A Practical Multivariate Blind Signature Scheme

Albrecht Petzoldt[1], Alan Szepieniec[2], Mohamed Saied Emam Mohamed[3]
albrecht.petzoldt@nist.gov, alan.szepieniec@esat.kuleuven.be,
mohamed@cdc.informatik.tu-darmstadt.de

[1] Kyushu University, Fukuoka, Japan & NIST, USA
[2] KU Leuven, ESAT/COSIC & imec, Belgium
[3] Technische Universität Darmstadt, Germany

Abstract. Multivariate Cryptography is one of the main candidates for creating post-quantum cryptosystems. Especially in the area of digital signatures, there exist many practical and secure multivariate schemes. However, there is a lack of multivariate signature schemes with special properties such as blind, ring and group signatures. In this paper, we propose a technique to transform the Rainbow multivariate signature schemes into a blind signature scheme. The resulting scheme satisfies the usual blindness criterion and a one-more-unforgeability criterion adapted to MQ signatures, produces short blind signatures and is very efficient.

Keywords: Multivariate Cryptography, Blind Signatures, Rainbow Signature Scheme

1 Introduction

Cryptographic techniques are an essential tool to guarantee the security of communication in modern society. Today, the security of nearly all of the cryptographic schemes used in practice is based on number theoretic problems such as factoring large integers and solving discrete logarithms. The best known schemes in this area are RSA [25], DSA [14] and ECC. However, schemes like these will become insecure as soon as large enough quantum computers are built. The reason for this is Shor's algorithm [29], which solves number theoretic problems like integer factorization and discrete logarithms in polynomial time on a quantum computer. Therefore, one needs alternatives to those classical public key schemes which are based on hard mathematical problems not affected by quantum computer attacks (so called post-quantum cryptosystems).
The increasing importance of research in this area has recently been emphasized by a number of authorities. For example, the american National Security Agency has recommended governmental organizations to change their security infrastructures from schemes like RSA to post-quantum schemes [17] and the National Institute of Standards and Technologies (NIST) is preparing to standardize these schemes [18]. According to NIST, multivariate cryptography is one of the main candidates for this standardization process. Multivariate schemes

are in general very fast and require only modest computational resources, which makes them attractive for the use on low cost devices like smart cards and RFID chips [5,6]. However, while there exist many practical multivariate standard signature schemes such as UOV [15], Rainbow [9] and Gui [24], there is a lack of multivariate signature schemes with special properties such as blind, ring, and group signatures.

Blind signature schemes allow a user, who is not in charge of the private signing key, to obtain a signature for a message d by interacting with the signer. The important point is that this signer, who holds the secret key, receives no information about the message d that is signed nor about the signature s that is created through the interaction. Nevertheless, anyone with access to the public verification key is capable of verifying that signature. Because of these unlinkability and public verifiability properties, blind signature schemes are an indispensable primitive in a host of privacy-preserving applications ranging from electronic cash to anonymous database access, e-voting, and anonymous reputation systems.

In this paper, we present a technique to transform Rainbow, a multivariate quadratic (MQ) signature scheme, into a blind signature scheme. This transformation is accomplished by joining the MQ signature scheme with the zero-knowledge MQ-based identification scheme of Sakumoto *et al.* [28]. The user queries the signer on a blinded version of the message to be signed; the signer's response is then combined with the blinding information in order to produce a non-interactive zero-knowledge proof of knowledge of a pre-image under the public verification key, which is a set of quadratic polynomials that contains the signer's public key in addition to a large random term. The only way the user can produce such a proof is by querying the signer at some point for a partial pre-image; however, because it is zero-knowledge, this proof contains no information on the message that was seen and signed by the signer, thus preventing linkage and ensuring the user's privacy.

We obtain one of the first multivariate signature schemes with special properties and more generally one of the very few candidates for establishing practical and secure post-quantum blind signatures. In terms of security requirements, our scheme satisfies the usual blindness notion, but an adapted one-more-unforgeability one which we call *universal*-one-more-unforgeability. This change is justified by the observation that the usual one-more-unforgeability notion generalizes *existential* unforgeability for regular signatures; however, MQ signatures can only be shown to offer *universal* unforgeability and hence require a universal one-more-unforgeability generalization. While our technique applies to some other MQ signature schemes also, we instantiate our scheme with the Rainbow signature scheme and propose parameters targeting various levels of security.

The rest of this paper is organized as follows. Section 2 recalls the basic concepts of blind signatures and discusses the basic security notions. In Section 3 we recall the basic concepts of multivariate cryptography and review the Rainbow signature scheme, Sakumoto's multivariate identification scheme [28], and

its transformation into a digital signature scheme due to Hülsing [12]. Section 4 presents our technique to extend multivariate signature schemes such as Rainbow to blind signature schemes, while Section 5 discusses the security of our construction. In Section 6 we give concrete parameter sets and analyze the efficiency of our scheme. Furthermore, in this section, we describe a proof of concept implementation of our scheme and compare it with other existing (classical and post-quantum) blind signature schemes. Finally, Section 7 concludes the paper.

2 Blind Signatures

Blind signature schemes as proposed by David Chaum in [3] allow a user, who is not in charge of the private signing key, to obtain a signature for a message d on behalf of the owner of the private key (called the signer). The key point hereby is that the signer gets no information about the content of the message d.
The signature generation process of a blind signature scheme is an interactive process between the user and the signer. In the first step, the user computes from the message d a blinded message $d^\star$ and sends it to the signer. The signer uses his private key to generate a signature $\sigma^\star$ for the message $d^\star$ and sends it back to the signer. Due to certain homomorphic properties in the inner structure of the blind signature scheme, the user is able to compute from $\sigma^\star$ a valid signature σ for the original message d. The receiver of a signed message can check the authenticity of the signature σ in the same way as in the case of a standard signature scheme. Figure 1 shows a graphical illustration of the signature generation process of a blind signature scheme.
Formally, a blind signature scheme $\mathcal{BS}$ is a three-tuple, consisting of two poly-

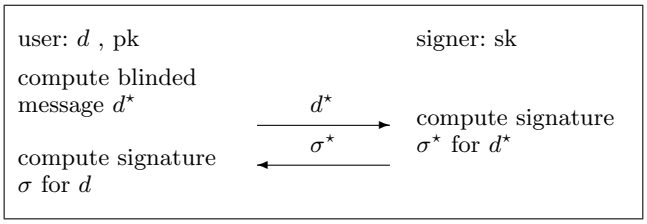

Fig. 1. Signature Generation Process of a Blind Signature Scheme

nomial time algorithms `KeyGen` and `Verify` and an interactive signing protocol `Sign` [13].

– `KeyGen`(1^κ): The probabilistic algorithm `KeyGen` takes as input a security parameter κ and outputs a key pair (sk, pk) of the blind signature scheme.

– `Sign`: The signature generation step is an interactive protocol between the `User`, who gets as input a message d and a public key pk and the `Signer` who is given the pair (pk, sk) generated by algorithm `KeyGen`. At the end of the protocol, the `Signer` outputs either "completed" or "non-completed", while the user outputs either "failed" or a signature σ.
– `Verify`$((d, \sigma), pk)$: The deterministic algorithm `Verify` takes as input a message/signature pair (d, σ) and the public key pk. It outputs **TRUE**, if σ is a valid signature for the message d and **FALSE** otherwise.

In the following, we assume the *correctness* of the blind signature scheme $\mathcal{BS}$: If both the `User` and the `Signer` follow the protocol, the `Signer` outputs always "completed", independently of the message d and the output (sk, pk) of the algorithm `KeyGen`. Similarly, the `User` always outputs a signature σ and we have

$$\Pr[\texttt{Verify}((d, \sigma), pk) = \textbf{TRUE}] = 1.$$

The basic security criteria of a blind signature scheme are Blindness and One-More-Unforgeability.

– **Blindness**: By signing the blinded message $d^\star$, the signer of a message gets no information about the content of the message to be signed nor about the final blind signature σ. More formally, blindness can be defined using the following security game.

Game[Blindness]:

1. The adversary $\mathcal{A}$ uses the algorithm `KeyGen` to generate a key pair (sk, pk) of the blind signature scheme. The public key pk is made public, while $\mathcal{A}$ keeps sk as his private key.
2. The adversary $\mathcal{A}$ outputs two messages d_0 and d_1, which might depend on sk and pk.
3. Let u_0 and u_1 be users with access to the public key pk but not to the secret key sk. For a random bit b that is unknown to $\mathcal{A}$, user u_0 is given the message d_b, while the message d_{1-b} is sent to user u_1. Both users engage in the interactive signing protocol (with $\mathcal{A}$ as signer), obtaining blind signatures σ_0 and σ_1 for the messages d_0 and d_1. The message/signature pairs (d_0, σ_0) and (d_1, σ_1) are given to the adversary $\mathcal{A}$.
4. $\mathcal{A}$ outputs a bit $\bar{b}$. He wins the game, if and only if $\bar{b} = b$ holds.

The blind signature scheme $\mathcal{BS}$ is said to fulfill the blindness property, if the advantage

$$\mathrm{Adv}_{\mathcal{BS}}^{\mathsf{blindness}}(\mathcal{A}) = |2 \cdot \Pr[b' = b] - 1|$$

for every PPT adversary $\mathcal{A}$ is negligible in the security parameter.

– **One-More-Unforgeability**: Even after having successfully completed L rounds of the interactive signing protocol, an adversary $\mathcal{A}$ not in charge of the private key sk cannot forge another valid blind signatures for a given message. More formally, we can define One-More-Unforgeability using the following game.

Game [Universal-One-More-Unforgeability]

1. The algorithm `KeyGen` is used to generate a key pair (sk, pk). The public key pk is given to the adversary $\mathcal{A}$, while sk is kept secret by the challenger.
2. The adversary $\mathcal{A}$ engages himself in polynomially many interactive signing protocols with different instances of `Signer`. Let L be the number of cases in which the `Signer` outputs *completed*.
3. $\mathcal{A}$ outputs a list $\mathcal{L}$ of L message / signature pairs. The challenger checks if all the message / signature pairs are valid and pairwise distinct.
4. The challenger outputs a message $d^\star$ not contained in the list $\mathcal{L}$. The adversary wins the game, if he is able to generate a valid blind signature σ for the message $d^\star$, i.e. if $\texttt{Verify}((d^\star, \sigma), pk) = \textbf{TRUE}$ holds.

The blind signature scheme $\mathcal{BS}$ is said to provide the One-More-Unforgeability property, if the success probability

$$\Pr[\mathcal{A} \text{ wins}]$$

is, for any PPT adversary $\mathcal{A}$, negligible in the security parameter.
We note that this formalism is different from the standard security game for blindness, where the adversary is allowed to choose his own message but is required to forge at least $L + 1$ valid and distinct signatures. We choose to restrict the adversary's choice to accurately reflect the similar lack of choice in the standard security model for MQ signatures: *universal* unforgeability as opposed to *existential* unforgeability.
In the existential unforgeability game, the adversary wins whenever he is capable of producing any forgery, regardless of which message is signed. In contrast, in the universal unforgeability game the adversary obtains a message from the challenger and the adversary only wins if he can forge a signature for that specific message. Nevertheless, the universal adversary is allowed to query signatures after obtaining the target message; just not signatures on the same message. The reason why our formalism of universal-one-more-unforgeability does not allow blind-signature queries after delivering the target message to the adversary is precisely because the signature-queries are blind: the challenger should not be able to tell if it is the target message that is being blind-signed or something else.

3 Multivariate Cryptography

The basic objects of multivariate cryptography are systems of multivariate quadratic polynomials. Their security is based on the **MQ Problem**: Given m multivariate quadratic polynomials $p^{(1)}(\mathbf{x}), \ldots, p^{(m)}(\mathbf{x})$ in n variables $x_1, \ldots, x_n$, find

a vector $\bar{\mathbf{x}} = (\bar{x}_1, \ldots, \bar{x}_n)$ such that $p^{(1)}(\bar{\mathbf{x}}) = \ldots = p^{(m)}(\bar{\mathbf{x}}) = 0$.

The MQ problem is proven to be NP-hard even for quadratic polynomials over the field GF(2) [11]. Moreover, it is widely assumed as well as experimentally validated that solving *random* instances of the MQ problem (with $m \approx n$) is a hard task, see for example [31].
To build a public key cryptosystem on the basis of the MQ problem, one starts with an easily invertible quadratic map $\mathcal{F} : \mathbb{F}^n \to \mathbb{F}^m$ (central map). To hide the structure of $\mathcal{F}$ in the public key, one composes it with two invertible affine (or linear) maps $\mathcal{S} : \mathbb{F}^m \to \mathbb{F}^m$ and $\mathcal{T} : \mathbb{F}^n \to \mathbb{F}^n$. The *public key* of the scheme is therefore given by $\mathcal{P} = \mathcal{S} \circ \mathcal{F} \circ \mathcal{T} : \mathbb{F}^n \to \mathbb{F}^m$. The *private key* consists of $\mathcal{S}$, $\mathcal{F}$ and $\mathcal{T}$ and therefore allows to invert the public key.

Note: Due to the above construction, the security of multivariate schemes is not only based on the MQ-Problem, but also on the EIP-Problem ("Extended Isomorphism of Polynomials") of finding the decomposition of $\mathcal{P}$.
In this paper we concentrate on multivariate signature schemes. The standard signature generation and verification process of a multivariate signature scheme works as shown in Figure 2.

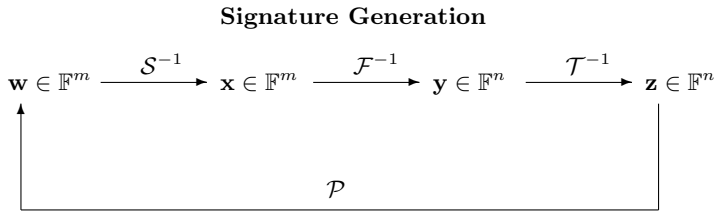

Fig. 2. Standard workflow of multivariate signature schemes

Signature generation: To sign a message $\mathbf{w} \in \mathbb{F}^m$, one computes recursively $\mathbf{x} = \mathcal{S}^{-1}(\mathbf{w}) \in \mathbb{F}^m$, $\mathbf{y} = \mathcal{F}^{-1}(\mathbf{x}) \in \mathbb{F}^n$ and $\mathbf{z} = \mathcal{T}^{-1}(\mathbf{y})$. The signature of the message $\mathbf{w}$ is $\mathbf{z} \in \mathbb{F}^n$. Here, $\mathcal{F}^{-1}(\mathbf{x})$ means finding one (of possibly many) preimage of $\mathbf{x}$ under the central map $\mathcal{F}$.

Verification: To check the authenticity of a signature $\mathbf{z} \in \mathbb{F}^n$, one simply computes $\mathbf{w}' = \mathcal{P}(\mathbf{z}) \in \mathbb{F}^m$. If $\mathbf{w}' = \mathbf{w}$ holds, the signature is accepted, otherwise rejected.

3.1 The Rainbow Signature Scheme

The Rainbow signature scheme [9] is one of the most promising and best studied multivariate signature schemes. The scheme can be described as follows:

Let $\mathbb{F} = \mathbb{F}_q$ be a finite field with q elements, $n \in \mathbb{N}$ and $v_1 < v_2 < \ldots < v_\ell < v_{\ell+1} = n$ be a sequence of integers. We set $m = n - v_1$, $O_i = \{v_i + 1, \ldots, v_{i+1}\}$ and $V_i = \{1, \ldots, v_i\}$ $(i = 1, \ldots, \ell)$.

Key Generation: The *private key* of the scheme consists of two invertible affine maps $\mathcal{S} : \mathbb{F}^m \to \mathbb{F}^m$ and $\mathcal{T} : \mathbb{F}^n \to \mathbb{F}^n$ and a quadratic map $\mathcal{F}(\mathbf{x}) = (f^{(v_1+1)}(\mathbf{x}), \ldots, f^{(n)}(\mathbf{x})) : \mathbb{F}^n \to \mathbb{F}^m$. The polynomials $f^{(i)}$ $(i = v_1 + 1, \ldots, n\}$ are of the form

$$f^{(i)} = \sum_{k,l \in V_j} \alpha_{k,l}^{(i)} \cdot x_k \cdot x_l + \sum_{k \in V_j, l \in O_j} \beta_{k,l}^{(i)} \cdot x_k \cdot x_l + \sum_{k \in V_j \cup O_j} \gamma_k^{(i)} \cdot x_k + \eta^{(i)} \quad (1)$$

with coefficients randomly chosen from $\mathbb{F}$. Here, j is the only integer such that $i \in O_j$. The *public key* is the composed map $\mathcal{P} = \mathcal{S} \circ \mathcal{F} \circ \mathcal{T} : \mathbb{F}^n \to \mathbb{F}^m$.

Signature Generation: To generate a signature for a document $\mathbf{w} \in \mathbb{F}^m$, we compute recursively $\mathbf{x} = \mathcal{S}^{-1}(\mathbf{w}) \in \mathbb{F}^m$, $\mathbf{y} = \mathcal{F}^{-1}(\mathbf{x}) \in \mathbb{F}^n$ and $\mathbf{z} = \mathcal{T}^{-1}(\mathbf{y})$. Here, $\mathcal{F}^{-1}(\mathbf{x})$ means finding one (of approximately q^{v_1}) pre-image of $\mathbf{x}$ under the central map $\mathcal{F}$. This is done as shown in Algorithm 1.

Algorithm 1 Inversion of the Rainbow central map

Input: Rainbow central map $\mathcal{F}$, vector $\mathbf{x} \in \mathbb{F}^m$.
Output: vector $\mathbf{y} \in \mathbb{F}^n$ such that $\mathcal{F}(\mathbf{y}) = \mathbf{x}$.

1: Choose random values for the variables $y_1, \ldots, y_{v_1}$ and substitute these values into the polynomials $f^{(i)}$ $(i = v_1 + 1, \ldots, n)$.
2: **for** $k = 1$ to ℓ **do**
3: Perform Gaussian Elimination on the polynomials $f^{(i)}$ $(i \in O_k)$ to get the values of the variables y_i $(i \in O_k)$.
4: Substitute the values of y_i $(i \in O_k)$ into the polynomials $f^{(i)}$, $i \in \{v_{k+1} + 1, \ldots, n\}$.
5: **end for**

It might happen that one of the linear systems in step 3 of the algorithm does not have a solution. In this case one has to choose other values for $y_1, \ldots, y_{v_1}$ and start again. The signature of the document $\mathbf{w}$ is $\mathbf{z} \in \mathbb{F}^n$.

Signature Verification: To verify the authenticity of a signature $\mathbf{z} \in \mathbb{F}^n$, one simply computes $\mathbf{w}' = \mathcal{P}(\mathbf{z}) \in \mathbb{F}^m$. If $\mathbf{w}' = \mathbf{w}$ holds, the signature is accepted, otherwise rejected.

3.2 The MQ-based Identification Scheme

In [28] Sakumoto *et al.* proposed an identification scheme based on multivariate polynomials. There exist two versions of the scheme: a 3-pass and a 5-pass variant. In this section we introduce the 5-pass variant.
The scheme uses a system $\mathcal{P}$ of m multivariate quadratic polynomials in n variables as a public parameter. The prover chooses a random vector $\mathbf{s} \in \mathbb{F}^n$ as his secret key and computes the public key $\mathbf{v} \in \mathbb{F}^m$ by $\mathbf{v} = \mathcal{P}(\mathbf{s})$.
To prove his identity to a verifier, the prover performs several rounds of the interactive protocol shown in Figure 3.
Here,

$$\mathcal{G}(\mathbf{x}, \mathbf{y}) = \mathcal{P}(\mathbf{x} + \mathbf{y}) - \mathcal{P}(\mathbf{x}) - \mathcal{P}(\mathbf{y}) + \mathcal{P}(\mathbf{0}) \tag{2}$$

is the polar form of the system $\mathcal{P}$.
The scheme is a zero-knowledge argument of knowledge for a solution of the system $\mathcal{P}(\mathbf{x}) = \mathbf{v}$.
The knowledge error per round is $\frac{1}{2} + \frac{1}{2q}$. To decrease the impersonation probability below $2^{-\eta}$, one therefore needs to perform $r = \lceil \frac{-\eta}{\log_2(1/2+1/2q)} \rceil$ rounds of the protocol. For identification purposes, $\eta \approx 30$ may be sufficient, but for signatures we require η to be at least as large as the security level.

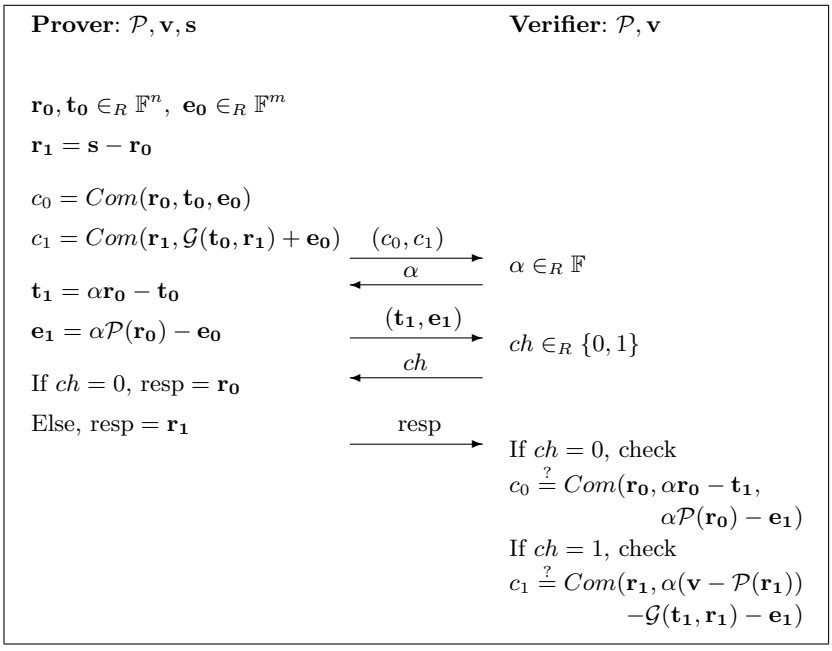

Fig. 3. The 5-pass MQ identification scheme of Sakumoto *et al.* [28].

3.3 The MQDSS signature scheme

In [12], Hülsing et al. developed a technique to transform (2n+1) pass identification schemes into signature schemes. The technique can be used to transform the above described 5-pass multivariate identification scheme into an EU-CMA secure signature scheme.
To generate an MQDSS signature for a message d, the signer produces a transcript of the above identification protocol over r rounds. The challenges $\alpha_1, \ldots, \alpha_r$ and $ch_1, \ldots, ch_r$ are hereby computed from the message d and the commitments (using a publicly known hash function $\mathcal{H}$). Therefore, the signature has the form

$$\sigma = (c_{0,1}, c_{1,1}, \ldots, c_{0,r}, c_{1,r}, t_{1,1}, e_{1,1}, \ldots, t_{1,r}, e_{1,r}, \text{resp}_1, \ldots, \text{resp}_r).$$

To check the authenticity of a signature σ, the verifier parses σ into its components, uses the commitments to compute the challenges α_i and ch_i $(i = 1, \ldots, r)$ and checks the correctness of the responses resp_i as shown in Figure 3 (for $i = 1, \ldots, r$).

4 Our Blind Signature Scheme

In this section we present MBSS, construction for blind signatures based on Rainbow. We chose to restrict our attention to Rainbow due to its short signatures and good performance. Moreover, the key sizes of Rainbow are acceptable and can be further reduced by the technique of Petzoldt *et al.* [22].
Nevertheless, our technique applies to any MQ signature scheme relying on the construction of Fig. 2, *i.e.*, relying on the hiding of a trapdoor to a quadratic map behind linear or affine transforms. As the other MQ signature schemes rely on the same construction, our technique applies to those cryptosystems as well. We do not use any property of Rainbow that is not shared by, *e.g.*, HFEv^- [24], pC^* [7], or UOV [15]. The exceptions are the MQ signature schemes that do not have the construction of Fig. 2, such as Quartz [19] and MQDSS [12].

4.1 The Basic Idea

The public key of our scheme consists of two multivariate quadratic systems $\mathcal{P} : \mathbb{F}^n \to \mathbb{F}^m$ and $\mathcal{R} : \mathbb{F}^m \to \mathbb{F}^m$. Hereby, $\mathcal{P}$ is the Rainbow public key, while $\mathcal{R}$ is a random system. The signer's private key allows him to invert the system $\mathcal{P}$ using the algorithm from Section 3.1.
In order to obtain a blind signature for a message (hash value) $\mathbf{w} \in \mathbb{F}^m$, the user chooses randomly a vector $\mathbf{z}^\star \in \mathbb{F}^m$, computes $\tilde{\mathbf{w}} = \mathbf{w} - \mathcal{R}(\mathbf{z}^\star)$ and sends $\tilde{\mathbf{w}}$ to the signer. The signer uses his private key to compute a signature $\mathbf{z}$ for the message $\tilde{\mathbf{w}}$ and sends it to the user. Therefore, the user obtains a solution $(\mathbf{z}, \mathbf{z}^\star)$ of the system $\mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) = \mathbf{w}$. However, the user can not publish $(\mathbf{z}, \mathbf{z}^\star)$ as his signature for the document $\mathbf{w}$ since this would destroy the blindness of the scheme. Instead, the user has to prove knowledge of a solution to the system $\mathcal{P}(\mathbf{x}_1)+\mathcal{R}(\mathbf{x}_2) = \mathbf{w}$ using a zero knowledge protocol. We use the MQDSS technique (see Section 3.3) for this proof.

4.2 Description of the Scheme

In this section we give a detailed description of our blind signature scheme. As every blind signature scheme, MBSS consists of three algorithms *KeyGen*, *Sign* and *Verify*, where *Sign* is an interactive protocol between user and signer.

Parameters: Finite field $\mathbb{F}$, integers m, n and r (depending on a security parameter κ). r hereby determines, how many rounds of the identification scheme are performed during the generation of a signature.

Key Generation: The signer chooses randomly a Rainbow private key (consisting of two affine maps $\mathcal{S} : \mathbb{F}^m \to \mathbb{F}^m$ and $\mathcal{T} : \mathbb{F}^n \to \mathbb{F}^n$ and a secret central map $\mathcal{F} : \mathbb{F}^n \to \mathbb{F}^m$). He computes the public key $\mathcal{P}$ as $\mathcal{P} = \mathcal{S} \circ \mathcal{F} \circ \mathcal{T} : \mathbb{F}^n \to \mathbb{F}^m$ (see Section 3.1) and uses a CSPRNG to generate the system $\mathcal{R} = \texttt{CSPRNG}(\mathcal{P}) : \mathbb{F}^m \to \mathbb{F}^m$. The *public key* of our blind signature scheme is the pair $(\mathcal{P}, \mathcal{R})$, the signer's *private key* consists of $\mathcal{S}, \mathcal{F}$ and $\mathcal{T}$. However, since $\mathcal{R}$ can be computed from the system $\mathcal{P}$, it is not necessary to publish $\mathcal{R}$ (if the CSPRNG in use is publicly accessible).

Signature Generation: The interactive signature generation process of our blind signature scheme can be described as follows: To get a signature for the message d with hash value $\mathcal{H}(d) = \mathbf{w} \in \mathbb{F}^m$, the user chooses randomly a vector $\mathbf{z}^\star \in \mathbb{F}^m$. He computes $\mathbf{w}^\star = \mathcal{R}(\mathbf{z}^\star) \in \mathbb{F}^m$ and sends $\tilde{\mathbf{w}} = \mathbf{w} - \mathbf{w}^\star \in \mathbb{F}^m$ to the signer. The signer uses his private key $(\mathcal{S}, \mathcal{F}, \mathcal{T})$ to compute a signature $\mathbf{z} \in \mathbb{F}^n$ such that $\mathcal{P}(\mathbf{z}) = \tilde{\mathbf{w}}$ and sends $\mathbf{z}$ back to the user, who therefore obtains a solution $(\mathbf{z}, \mathbf{z}^\star)$ of the system $\bar{\mathcal{P}}(\mathbf{x}) = \mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) = \mathbf{w}$.
To prove this knowledge to the verifier in a zero knowledge way, the user generates an MQDSS signature for the message $\mathbf{w}$. As the public parameter of the scheme he hereby uses the system $\bar{\mathcal{P}}(\mathbf{x}) = \mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2)$, which is a system of m quadratic equations in $n+m$ variables. Furthermore, $\mathcal{G}(\mathbf{x}, \mathbf{y})$ is the polar form of the system $\bar{\mathcal{P}}$, i.e. $\mathcal{G}(\mathbf{x}, \mathbf{y}) = \bar{\mathcal{P}}(\mathbf{x} + \mathbf{y}) - \bar{\mathcal{P}}(\mathbf{x}) - \bar{\mathcal{P}}(\mathbf{y}) + \bar{\mathcal{P}}(\mathbf{0})$. In particular, the user performs the following steps.

1. Use a publicly known hash function $\mathcal{H}$ to compute $\mathcal{C} = \mathcal{H}(\mathcal{P}||\mathbf{w})$ and $\mathcal{D} = \mathcal{H}(\mathcal{C}||\mathbf{w})$.
2. Choose random values for $\mathbf{r}_{0,1}, \ldots, \mathbf{r}_{0,r}, \mathbf{t}_{0,1}, \ldots, \mathbf{t}_{0,r} \in \mathbb{F}^{m+n}$, $\mathbf{e}_{0,1}, \ldots, \mathbf{e}_{0,r} \in \mathbb{F}^m$, set $\mathbf{r}_{1,i} = (\mathbf{z}||\mathbf{z}^\star) - \mathbf{r}_{0,i}$ $(i = 1, \ldots, r)$ and compute the commitments

$$c_{0,i} = Com(\mathbf{r}_{0,i}, \mathbf{t}_{0,i}, \mathbf{e}_{0,i}) \text{ and}$$
$$c_{1,i} = Com(\mathbf{r}_{1,i}, \mathcal{G}(\mathbf{t}_{0,i}, \mathbf{r}_{1,i}) - \mathbf{e}_{0,i}) \quad (i = 1, \ldots, r).$$

   Set COM $= (c_{0,1}, c_{1,1}, c_{0,2}, c_{1,2}, \ldots, c_{0,r}, c_{1,r})$.
3. Derive the challenges $\alpha_1, \ldots, \alpha_r \in \mathbb{F}$ from $(\mathcal{D}, COM)$.
4. Compute $\mathbf{t}_{1,i} = \alpha_i \cdot \mathbf{r}_{0,i} - \mathbf{t}_{0,i} \in \mathbb{F}^{m+n}$ and $\mathbf{e}_{1,i} = \alpha_i \cdot \bar{\mathcal{P}}(\mathbf{r}_{0,i}) - \mathbf{e}_{0,i}$ $(i = 1, \ldots, r)$. Set $Rsp_1 = (\mathbf{t}_{1,1}, \mathbf{e}_{1,1}, \ldots, \mathbf{t}_{1,r}, \mathbf{e}_{1,r})$.
5. Derive the challenges $(ch_1, \ldots, ch_r)$ from $(\mathcal{D}, COM, Rsp_1)$.
6. Set $Rsp_2 = (\mathbf{r}_{ch_1,1}, \ldots, \mathbf{r}_{ch_r,r})$.

7. The blind signature σ for the message $\mathbf{w} \in \mathbb{F}^m$ is given by

$$\sigma = (\mathcal{C}, COM, Rsp_1, Rsp_2).$$

The length of the blind signature σ is given by

$$|\sigma| = 1 \cdot |\text{hash value}| + 2r \cdot |\text{Commitment}| + r \cdot (2n + 3m) \ \ \mathbb{F}-\text{elements}.$$

Figure 4 shows the full protocol for obtaining a blind signature.
Signature Verification: To check the authenticity of a blind signature σ for a message d with hash value $\mathbf{w} \in \mathbb{F}^m$, the verifier parses σ into its components and computes $\mathcal{D} = \mathcal{H}(\mathcal{C}||\mathbf{w})$. He derives the challenges $\alpha_i \in \mathbb{F}$ from $(\mathcal{D}, COM)$ and ch_i from $(\mathcal{D}, COM, Rsp_1)$ $(i = 1, \ldots, r)$.
Finally, he parses COM into $(c_{0,1}, c_{1,1}, c_{0,2}, c_{1,2}, \ldots, c_{0,r}, c_{1,r}), Rsp_1$ into $\mathbf{t}_1, \mathbf{e}_1, \ldots, \mathbf{t}_r, \mathbf{e}_r$ and Rsp_2 into $\mathbf{r}_1, \ldots, \mathbf{r}_r$ and checks if, for all $i = 1, \ldots, r$, $\mathbf{r}_i$ is a correct response to ch_i with respect to COM, $\mathbf{t}_i$ and $\mathbf{e}_i$, i.e.

$$\begin{aligned} c_{0,i} &\stackrel{?}{=} Com(\mathbf{r_i}, \alpha_i \cdot \mathbf{r_i} - \mathbf{t_i}, \alpha_i \cdot \mathcal{P}(\mathbf{r_i}) - \mathbf{e_i}) \ \ (\text{for } ch_i = 0) \\ c_{1,i} &\stackrel{?}{=} Com(\mathbf{r_i}, \alpha_i \cdot (\mathbf{w} - \mathcal{P}(\mathbf{r_i})) - \mathcal{G}(\mathbf{t_i}, \mathbf{r_i}) - \mathbf{e_i}) \ \ (\text{for } ch_i = 1). \end{aligned} \tag{3}$$

If all of these tests are fulfilled, the blind signature σ is accepted, otherwise rejected.

Note: As the resulting blind signature depends on the randomness sampled for generating the zero-knowledge proof, there may be many signatures associated to one tuple $(\mathbf{z}, \mathbf{z}^\star)$. To prevent a malicious user from reusing the same preimage to $\mathcal{P}(\bar{\mathbf{x}}_1) + \mathcal{R}(\bar{\mathbf{x}}_2)$, two signatures to messages d_1, d_2 are considered *essentially* different whenever $\mathbf{w}_1 = \mathcal{H}(d_1) \neq \mathbf{w}_2 = \mathcal{H}(d_2)$. In other words, the zero-knowledge proof is taken into account for validity but not for distinctness.

4.3 Reducing the Signature Length

In this section we present a technique to reduce the length of the blind signature σ, which was already mentioned in [28] and [12].
Instead of including all of the commitments $c_{0,1}, c_{1,1}, \ldots, c_{0,r}, c_{1,r}$ into the signature, we just transmit $COM = \mathcal{H}(c_{0,1}||c_{1,1} \ldots c_{0,r}||c_{1,r})$. However, in this scenario, we have to add $(c_{1-ch_1,1}, \ldots, c_{1-ch_r,r})$ to Rsp_2. In the verification process, the verifier recovers $(c_{ch_1,1}, \ldots, c_{ch_r,r})$ by equation (3) and checks if

$$COM \stackrel{?}{=} \mathcal{H}(c_{0,1}, c_{1,1}, \ldots, c_{0,r}, c_{1,r})$$

is fulfilled. By doing so, we can reduce the length of the blind signature σ to

$$|\sigma| = 2 \cdot |\text{hash value}| + r \cdot (2n + 3m) \ \mathbb{F} \text{ elements} + r \cdot |\text{Commitment}| \ .$$

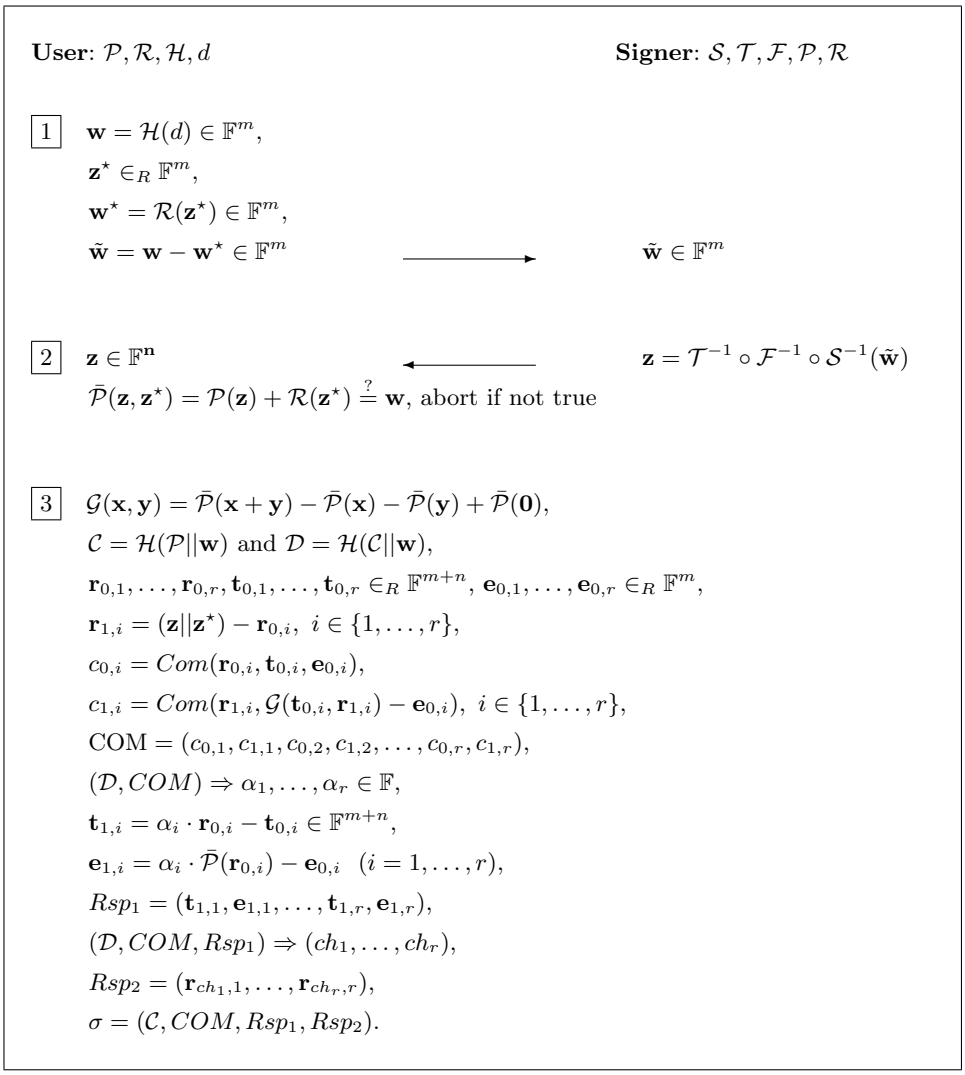

Fig. 4. Our blind signing protocol.

4.4 Correctness

Theorem 1. *Blind signatures generated by honest participants in the protocols of our multivariate blind signature scheme will be accepted with probability 1.*

Proof. The proof consists out of two steps. In the first step we show that, at the end of the interactive process, the user obtains a solution $(\mathbf{z}, \mathbf{z}^\star)$ of the system $\mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) = \mathbf{w}$. This can be seen as follows. In the course of the interactive protocol, the (honest) user chooses randomly a vector $\mathbf{z}^\star$, computes $\mathbf{w}^\star = \mathcal{R}(\mathbf{z}^\star)$ and $\tilde{\mathbf{w}} = \mathbf{w} - \mathbf{w}^\star$ and sends $\tilde{\mathbf{w}}$ to the signer. The (honest) signer uses his private key to compute a vector $\mathbf{z}$ such that $\mathcal{P}(\mathbf{z}) = \tilde{\mathbf{w}}$. Altogether, we get $\mathcal{P}(\mathbf{z}) + \mathcal{R}(\mathbf{z}^\star) = \tilde{\mathbf{w}} + \mathbf{w}^\star = \mathbf{w} - \mathbf{w}^\star + \mathbf{w}^\star = \mathbf{w}$, which means that $(\mathbf{z}, \mathbf{z}^\star)$ is indeed a solution of the public system $\bar{\mathcal{P}}(\mathbf{x}) = \mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2)$.
In the second step we simply use the correctness proof of the MQDSS [12] to show that an MQDSS signature produced by an honest signer knowing a solution to the public system $\bar{\mathcal{P}}$ is, by an honest verifier, accepted with probability 1.

5 Security

In this section, we analyze the security of our construction, assuming abstractly that Rainbow is secure. (For a concrete security analysis of the underlying Rainbow scheme we refer to [21].) For this, we have to show the blindness and one-more-unforgeability of the derived scheme.

5.1 Blindness

Theorem 2. *Assume that the distribution of $\mathcal{R}(\mathbf{x})$ for uniform $\mathbf{x} \in \mathbb{F}_q^m$ is computationally indistinguishable from uniform, and assume that a perfectly hiding commitment scheme is used. Then our multivariate blind signature scheme provides blindness against any computationally bounded adversary. In particular, for all PPT adversaries $\mathcal{A}$, their advantage in the blindness game (of Section 2) for our scheme is at most negligible:*

$$\forall \mathcal{A} \,.\, \mathrm{Adv}^{\mathsf{blindness}}_{\mathcal{MBSS}}(\mathcal{A}) \leq \mathsf{negl} \ .$$

Proof. The adversary has to link $\tilde{\mathbf{w}}$ from one interaction, to the pair (d, σ) from another interaction. Due to the perfect zero-knowledge property of the perfectly hiding commitment scheme, σ contains no information about the solution $(\mathbf{z}, \mathbf{z}^\star)$ and hence no information about $\mathcal{R}(\mathbf{z}^\star)$ or $\mathcal{P}(\mathbf{z})$. Therefore the adversary's task is equivalent linking $\tilde{\mathbf{w}}$ to d, since knowledge of σ gives him no advantage. However, $\mathbf{z}^\star$ is chosen uniformly at random and so $\mathcal{R}(\mathbf{z}^\star)$ is computationally indistinguishable from uniform. As a result, the blinded message $\tilde{\mathbf{w}} = \mathbf{w} - \mathcal{R}(\mathbf{z}^\star)$ is computationally indistinguishable from uniform and no polynomial-time adversary can compute any predicate of $\mathbf{w}$ from $\tilde{\mathbf{w}}$ with more than a negligible success probability. This includes the predicate $\mathcal{H}(d) \stackrel{?}{=} \mathbf{w}$ or any similar predicate that would allow the adversary to link $\tilde{\mathbf{w}}$ to d.

5.2 Universal One-More-Unforgeability

Theorem 3. *If Rainbow is secure and if finding a solution* $(\mathbf{x}_1, \mathbf{x}_2)$ *to* $\mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) = \mathbf{0}$ *for a randomly chosen quadratic map* $\mathcal{R} : \mathbb{F}_q^m \to \mathbb{F}_q^m$ *and a Rainbow public key* $\mathcal{P} : \mathbb{F}_q^n \to \mathbb{F}_q^m$ *is a hard problem, then our multivariate blind signature scheme satisfies universal-one-more-unforgeability against computationally bounded adversaries. That is to say, for all PPT adversaries* $\mathcal{A}$*, their advantage in winning the universal-one-more-unforgeability game (of Section 2) is at most negligible:*

$$\forall \mathcal{A} \,.\, \mathsf{Adv}_{\mathcal{MBSS}}^{\mathsf{universal-one-more-unforgeability}}(\mathcal{A}) \leq \mathsf{negl} \ .$$

Proof. We present a sequence of games argument showing that any adversary winning the Universal-One-More-Unforgeability game logically implies that the mentioned hard problem is efficiently solvable.
Let **Game 0** be the universal-one-more-unforgeability game as defined in Section 2. By assumption, we have an adversary $\mathcal{A}$ who wins with noticeable probability in polynomial time.
Let **Game 1** be the universal-one-more-unforgeability game but for the modified blind signature scheme where for each signature knowledge of $(\mathbf{z}, \mathbf{z}^\star)$ satisfying $\mathcal{P}(\mathbf{z}) + \mathcal{R}(\mathbf{z}^\star) = \mathcal{H}(d)$ is proven interactively using the protocol of Section 3.2, instead of producing a non-interactive proof σ. The simulator can win this game by simulating an instance of **Game 0** and presenting the **Game 0**-adversary with a random oracle that is programmed to respond with the same challenge-message that the simulator receives from the challenger.
Let **Game 2** be the universal-one-more-unforgeability game for the modified scheme that drops blindness altogether. Instead of proving knowledge of $(\mathbf{z}, \mathbf{z}^\star)$ in zero-knowledge, knowledge is proven straightforwardly by simply sending this pair to the challenger. The simulator can win this game by simulating **Game 1** and using the extractor machine associated with the zero-knowledge proof to obtain $(\mathbf{z}, \mathbf{z}^\star)$.
Let **Game 3** be the universal unforgeability under chosen message attack game for the signature scheme whose public key is $(\mathcal{P}, \mathcal{R})$, with the additional option for the adversary to query inverses under $\mathcal{P}$ as long as the message $d^\star$, the message for which a signature is to be forged, was not yet sent. The simulator wins this game by simulating **Game 2**. The blind-signature requests are answered by querying for an inverse under $\mathcal{P}$. After the adversary outputs his list $\mathcal{L}$ of message / signature pairs, the simulator requests the message $d^\star$ from the challenger for which a signature is to be forged. This message is relayed to the simulated adversary.
Let **Game 4** be the proper universal unforgeability under chosen message attack game for the signature scheme whose public key is $(\mathcal{P}, \mathcal{R})$, *i.e.*, without the ability to query for inverses under $\mathcal{P}$. Heuristically, the same adversary that wins **Game 3** should win **Game 4**. The reason is that the ability to query inverses under $\mathcal{P}$ before $d^\star$ is known does not help the adversary at all. Since $\mathcal{P}$ is a Rainbow public key and Rainbow is secure in its own right, the ability to query inverses should not help the adversary to either recover the secret key or find

his own inverses. Otherwise it would be possible to mount an attack exploiting this fact.
Let **Game 5** be the following non-interactive game, or problem: given $(\mathcal{P}, \mathcal{R})$, find $(\mathbf{x}_1, \mathbf{x}_2) \in \mathbb{F}_q^n \times \mathbb{F}_q^m$ such that $\mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) = 0$. The simulator can solve this problem by picking a random $\mathbf{s} \in_R \mathbb{F}_q^m$. He then simulates **Game 4** and presents its adversary with $(\mathcal{P}, \mathcal{R} + \mathbf{s})$ and with access to the backdoored random oracle $\mathcal{H}'(x) = \mathcal{P}(\mathcal{H}_1(x)) + \mathcal{R}(\mathcal{H}_2(x)) + \mathbf{s}$, where $\mathcal{H}_1 : \{0,1\}^* \to \mathbb{F}_q^n$ and $\mathcal{H}_2 : \{0,1\}^* \to \mathbb{F}_q^m$ are true random oracles. Under the (very reasonable) assumption that the distribution of $\mathcal{H}'$ is computationally indistinguishable from that of a true random oracle, the adversary's winning probability is still significant. The simulator answers a signature query $d \in \{0,1\}^*$ with $(\mathbf{x}_1, \mathbf{x}_2)$ where $\mathbf{x}_1 = \mathcal{H}_1(d)$ and $\mathbf{x}_2 = \mathcal{H}_2(d)$, which is necessarily a valid signature from the point of view of the adversary who can verify that $\mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) + \mathbf{s} = \mathcal{H}'(d)$. When the adversary indicates he is done with querying signatures, the simulator chooses a new message $d^\star$, programs $\mathcal{H}'(d^\star) = \mathbf{s}$, and sends $d^\star$ to the adversary. A winning adversary therefore solves $\mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) + \mathbf{s} = \mathbf{s}$, which is hard because it is equivalent to solving $\mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) = \mathbf{0}$. This concludes the proof of Thm. 3.

One of the premises of Thm. 3 remains to be shown: that finding a solution to the system $\bar{\mathcal{P}}(\mathbf{x}) = \mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) = \mathbf{0}$, which is a system of m quadratic equations in $n + m$ variables, is a difficult task. We have no rigorous proof for this (such a proof would imply $\mathbf{P} \neq \mathbf{NP}$) but we justify making this assumption based on common hardness arguments from MQ cryptography. In particular, there are two attack strategies known against multivariate systems:

Direct Attacks: In a direct attack, one tries to solve the system $\bar{\mathcal{P}}(\mathbf{x}) = \mathbf{0}$ as an instance of the MQ Problem. Since the system $\bar{\mathcal{P}}$ is underdetermined, there are two possibilities to do this. One can use a special algorithm against underdetermined multivariate systems [30] or, after fixing n of the variables, a Gröbner Basis algorithm such as Faugéres F_4 [10]. For suitably chosen parameters, both approaches are infeasible.

The second possibility to solve a multivariate system such as $\mathcal{P}'$ are the so called **Structural Attacks**. In this type of attack one uses the known structure of the system $\bar{\mathcal{P}}$ in order to find a decomposition $\bar{\mathcal{P}}$ into easily invertible maps. Note that, in our case we can write

$$\begin{aligned}\bar{\mathcal{P}}(\mathbf{x}) &= \mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) \\ &= \mathcal{S} \circ \mathcal{F} \circ \mathcal{T}(\mathbf{x}_1) + \mathcal{S} \circ \underbrace{\mathcal{S}^{-1} \circ \mathcal{R}}_{\mathcal{R}'}(\mathbf{x}_2) \\ &= \mathcal{S} \circ \underbrace{(\mathcal{F} + \mathcal{R}')}_{\mathcal{F}'} \circ \mathcal{T}'(\mathbf{x}),\end{aligned}$$

where the matrix T' representing the linear transformation $\mathcal{T}'$ is given by

$$T' = \begin{pmatrix} T & 0 \\ 0 & 1_m \end{pmatrix} \in \mathbb{F}^{(n+m)\times(n+m)}.$$

In order to solve the system $\bar{\mathcal{P}}$ using a structural attack, we have to use the known structure of the map $\mathcal{F}' = \mathcal{F} + \mathcal{S}^{-1} \circ \mathcal{R}$ to recover the linear maps $\mathcal{S}$ and $\mathcal{T}'$ (or, since the structure of $\mathcal{T}'$ is mostly known, the matrix T). However, since the coefficients of both $\mathcal{S}$ and $\mathcal{R}$ are chosen uniformly at random, the map $\mathcal{R}' = \mathcal{S}^{-1} \circ \mathcal{R}$ is a random quadratic map over $\mathbb{F}^m$. The only structure we can use for a structural attack therefore comes from the map $\mathcal{F}$, which is the central map of the underlying multivariate signature scheme. Therefore, we are in exactly the same situation as if attacking the underlying multivariate scheme using a structural attack. This means that a structural attack against our blind signature scheme is at least as hard as a structural attack against the underlying multivariate signature scheme. By choosing the parameters of the underlying scheme in an appropriate way, we therefore can prevent this type of attack against our blind signature scheme.

5.3 Quantum Security

The technique proposed in [12] is capable of transforming $(2n + 1)$-pass zero-knowledge proofs into non-interactive zero-knowledge proofs that are secure against classical adversaries in the random oracle model. However, the behaviour of this transform against quantum adversaries is not well understood because the random oracle should be accessible to the quantum adversary and answer queries *in quantum superposition*, and many standard proof techniques do not carry over to this setting. See Boneh et al. [2] for an excellent treatment of proofs that fail in the quantum random oracle model.
Formally proving soundness against quantum adversaries seems to be a rather involved task beyond the scope of this paper. Instead, we are content to conjecture that there exists a commitment scheme such that the technique of [12] results in a non-interactive zero-knowledge proof that is secure against quantum adversaries as well as classical ones. This conjecture is implicit in the works of Sakumoto et al. [28], and Hülsing et al. [12].

6 Discussion

6.1 Parameters

In this section we propose concrete parameter sets for our blind signature scheme. As observed in the previous section, we have to choose the parameters in a way that

a) solving a random system of m quadratic equations in m variables is infeasible,
b) inverting an MQ public key with the given parameters is infeasible, and
c) a direct attack against a system of m quadratic equations in $n+m$ variables is infeasible.

Since condition (a) is implied by (c), we only have to consider (b) and (c). In order to defend our scheme against attacks of type (b), we follow the recommendations

of [21]. Regarding (c), we have to consider that the system $\mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2) = \mathbf{w}$ is highly underdetermined (in the case of $\mathcal{P}$ being a Rainbow public key, the number of variables in this system exceeds the number of equations by a factor of about 3). As a result of Thomae et al. shows, such systems can be solved significantly faster than determined systems.

Proposition 1. *[30] Solving an MQ system of m equations in $n = \omega \cdot m$ variables is only as hard as solving a determined MQ system of $m - \lfloor \omega \rfloor + 1$ equations.*

According to this result, we have to increase the number of equations in our system by 2 (compared to the parameters of a standard Rainbow instance). Table 1 shows the parameters we propose for our scheme for various targeted security levels.

security level (bit)	parameters $(\mathbb{F}, (v_1, o_1, o_2))$	# rounds	public key size (kB)	private key size (kB)	blind sig. size (kB)
80	(GF(31),(16,18,17))	84	29.4	20.1	11.5
100	(GF(31),(20,22,21))	105	54.6	36.6	17.6
128	(GF(31),(25,27,27))	135	106.8	70.2	28.5
192	(GF(31),(37,35,35))	202	342.8	219.0	63.2
256	(GF(31),(50,53,53))	269	802.4	507.1	111.9

Table 1. Proposed parameters for our blind signature scheme (GF(31)).

6.2 Efficiency

During the interactive part of the signature generation process, the signer has to generate one Rainbow signature for the message $\tilde{\mathbf{w}} = \mathbf{w} - \mathbf{w}^\star$.
For the user, the most costly part of the signature generation is the repeated evaluation of the system $\bar{\mathcal{P}}(\mathbf{x}) = \mathcal{P}(\mathbf{x}_1) + \mathcal{R}(\mathbf{x}_2)$. During the computation of the commitments $\mathbf{c}_{0,i}$ and $\mathbf{c}_{1,i}$ $(i = 1, \ldots, r)$ (step 2 of the signature generation process) this has to be done $3 \cdot r$ times (one evaluation of $\mathcal{G}$ corresponds to 3 evaluations of $\bar{\mathcal{P}}$). In step 4 of the process (computation of $\mathbf{e}_{1,i}$) we need r evaluations of $\bar{\mathcal{P}}$. Altogether, the user has to evaluate the system $4r$ times.
During verification, the verifier has to compute the commitments $c_{ch_i,i}$ $(i = 1, \ldots, r)$. If $ch_i = 0$, he needs for this 1 evaluation of $\bar{\mathcal{P}}$, in the case of $ch_2 = 1$ he needs 4 evaluations. On average, the verifier needs therefore $\frac{r}{2} \cdot (1 + 4) = 2.5 \cdot r$ evaluations of the system $\bar{\mathcal{P}}$.
While the system $\bar{P}$ consists of m quadratic equations in $m + n$ variables, the inner structure of the system can be used to speed up the evaluation. In fact, the system $\bar{\mathcal{P}}$ is the sum of two smaller systems $\mathcal{P} : \mathbb{F}^n \to \mathbb{F}^m$ and $\mathcal{R} : \mathbb{F}^m \to \mathbb{F}^m$. Therefore, we can evaluate $\bar{\mathcal{P}}$ by evaluating $\mathcal{P}$ and $\mathcal{R}$ separately and adding the results.

6.3 Implementation

We implemented all functionalities in Sage [27] to prove concept validity. Table 2 contains the timing results for the matching parameter sets of Table 1, demonstrating that our scheme is somewhat efficient and practicable even for very poorly-optimized Sage code. These results were obtained on a 3.3 GHz Intel Quadcore with 6,144 kB of cache.
Despite of these relatively large numbers, we are very optimistic about the speed of our blind signatures when implemented in a less abstract and more memory-conscious programming language. For instance, Hülsing et al.'s optimized MQDSS manages to generate (classically) 256-bit-secure signatures in 6.79 ms and verify them in even less time [12]. As the MQDSS represents the bottleneck of our scheme, a similarly optimized implementation could potentially drop signature generation and verification time by several orders of magnitude.

sec. lvl.	Key Gen.	Sign (`Signer`)	Sig. Gen. (`User`)	Sig. Verification
80	4,007	7	2,018	1,424
100	9,392	13	3,649	2,656
128	25,517	19	7,760	5,505
192	87,073	41	23,692	16,040
256	613,968	103	86,540	59,669

Table 2. Timing results of a Sage implementation of our blind signature scheme. All units are milliseconds, except for the security level.

6.4 Comparison

Table 3 shows a comparison of our scheme to the standard RSA blind signature scheme and the lattice-based blind signature scheme of Rückert [26]. The RSA blind signature scheme does not offer any security against quantum computers. The public keys of Rückert's scheme are smaller than those of our scheme, although ours are still competitive. Like the standard RSA blind signature scheme, our scheme requires 2 steps of communication between the user and the signer in order to produce the blind signature. This is in contrast to Rückert's scheme where this number is 4. More importantly, our scheme outperforms that of Rückert in terms of signature size.
At this point, an apples-to-apples comparison of operational speed is not possible. Nevertheless, regardless of speed, the main selling point of our scheme is its reliance on different computational problems from those used in other branches of cryptography, including lattice-based cryptography.

Security lvl. (bit)	Scheme	comm.	Pub. key size (kB)	Sig. size (kB)	Post-quantum?
76	RSA-1229	2	1.2	1.2	×
	Lattice-1024	4	10.2	66.9	✓
	Our scheme(GF(31),16,18,17)	2	29.4	11.5	✓
102	RSA-3313	2	3.3	3.3	×
	Lattice-2048	4	23.6	89.4	✓
	Our scheme(GF(31),20,22,21)	2	54.6	17.6	✓

Table 3. Comparison of different blind signature schemes. The secrutiy levels are adopted from Rückert [26].

7 Conclusion

In this paper we proposed the first multivariate based blind signature scheme. Our scheme is very efficient and produces much shorter blind signatures than the lattice based scheme of Rückert [26], making our scheme the most promising candidate for establishing a post-quantum blind signature scheme.
Our construction is notably generic. While we only show that it applies to Rainbow and MQDSS, we only use their properties abstractly and it is perfectly conceivable that another combination of trapdoor-based MQ signature scheme with a non-interactive proof of knowledge of the solution to an MQ system will give the same result. Indeed, our design demonstrates that the combination of a dedicated signature scheme with an identification scheme relying on the same hard problem, is a powerful construction — and may apply in other branches of cryptography as well.
Lastly, one major use case of blind signatures is anonymous identification. In this scenario, one may reasonably dispense with the transformed signature scheme and instead directly use the underlying interactive identification scheme, thus sacrificing non-interactivity for less computation and bandwidth. Likewise, other use cases such as anonymous database access require *reusable* anonymous credentials. Our scheme can be adapted to fit this scenario as well, simply by specifying that all users obtain a blind signature on the same public parameter.

Acknowledgements

The authors would like to thank the reviewers and the shepherd in particular for their helpful comments. This work was supported in part by the Research Council KU Leuven: C16/15/058. In addition, this work was supported by the European Commission through the Horizon 2020research and innovation programme under grant agreement No H2020-ICT-2014-644371 WITDOM, H2020-ICT-2014-645622 PQCRYPTO and H2020-DS-2014-653497 PANORAMIX, and through the SECURITY programme under FP7-SEC-2013-1-607049 EKSISTENZ. Alan Szepieniec is being supported by a doctoral grant of the Flemish Agency for Innovation and Entrepreneurship (VLAIO, formerly IWT).

References

1. M. Bellare, C. Namprempre, D. Pointcheval, M. Semanko: The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme. Journal of Cryptology. Volume 16, Issue 3, pp. 185 - 215. Springer, Jun. 2003.
2. D. Boneh, Ö. Dagdelen, M. Fischlin, A. Lehmann, C. Schaffner, M. Zhandry: Random oracles in a quantum world. In International Conference on the Theory and Application of Cryptology and Information Security, pp. 41-69. Springer Berlin Heidelberg, 2011.
3. D. Chaum: Blind Signatures for untraceable payment. Proceedings of CRYPTO 1982, pp. 199 - 203. Plenum Press, 1983.
4. D.J. Bernstein, J. Buchmann, E. Dahmen (eds.): post-quantum Cryptography. Springer, 2009.
5. A. Bogdanov, T. Eisenbarth, A. Rupp, C. Wolf: Time-area optimized public-key engines: MQ-cryptosystems as replacement for elliptic curves? CHES 2008, LNCS vol. 5154, pp. 45-61. Springer, 2008.
6. A.I.T. Chen, M.-S. Chen, T.-R. Chen, C.-M. Cheng, J. Ding, E. L.-H. Kuo, F. Y.-S. Lee, B.-Y. Yang: SSE implementation of multivariate PKCs on modern x86 cpus. CHES 2009, LNCS vol. 5747, pp. 33 - 48. Springer, 2009.
7. J. Ding, V. Dubois, B.-Y. Yang, O. C.-H. Chen, C.-M. Cheng: Could SFLASH be repaired? International Colloquium on Automata, Languages, and Programming, 2008. pp. 691-701.
8. J. Ding, J. E. Gower, D. S. Schmidt: Multivariate Public Key Cryptosystems. Springer, 2006.
9. J. Ding, D. S. Schmidt: Rainbow, a new multivariate polynomial signature scheme. ACNS 2005, LNCS vol. 3531, pp. 164-175. Springer, 2005.
10. J.C. Faugère: A new efficient algorithm for computing Gröbner bases (F4). Journal of Pure and Applied Algebra 139, pp. 61-88 (1999).
11. M. R. Garey and D. S. Johnson: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Company 1979.
12. A. Hülsing, J. Rijneveld, S. Samardjiska, P. Schwabe: From 5-pass MQ-based identification to MQ-based signatures. Cryptology ePrint Archive: Report 2016/708
13. A. Juels, M. Luby, R. Ostrovsky: Security of Blind Digital Signatures. CRYPTO 1997, LNCS vol. 1294, pp. 150 - 164. Springer 1997.
14. D. Kravitz: Digital Signature Algorithm. US patent 5231668 (July 1991).
15. A. Kipnis, L. Patarin, L. Goubin: Unbalanced Oil and Vinegar Schemes. EUROCRYPT 1999, LNCS vol. 1592, pp. 206–222. Springer, 1999.
16. T. Matsumoto, H. Imai: Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. EUROCRYPT 1988. LNCS vol. 330, pp. 419-453. Springer, 1988.
17. D. Goodin: NSA preps quantum-resistant algorithms to head off crypto-apocalypse. http://arstechnica.com/security/2015/08/nsa-preps-quantum-resistant-algorithms-to-head-off-crypto-apocolypse/.
18. National Institute of Standards and Technology: Report on post-quantum Cryptography. NISTIR draft 8105, `http://csrc.nist.gov/publications/drafts/nistir-8105/nistir_8105_draft.pdf`.
19. J. Patarin, N. Courtois, and L. Goubin. "Quartz, 128-bit long digital signatures." Cryptographers' Track at the RSA Conference. Springer Berlin Heidelberg, 2001.
20. A. Petzoldt, S. Bulygin, J. Buchmann: A Multivariate based Threshold Ring Signature Scheme. Appl. Algebra Eng. Commun. Comput. 24(3-4); 255-275 (2012).

21. A. Petzoldt, S. Bulygin, J. Buchmann: Selecting Parameters for the Rainbow Signature Scheme. PQCrypto 2010, LNCS vol. 6061, pp. 218-240. Springer, 2010.
22. A. Petzoldt, S. Bulygin, J. Buchmann: CyclicRainbow - A Multivariate Signature Scheme with a Partially Cyclic Public Key. INDOCRYPT 2010, LNCS vol. 6498, pp. 33-48. Springer, 2010.
23. A. Petzoldt, S. Bulygin, J. Buchmann: Fast Verification for Improved Versions of the UOV and Rainbow Signature Schemes. PQCrypto, LNCS vol. 7932, pp. 188-202. Springer, 2013.
24. A. Petzoldt, M.S. Chen, B.Y. Yang, C. Tao, J. Ding: Design Principles for HFEv-based Signature Schemes. ASIACRYPT 2015 - Part 1, LNCS vol. 9452, pp. 311-334.Springer, 2015.
25. R. L. Rivest, A. Shamir, L. Adleman: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Commun. ACM 21 (2), pp. 120-126 (1978).
26. M. Rückert: Lattice-Based Blind Signatures. ASIACRYPT 2010 , LNCS vol. 6477, pp. 413-430. Springer, 2010.
27. SageMath, the Sage Mathematics Software System (Version 7.1), The Sage Developers, 2016, `http://www.sagemath.org`.
28. K. Sakumoto, T. Shirai, H. Hiwatari: Public-Key Identification Schemes Based on Multivariate Quadratic Polynomials. CRYPTO 2011, LNCS vol. 6841, pp. 706 - 723. Springer, 2011.
29. P. Shor: Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, SIAM J. Comput. 26 (5), pp. 1484 - 1509 (1997).
30. E. Thomae, C. Wolf: Solving Underdetermined Systems of Multivariate Quadratic Equations Revisited. PQCrypto 2012, LNCS vol. 7293, pp. 156–171. Springer, 2012.
31. T. Yasuda, X. Dahan, Y-J Huang, T. Takagi, K. Sakurai: MQ Challenge: Hardness Evaluation of Solving Multivariate Quadratic Problems. IACR Cryptology ePrint Archive 2015 (2015): 275.
32. B.Y. Yang, J.M. Chen: Building secure tame-like multivariate public-key cryptosystems.: The new TTS. CHES 2004, LNCS vol. 3156, pp. 371- 385. Springer, 2004.