldots, x_{v_1}$ and uses the first layer maps to solve for the first layer oil variables, $x_{v_1+1}, \ldots, x_{v_2}$. To invert the second layer maps, they are evaluated at $x_1, \ldots, x_{v_2}$ to recover o_2 equations in o_2 variables. These equations are quadratic, however, only s variables occur in quadratic terms, thus by Gaussian elimination, we may recover a system of s quadratic equations in s variables whose resolution by standard Gröbner basis techniques allows for the remaining variables to be linearly solved, see Algorithm 2 in Appendix A.

4.2 Security Analysis

Simple Attack The simple attack of [4] remains applicable to IPRainbow, with some slight differences. Note that the matrix stucture of $D_{\mathbf{x}}$ remains the same as in the case of Rainbow. Thus, with probability roughly q^{-1} the linear map defined by $D_{\mathbf{x}}$ contains an O_2 vector $\mathbf{y}$ in its left kernel. For our new IPRainbow scheme, an oil vector in the kernel of $D_{\mathbf{x}}$ may not necessarily be in the kernel of the public key. Given that the second layer maps contain quadratic summands in s of the second layer oil variables, we expect the simple attack of [4] to proceed with probability roughly q^{-s-1} (See Lemma 1).

Lemma 1 *For sufficiently small s, the linear map $D_{\mathbf{x}}$ has an O_2 vector $\mathbf{y}$ in its left kernel that satisfies $P(\mathbf{y}) = \mathbf{0}$ with probability approximately q^{-s-1}.*

Proof. Let $\mathbf{y}$ be an O_2 vector satisfying $P(\mathbf{y}) = \mathbf{0}$. First, note that there are $\binom{s+1}{2}$ homogeneous quadratic monomials in the variables $y_{v_2+1}, \ldots, y_{v_2+s}$. Since the m unperturbed 2nd layer maps vanish at $\mathbf{y}$, the possibly nonzero terms of the perturbed second layer maps involve precisely these monomials. Thus, the probability that all of these monomials are zero (and hence $y_{v_2+i} = 0$ for i from 1 to s) is bounded below by the probability that this set of m equations has rank $\binom{s+1}{2}$, which is

$$\mathfrak{p}_r = \frac{\prod_{i=0}^{\binom{s+1}{2}-1} q^m - q^i}{q^{m\binom{s+1}{2}}} = \prod_{i=0}^{\binom{s+1}{2}-1} 1 - q^{i-m}.$$

Next, we work under the condition that the values $y_{v_2+1}, \ldots, y_{v_2+s}$ are all zero and determine the probability that such an O_2 vector is in the left kernel of $D_{\mathbf{x}}$. This probability is the same as the probability that there exists a nontrivial kernel vector of $D_{\mathbf{x}}$ restricted to this $o_2 - s$-dimensional subspace of O_2. This restricted linear map, which we may represent as a random $(o_2 - s) \times o_2$ matrix over $\mathbb{F}_q$, is of full rank with probability

$$\mathfrak{p}_k = \frac{\prod_{i=0}^{o_2-s-1} q^{o_2} - q^i}{q^{o_2(o_2-s)}} = \prod_{i=0}^{o_2-s-1} 1 - q^{i-o_2},$$

Finally, by Markov's inequality, the probability that there is at least a one-dimensional subspace W of O_2 in the left kernel of $D_{\mathbf{x}}$ such that $P(\mathbf{y}) = \mathbf{0}$ for all $\mathbf{y} \in W$ is then bounded by q^{-1} times the expected number of such vectors. We then note that the dominant term in the second expression is bounded by $(1 - \mathfrak{p}_r) + (1 - \mathfrak{p}_k)$, which is approximately q^{-s-1} for sufficiently small s. □

We further remark that the constraint on s being small is not very strict. Even if s is such that $1 < \mathfrak{p}_r, \mathfrak{p}_k$, there is still a rank condition that must be satisfied for such a vector to exist in the kernel of $D_{\mathbf{x}}$. Thus, we find that the above probability estimate is accurate even when $\binom{s+1}{2}$ is somewhat larger than m, a fact we have verified experimentally.

Rectangular MinRank Attack As is the case with Rainbow, the Simple attack of [4] can be combined with the Rectangular MinRank attack of [3]. As the attack still involves the finding a second layer oil variable and uses the property that such a vector satisfies the public equations, Lemma 1 applies, and we find that the complexity of the combined Rectangular MinRank attack costs a factor of approximately q^s times more for IPRainbow than for Rainbow. Thus, the complexity of the enhanced Rectangular MinRank Attack is given by

$$3q^{s+1}(n - m - 1)(o_2 + 1)\binom{m'}{r}^2 \binom{n - m + b - 3}{b}^2$$

field multiplications, where $m' \leq m$ and b are chosen to optimize the attack.

Intersection Attack In addition to the Simple attack and Rectangular MinRank attacks, Beullens also enhanced the Rainbow Band Separation attack of [12] and the tighter analysis of [23] with what he calls the Intersection Attack, see [3]. Once again, this attack relies on finding vectors in O_2 that satisfy the public polynomials. Therefore, once again, Lemma 1 applies and the complexity of these attacks is increased by a factor of about q^s. Even in the case that $n = 3m$, this attack is not the limiting attack.

4.3 Efficiency and Key Size

The complexity of the signing procedure is dominated by the complexity of the Gröbner basis algorithm used to solve the s-quadratic terms introduced in the IP modifier. Since the security of IPRainbow is exponential in s with base q, we choose $q = 257$ so that s can remain small for the fastest inversion. Table 5 compares the efficiency of IPRainbow with comparable UOV parameters. These estimates were computed with unoptimized implementations using the Magma Computer Algebra System,[1] see [5], on a 2.4 GHz Quad-Core Intel Core i5 processor.

[1] Any mention of commercial products does not indicate endorsement by NIST.

Scheme-(q, o_1, o_2, v, s)	Signing time	Verif. time	Key size	Sign. size	Security
UOV-$(257, 47, 0, 71, 0)$	0.750ms	0.370ms	330.2KB	118	144.5
IPRainbow-$(257, 32, 32, 32, 9)$	13700ms	0.370ms	298.2KB	96	145
IPRainbow-$(257, 32, 32, 36, 8)$	1976.5ms	0.380ms	323.4KB	100	144.3
IPRainbow-$(257, 32, 32, 38, 7)$	491ms	0.440ms	336.4KB	102	142.4
IPRainbow-$(257, 32, 36, 44, 6)$	127ms	0.510ms	430.6KB	112	143.1
UOV-$(257, 71, 0, 107, 0)$	138ms	1.190ms	1131.9KB	178	205.5
IPRainbow-$(257, 32, 42, 68, 9)$	16552ms	0.850ms	751.9KB	142	207.1
IPRainbow-$(257, 32, 48, 70, 8)$	4579ms	1.100ms	906.6KB	150	206.8
IPRainbow-$(257, 32, 48, 76, 7)$	987ms	1.020ms	980.4KB	156	206.9
IPRainbow-$(257, 32, 50, 84, 6)$	269ms	1.440ms	1137.4KB	166	206.9
UOV-$(257, 97, 0, 146, 0)$	5.240ms	4.630ms	2854.1KB	243	271
UOV-$(257, 98, 0, 147, 0)$	5.320ms	4.670ms	2931.3KB	245	275
IPRainbow-$(257, 36, 64, 112, 9)$	22026ms	2.390ms	2259.4KB	212	272
IPRainbow-$(257, 36, 64, 122, 8)$	29597ms	2.460ms	2477KB	222	271
IPRainbow-$(257, 36, 64, 135, 7)$	1123ms	5.300ms	2774.9KB	235	271.5
IPRainbow-$(257, 36, 66, 148, 6)$	298ms	5.280ms	3202.5KB	250	272.4

Fig. 5. Parameters targeting NIST security levels I, III and V.

We find that it is easy to achieve secure parameters of IPRainbow with smaller keys and smaller signatures. While it is possible to set parameters so that IPRainbow verification is faster than UOV, in all of the experiments we performed the signing times for these instances are very costly, to the point of possibly being disqualifying even for applications using offline signing. Still, it is important to note that our data seem a bit noisy and better implementation can make the relationship between key size and verification time tighter.

5 Conclusion

In the past year and a half, the new attacks from Beullens have significantly improved the cryptanalysis of Rainbow and have rendered it less efficient than UOV. As the motivation of Rainbow was originally to create a more efficient scheme based on the oil-vinegar structure, these attacks are particularly problematic for Rainbow.

Still, the appeal of schemes such as Rainbow is their ability to provide low cost for applications that are not dominated by investment in public key transmission. Such applications are naturally amenable to offline signing, so a penalty in the cost of inversion may be acceptable if there is sufficient benefit in verification speed or signature size.

As we have shown, the implementation of the IP modifier on Rainbow adds solid theoretical protection from these new attacks at the cost of a significant increase in the complexity of inversion. Our data indicate that it is indeed feasible to salvage an advantage in verification time, key size and signature size at the cost of additional signing time. The next step for future work is optimizing this construction and determining the market for such a product.

Acknowledgements

The authors would like to thank Kyle Salyer for his help on this project.

References

1. Magali Bardet, Pierre Briaud, Maxime Bros, Philippe Gaborit, Vincent Neiger, Olivier Ruatta, and Jean-Pierre Tillich. An algebraic attack on rank metric code-based cryptosystems. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part III*, volume 12107 of *Lecture Notes in Computer Science*, pages 64–93. Springer, 2020.
2. Magali Bardet, Maxime Bros, Daniel Cabarcas, Philippe Gaborit, Ray A. Perlner, Daniel Smith-Tone, Jean-Pierre Tillich, and Javier A. Verbel. Improvements of algebraic attacks for solving the rank decoding and minrank problems. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I*, volume 12491 of *Lecture Notes in Computer Science*, pages 507–536. Springer, 2020.
3. Ward Beullens. Improved cryptanalysis of UOV and rainbow. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I*, volume 12696 of *Lecture Notes in Computer Science*, pages 348–373. Springer, 2021.
4. Ward Beullens. Breaking rainbow takes a weekend on a laptop. 2022. `https://eprint.iacr.org/2022/214.pdf`.
5. Wieb Bosma, John Cannon, and Catherine Playoust. The magma algebra system i: The user language. *J. Symb. Comput.*, 24(3–4):235–265, October 1997.
6. Ryann Cartor and Daniel Smith-Tone. All in the c* family. *Des. Codes Cryptogr.*, 88(6):1023–1036, 2020.
7. Don Coppersmith. Solving homogeneous linear equations over gf(2) via block wiedemann algorithm. *Mathematics of Computation*, 62(205):333–350, 1994.
8. Nicolas T. Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir. Efficient algorithms for solving overdefined systems of multivariate polynomial equations. In Bart Preneel, editor, *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*, volume 1807 of *Lecture Notes in Computer Science*, pages 392–407. Springer, 2000.
9. Jintai Ding. A new variant of the matsumoto-imai cryptosystem through perturbation. In Feng Bao, Robert H. Deng, and Jianying Zhou, editors, *Public Key Cryptography - PKC 2004, 7th International Workshop on Theory and Practice*

in Public Key Cryptography, Singapore, March 1-4, 2004, volume 2947 of *Lecture Notes in Computer Science*, pages 305–318. Springer, 2004.

10. Jintai Ding, Ming-Shing Chen, Albrecht Petzoldt, Dieter Schmidt, and Bo-Yin Yang. Rainbow. NIST CSRC, 2020. `https://csrc.nist.gov/Projects/post-quantum-cryptography/round-3-submissions`.
11. Jintai Ding and Dieter Schmidt. Rainbow, a new multivariable polynomial signature scheme. In John Ioannidis, Angelos D. Keromytis, and Moti Yung, editors, *Applied Cryptography and Network Security, Third International Conference, ACNS 2005, New York, NY, USA, June 7-10, 2005, Proceedings*, volume 3531 of *Lecture Notes in Computer Science*, pages 164–175, 2005.
12. 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 Steven M. Bellovin, Rosario Gennaro, Angelos D. Keromytis, and Moti Yung, editors, *Applied Cryptography and Network Security, 6th International Conference, ACNS 2008, New York, NY, USA, June 3-6, 2008. Proceedings*, volume 5037 of *Lecture Notes in Computer Science*, pages 242–257, 2008.
13. Jean-Charles Faugère, Françoise Levy-dit-Vehel, and Ludovic Perret. Cryptanalysis of minrank. In David A. Wagner, editor, *Advances in Cryptology - CRYPTO 2008, 28th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2008. Proceedings*, volume 5157 of *Lecture Notes in Computer Science*, pages 280–296. Springer, 2008.
14. A. Kipnis and A. Shamir. Cryptanalysis of the hfe public key cryptosystem by relinearization. *Advances in Cryptology - CRYPTO 1999, Springer*, 1666:788, 1999.
15. Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced oil and vinegar signature schemes. In Jacques 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.
16. Aviad Kipnis and Adi Shamir. Cryptanalysis of the oil & vinegar signature scheme. In Hugo Krawczyk, editor, *Advances in Cryptology - CRYPTO '98, 18th Annual International Cryptology Conference, Santa Barbara, California, USA, August 23-27, 1998, Proceedings*, volume 1462 of *Lecture Notes in Computer Science*, pages 257–266. Springer, 1998.
17. Tsutomu Matsumoto and Hideki Imai. Public quadratic polynominal-tuples for efficient signature-verification and message-encryption. In Christoph G. Günther, editor, *Advances in Cryptology - EUROCRYPT '88, Workshop on the Theory and Application of of Cryptographic Techniques, Davos, Switzerland, May 25-27, 1988, Proceedings*, volume 330 of *Lecture Notes in Computer Science*, pages 419–453. Springer, 1988.
18. National Institute of Standards and Technology. Post-quantum cryptography, round 3 submissions, 2022.
19. Jacques Patarin. Cryptoanalysis of the matsumoto and imai public key scheme of eurocrypt'88. In *Advances in Cryptology - CRYPTO '95, 15th Annual International Cryptology Conference, Santa Barbara, California, USA, August 27-31, 1995, Proceedings*, volume 963 of *Lecture Notes in Computer Science*, pages 248–261. Springer, 1995.
20. Jacques Patarin. The oil and vinegar signature scheme. Presented at the Dagstuhl Workshop on Cryptography, September 1997.
21. Jacques Patarin and Louis Goubin. Trapdoor one-way permutations and multivariate polynominals. In Yongfei Han, Tatsuaki Okamoto, and Sihan Qing, editors, *Information and Communication Security, First International Conference,*

ICICS'97, Beijing, China, November 11-14, 1997, Proceedings, volume 1334 of *Lecture Notes in Computer Science*, pages 356–368. Springer, 1997.

22. Jacques Patarin, Louis Goubin, and Nicolas T. Courtois. C^{*}_{-+} and HM: variations around two schemes of t. matsumoto and h. imai. In Kazuo Ohta and Dingyi Pei, editors, *Advances in Cryptology - ASIACRYPT '98, International Conference on the Theory and Applications of Cryptology and Information Security, Beijing, China, October 18-22, 1998, Proceedings*, volume 1514 of *Lecture Notes in Computer Science*, pages 35–49. Springer, 1998.
23. Ray A. Perlner and Daniel Smith-Tone. Rainbow band separation is better than we thought. *IACR Cryptol. ePrint Arch.*, page 702, 2020.
24. Bo-Yin Yang and Jiun-Ming Chen. Theoretical analysis of XL over small fields. In Huaxiong Wang, Josef Pieprzyk, and Vijay Varadharajan, editors, *Information Security and Privacy: 9th Australasian Conference, ACISP 2004, Sydney, Australia, July 13-15, 2004. Proceedings*, volume 3108 of *Lecture Notes in Computer Science*, pages 277–288. Springer, 2004.

A Algorithms

Below are the key generation and central map inversion algorithms of IPRainbow.

Algorithm 1 IPRainbowKeyGen

```
**Input:** IPRainbow Parameters $(q, v_1, o_1, o_2, s)$
**Output:** IPRainbow Key Pair $(sk, pk)$
Set $m := o_1 + o_2, n := m + v_1$
$\mathcal{T}, \mathcal{U} \leftarrow GL(n, \mathbb{F}_q)$
$\mathcal{F} \leftarrow$ RainbowMap$(q, v_1, o_1, o_2)$
$\mathcal{Q} \leftarrow$ IPModifier$(s)$
$\mathcal{P} = \mathcal{T} \circ (\mathcal{F} + Q) \circ \mathcal{U}$
$sk = (\mathcal{T}, \mathcal{F}, \mathcal{Q}, \mathcal{U})$
$pk = \mathcal{P}$
**return** $(sk, pk)$
```

Algorithm 2 Inversion of IPRainbow Central Map

```
Input: IPRainbow central map $\mathcal{F} + \mathcal{Q} = (f_{v_1+1}, \dots, f_m)$, vector $\mathbf{x} \in \mathbb{F}^m$
Output: $\mathbf{y} \in \mathbb{F}^n$ with $\tilde{\mathcal{F}}(\mathbf{y}) = \mathbf{x}$
$y_1, \dots, y_{v_1} \xleftarrow{\$} \mathbb{F}_q$
$\tilde{f}_i := f_i(y_1, \dots, y_{v_1})$ for $i \in \{v_1 + 1, \dots, m\}$.
$y_{v_1+1}, \dots, y_{v_2} :=$ GaussElim($\tilde{f}_{v_1+1}, \dots, \tilde{f}_m$).
$\hat{f}_j := \tilde{f}_j(y_{v_1+1}, \dots, y_{v_2})$ for $j \in \{v_2 + 1, \dots, m\}$.
$g_1, \dots, g_s :=$ GaussElim($\hat{f}_{v_2+1}, \dots, \hat{f}_m$).
$y_{v_2+1}, \dots, y_n :=$ PolySolve($g_1, \dots, g_s$).
$\mathbf{y} := y_1, \dots, y_{v_1}, y_{v_1+1}, \dots, y_{v_2}, y_{v_2+1}, \dots, y_n$.
return $\mathbf{y}$.
```