A Multivariate based Threshold Ring Signature Scheme

Albrecht Petzoldt[1,2], Stanislav Bulygin[1,2], and Johannes Buchmann[1,2]

[1] Technische Universität Darmstadt, Department of Computer Science
Hochschulstraße 10, 64289 Darmstadt, Germany
{apetzoldt,buchmann}@cdc.informatik.tu-darmstadt.de

[2] Center for Advanced Security Research Darmstadt - CASED
Mornewegstraße 32, 64293 Darmstadt, Germany
{johannes.buchmann,Stanislav.Bulygin}@cased.de

Abstract. In [16], Sakumoto et al. presented a new multivariate identification scheme, whose security is based solely on the MQ-Problem of solving systems of quadratic equations over finite fields. In this paper we extend this scheme to a threshold ring identification and signature scheme. Our scheme is the first multivariate scheme of this type and generally the first multivariate signature scheme with special properties. Despite the fact that we need more rounds to achieve given levels of security, the signatures are at least twice shorter than those obtained by other post-quantum (e.g. code based) constructions. Furthermore, our scheme offers provable security, which is quite a rare fact in multivariate cryptography.
Keywords: Threshold Ring Signature, Post-Quantum Cryptography, Multivariate Cryptography, MQ Problem

1 Introduction

Since the introduction of threshold ring signatures in 2002 [6], a number of schemes in this area were proposed [5,13]. Most of them are based on number theoretic problems and share two disadvantages:

1) the signature length and/or the computational complexity depends not only on the number of users N, but also on the number of actual signers t
2) they will be broken when large enough quantum computers are built [2].

Therefore we need alternatives for these schemes. In the last years much work has been done to develop post-quantum threshold ring signature schemes. These schemes are based on mathematical problems which are not affected by Shor's algorithm [17] and therefore are believed to resist attacks with quantum computers [2]. We want to mention here the code-based construction by Aguilar et al. [1] and the lattice-based scheme of Cayrel et al. [7].

In this paper we propose a new threshold ring signature scheme based on multivariate polynomial systems. We achieve this by extending the identification scheme of [16] to a threshold ring identification scheme and applying the Fiat-Shamir paradigm [11] to transform it into a signature scheme. Although our scheme requires more rounds than code- and lattice-based schemes to reach the same level of security, it produces significantly shorter signatures. In particular, both the signature length and the computational complexity of our scheme are independent of t and linear in N. Our scheme is the first multivariate scheme of this type and generally the first multivariate signature scheme with special properties. The security of our scheme is based solely on the MQ-Problem of solving systems of multivariate quadratic equations over finite fields. Therefore our scheme is provably secure, which is quite a rare fact in multivariate cryptography.

The outline of this paper is as follows: In the next section we recall the basic definitions about threshold ring signatures and the Fiat-Shamir paradigm. Section 3 introduces the identification scheme of [16] which is the basis of our construction. In Section 4 we show how to extend this scheme to a threshold ring identification and signature scheme and describe our scheme in detail, whereas Section 5 considers the security of the scheme. In Section 6 we give concrete parameters for our scheme and compare it with other existing threshold ring signature schemes. Finally, Section 7 concludes the paper.

2 Preliminaries

2.1 Threshold Ring Signatures

Threshold ring signatures were introduced in 2002 by Bresson, Stern and Szydlo [6].
The receiver of a (t, N)-threshold ring signature shall be convinced that t (among a larger group of N) users have signed a message without being able to identify this subgroup.

Definition 1. *Let $t < N$ be integers. A (t, N)-threshold ring signature scheme consists of 3 algorithms*

- *KeyGen: is a probabilistic algorithm which outputs N pairs of private and public keys $(sk_1, pk_1), \ldots, (sk_N, pk_N)$.*
- *Sign: is a probabilistic interactive protocol between t users, involving a set $(pk_1, \ldots, pk_N)$ of public keys, a set $(sk_{i_1}, \ldots, sk_{i_t})$ of private keys and a message m, and which outputs a (t, N)-threshold ring signature σ for the message m.*
- *Verify: is a deterministic algorithm which takes as input a threshold value t, a set of public keys $(pk_1, \ldots, pk_N)$ and a message/signature pair (m, σ), and outputs 1 if σ is a valid (t, N)-threshold ring signature for the message m w.r.t. the public keys $(pk_1, \ldots, pk_N)$ and 0 otherwise.*

The basic security criteria of a threshold ring signature scheme are

- *Correctness*: A fairly generated (t, N)-threshold ring signature is accepted with overwhelming probability.
- *Unforgeability*: Without the knowledge of at least t secret keys it is infeasible to generate a valid (t, N)-threshold ring signature. More formally we can define this property by the following game:
  1. The challenger $\mathcal{C}$ uses algorithm `KeyGen` to produce key pairs sk_i, pk_i $(i = 1, \ldots, N)$. He gives all the public keys to the forger $\mathcal{F}$ and keeps the secret keys for himself.
  2. $\mathcal{F}$ is allowed to ask the following queries:
     - *Signing query*: $\mathcal{F}$ chooses a message m and gives it to $\mathcal{C}$. The challenger uses algorithm *Sign* to produce a threshold ring signature σ for the message m and gives it to $\mathcal{F}$.
     - *Corrupt query*: $\mathcal{F}$ chooses an integer $i \in 1, \ldots, N$. $\mathcal{C}$ gives him the corresponding secret key sk_i. Note that the number of corrupt queries must be strictly less than t.

  $\mathcal{F}$ wins the game, if he can generate a valid threshold ring signature $\sigma^\star$ for a new message $m^\star$. A threshold ring signature scheme, for which the success probability of $\mathcal{F}$ is negligible, is called *existentially unforgeable under chosen message attacks*.
- *Source-Hiding*: Given a message-signature pair (m, σ), it is infeasible for the verifier to reveal which t-subset of signers contributed to σ. More formally, we can define this property by the following game: An attacker is given two different sets of signers $S_1 = \{P_{11}, \ldots, P_{1t}\}$ and $S_2 = \{P_{21}, \ldots, P_{2t}\}$. He can ask both groups to sign messages. Finally, he gets a message $m^\star$ and a signature $\sigma^\star$ which was generated by one of the groups. He succeeds, if he can decide which group signed the message with probability $> \frac{1}{2}$.

Many threshold ring signature schemes are based on a threshold ring identification scheme and the Fiat-Shamir paradigm.
In a threshold ring identification scheme the t signers choose a leader L, which gathers the individual commitments and responses and communicates with the verifier. Figure 1 shows one round of such a scheme.

2.2 Fiat-Shamir heuristic

The Fiat-Shamir heuristic [11] is a general way to convert an identification scheme into a signature scheme. The idea is to start from a 3-pass identification scheme (with commitment Com, challenge Ch and response Rsp). To sign a message m, the signer produces a valid transcript (Com, Ch, Rsp)

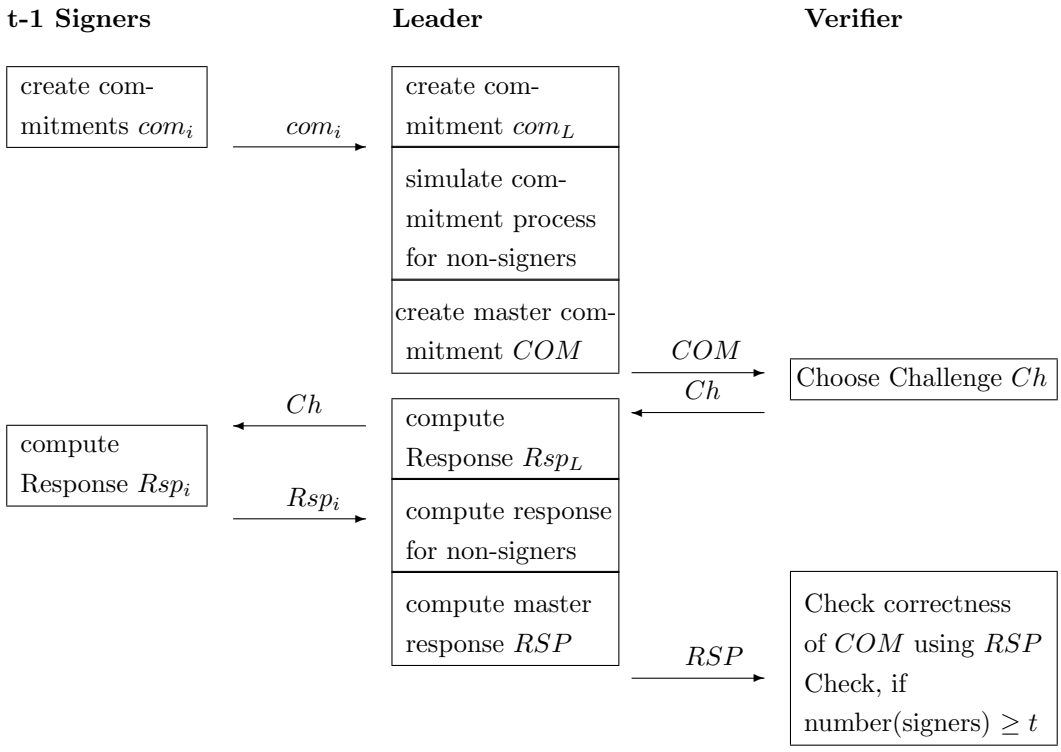

Fig. 1. Threshold Ring Identification

of the interactive identification protocol. Here, $Ch = \mathcal{R}(m, Com)$ for a random oracle $\mathcal{R}$.
Since the challenge Ch can be computed out of the message and the commitments, it has not to be part of the signature. Therefore, a signature has the form

$$\sigma = (Com|Rsp).^3 \tag{1}$$

As shown by Pointcheval and Stern in [14], an honest-verifier zero-knowledge 3-pass identification scheme leads via the Fiat-Shamir heuristic to a signature scheme, which is existentially unforgeable under chosen message attacks (see Subsection 2.1) in the random oracle model.

3 Multivariate Cryptography

Multivariate Cryptography is one of the main candidates to guarantee the security of communication in the post-quantum world. Since multivariate cryptosystems need only simple operations (namely addition and multiplication over small finite fields), they require only modest computational resources which makes them suitable for the use on low cost devices like RFID chips and smartcards. Additionally, multivariate schemes seem to be faster than classical Public-Key-Cryptosystems like RSA and ECC [4,8]. A good overview on existing multivariate schemes can be found in [9].

3.1 Multivariate quadratic systems

The basic objects of multivariate cryptography are systems of multivariate quadratic equations over finite fields. We write such a system of m equations in n variables as

$$\begin{aligned} \sum_{i=1}^{n}\sum_{j=i}^{n} p_{ij}^{(1)} \cdot x_i \cdot x_j + \sum_{i=1}^{n} p_i^{(1)} \cdot x_i + p_0^{(1)} &= 0 \\ &\vdots \\ \sum_{i=1}^{n}\sum_{j=i}^{n} p_{ij}^{(m)} \cdot x_i \cdot x_j + \sum_{i=1}^{n} p_i^{(m)} \cdot x_i + p_0^{(m)} &= 0. \end{aligned} \tag{2}$$

The security of multivariate cryptosystems is based on the

MQ-Problem: Given m quadratic polynomials $p_1, \ldots, p_m$ in n variables over a finite field $\mathbb{F}$, find a vector $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{F}^n$ such that $p_1(\mathbf{x}) = \cdots = p_m(\mathbf{x}) = 0$.

The MQ-Problem is proven to be NP-hard even for quadratic systems over the field of two elements [12].
The most efficient method for solving multivariate polynomial systems is the Hybrid approach [3] of Bettale, Faugère and Perret, which combines exhaustive search and Faugères F_5 algorithm [10].

Remark: The security of most of the existing multivariate schemes is based not only on the MQ-Problem, but also on the EIP-Problem (Extended Isomorphism of Polynomials). This fact prevented researchers to give security proofs for their schemes. The security of our threshold signature scheme is based solely on the MQ-Problem. Therefore, it is one of the first multivariate schemes with provable security (see Section 5).

[3] To achieve given levels of security, it might be neccessary to run the identification scheme several (say M) times. In this case, the challenge is given as $Ch = \mathcal{R}(m, Com_1, \ldots, Com_M)$ and the signature has the form $\sigma = (Com_1, \ldots, Com_M, Rsp_1, \ldots, Rsp_M)$.

3.2 The MQ-based identification scheme

At CRYPTO 2011 Sakumoto et al. presented a new identification scheme whose security is based solely on the MQ-Problem [16] .

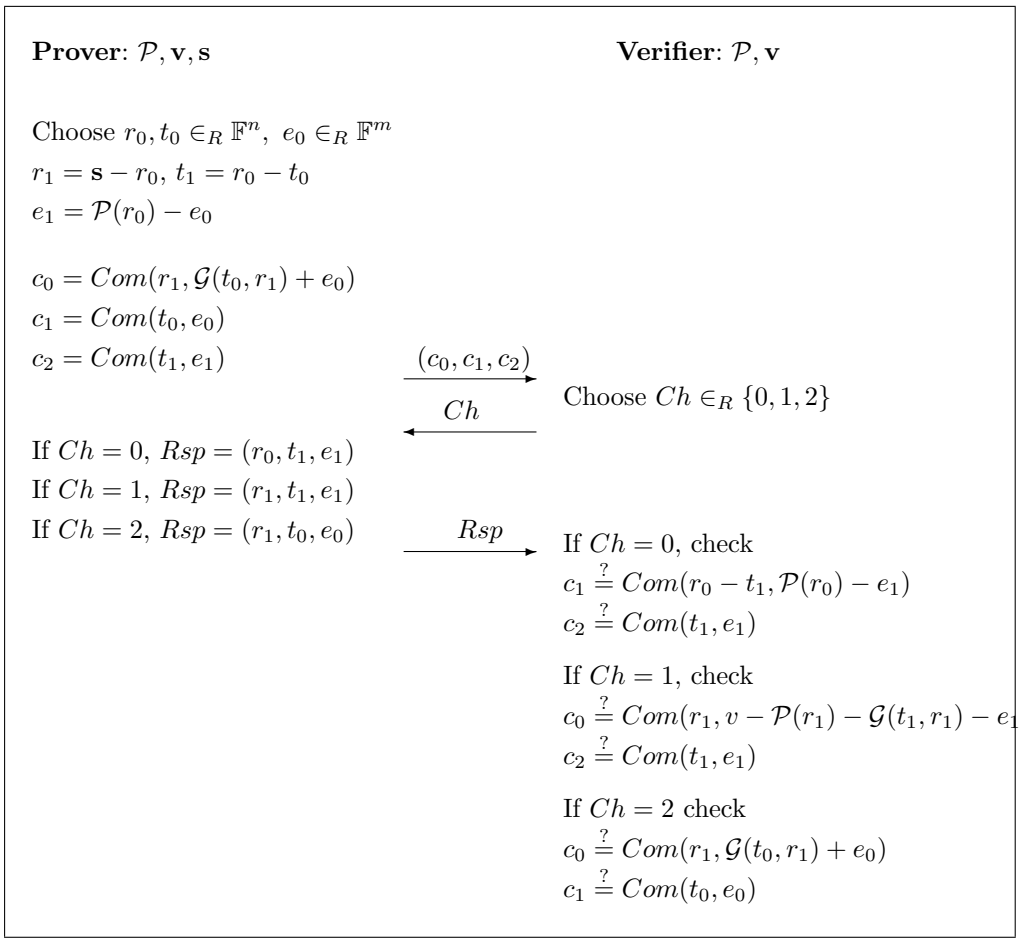

Fig. 2. The identification scheme of [16]

In the scheme we have a multivariate quadratic system $\mathcal{P} : \mathbb{F}^n \to \mathbb{F}^m$ which is viewed as a system parameter and fixed for a large number of users. Every user chooses a vector $s \in \mathbb{F}^n$ as his secret key and computes his public key as $v = \mathcal{P}(s) \in \mathbb{F}^m$.
To identify himself to a verifyer, he has to show that he indeed knows s (without revealing any information about s).
To create a zero-knowledge proof of the vector s, we need the so called *polar form* of the multivariate system $\mathcal{P}$, which is defined as

$$\mathcal{G}(x, y) = \mathcal{P}(x + y) - \mathcal{P}(x) - \mathcal{P}(y). \tag{3}$$

Note that $\mathcal{G}(x, y)$ is bilinear in x and y.

The basic observation of [16] is the following: The knowledge of s is equivalent to knowing a tuple $(r_0, r_1, t_0, t_1, e_0, e_1)$ satisfying

$$\mathcal{G}(t_0, r_1) + e_0 = v - \mathcal{P}(r_1) - \mathcal{G}(t_1, r_1) - e_1 \tag{4}$$

$$(t_0, e_0) = (r_0 - t_1, \mathcal{P}(r_0) - e_1). \tag{5}$$

Under the assumption that there exists a computationally binding and statistically hiding commitment scheme Com[4], the authors of [16] used this observation to create a zero knowledge proof for a solution of the system $\mathcal{P}(x) = v$ (see Figure 2).

The scheme as shown in Figure 2 has a cheating probability per round of $\frac{2}{3}$. Therefore, one needs 52 rounds to reduce the impersonation probability to less than 2^{-30}.
The authors of [16] propose for their scheme $\mathbb{F} = GF(2)$, $n = 84$, $m = 80$ to achieve a security level of 80 bit. For this parameter set, the communication cost of the scheme (52 rounds) is 29,640 bits or 3.6 kB.

Additionally to the 3-pass version shown in Figure 2, the authors of [16] also presented a 5-pass version of their scheme. In this paper, we restrict ourselves to the 3-pass version.

4 Our threshold ring signature scheme

4.1 From identification to threshold ring identification

The authors of [16] propose to use the system $\mathcal{P}$ as a system parameter which stays the same for all users. Every user chooses randomly a vector $s \in \mathbb{F}^n$ as his private key and publishes $v = \mathcal{P}(s)$ as a public key. In our scheme, we turn this around. In a threshold ring signature scheme, the leader must be able to simulate the actions of the non-signers without knowing their secrets. To enable this, we fix the vector $v \in \mathbb{F}^m$ of the identification scheme to zero. Then, every user P_i chooses randomly a private key $s_i \in \mathbb{F}^n$ and creates a system $\mathcal{P}_i$ such that $\mathcal{P}_i(s_i) = 0$ (see Subsection 4.2 and Algorithm 1). If the system $\mathcal{P}_i$ does not contain constant terms, the zero vector $0 \in \mathbb{F}^n$ is always a solution of the system. This enables the leader to simulate the actions of the non-signers without knowing their secrets.

The verifier of the threshold ring signature scheme must be able to recognize how many of the possible signers contributed to the signature. To enable this, we add the following test to the scheme. For fields of characteristic 2 we have

$$s = r_0 + r_1 = 0 \Leftrightarrow r_0 = r_1 \text{ " } \Leftrightarrow \text{ " } Com(r_0) = Com(r_1). \tag{6}$$

Note that the second "$\Leftrightarrow$" is not really an "$\Leftrightarrow$ ". While the "$\Rightarrow$ " always holds for a deterministic commitment function the "$\Leftarrow$" holds only computationally due to the binding property of the commitment function. In the following we always work over fields of characteristic 2.

Basically we could make the test (6) at any point of the verification process. However, in a threshold ring signature scheme, the verifier must not be able to identify non-signers. To achieve this, we use a permutation $\Sigma \in S_N$ which permutes the N users before applying the test (6). This leads to a rather difficult problem: Regardless of the challenge, the public key $\mathcal{P}$ (or its polar form $\mathcal{G}$) is used during the verification step of the identification scheme. Therefore, if we permute the secret keys, the verification step can no longer be performed correctly. One possibility to solve this problem is to extend the protocol as follows:

[4] In practice this is realized by a collision- and pre-image resistant hash function.

Each signer P_i sends the two additional commitments $c_3^{(i)} = Com(r_0^{(i)})$ and $c_4^{(i)} = Com(r_1^{(i)})$ to the leader, which chooses randomly a permutation $\Sigma \in S_N$ and sends $C_3 = Com(\Sigma(c_3^{(1)}, \ldots, c_3^{(N)}))$ and $C_4 = Com(\Sigma(c_4^{(1)}, \ldots, c_4^{(N)}))$ to the verifier. On the Challenge $Ch = 3$, the leader reveals $\Sigma(c_3^{(1)}, \ldots, c_3^{(N)})$ and $\Sigma(c_4^{(1)}, \ldots, c_4^{(N)})$. The verifier checks the correctness of the commitments C_3 and C_4 and checks if there are at least t positions i with $c_3^{(i)} \neq c_4^{(i)}$ and therefore $s_i \neq 0$.

4.2 The Identification Scheme

Let U be a group of N users. A subgroup $S \subset U$ of t signers wants to identify itself to a verifier. To do this, the signers choose a leader L which gathers the commitments and responses of the single signers and communicates with the verifier (see Figure 3).

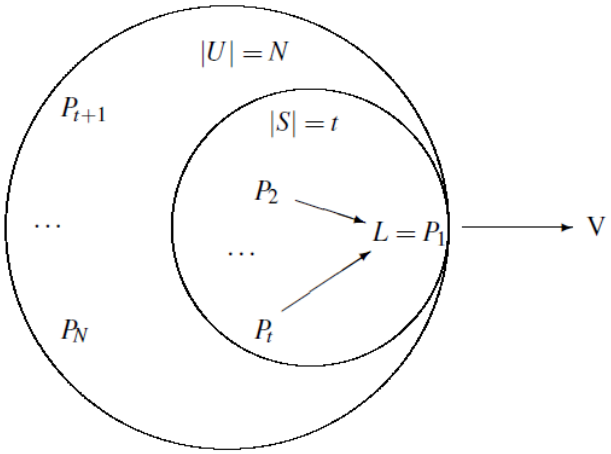

Fig. 3. Threshold Ring Identification. For simplicity we assume that $L = P_1$ and $S = \{P_1, \ldots, P_t\}$.

Key Generation Let $\mathbb{F}$ be a field of characteristic 2. Every user $P_i \in U$ chooses randomly a vector $s_i \in \mathbb{F}^n$ and creates a quadratic system $\mathcal{P}_i : \mathbb{F}^n \to \mathbb{F}^m$ such that $\mathcal{P}_i(s_i) = 0$. $\mathcal{P}_i$ must not contain constant terms. Algorithm 1 shows the key generation process of our scheme.
Here, $p_t^{(j)}$ is the coefficient of x_t in the $j-th$ component of $\mathcal{P}$.
The vector v computed in line 4 of the algorithm is a random looking vector in $\mathbb{F}^m$. In line 7 we change some of the linear coefficients of the system $\mathcal{P}$ in such a way that all the non zero elements of v are put to zero. Therefore we ensure that the public key derived by Algorithm 1 is a system $\mathcal{P}$ of m quadratic polynomials in n variables without constant terms such that $\mathcal{P}(s) = 0$.

The homogeneous quadratic part of the system $\mathcal{P}$ can be seen as a system parameter which is fixed for all N users and might be given by a random seed. This reduces the size of the public key by a large factor (see Table 3).

Remark: Since both the vector s (line 1) and the coefficients of the system $\mathcal{P}$ (line 3) were chosen uniformly at random from $\mathbb{F}$, the coefficients of the public key pk derived by Algorithm 1 are uniformly distributed in $\mathbb{F}$.

The public key of the group is simply the concatenation of all public keys, i.e. $\mathcal{P} = \mathcal{P}_1 || \ldots || \mathcal{P}_N$.

Algorithm 1 Key Generation process

Input: parameters $m,\ n$
Output: keypair (sk, pk)

Choose randomly a vector $s \in \mathbb{F}^n$
$t \leftarrow max\{j | s_j \neq 0\}$
Choose randomly a system $\mathcal{P}$ of m quadratic polynomials in n variables without constant terms
$v \leftarrow \mathcal{P}(s)$
for $j = 1$ to m **do**
  if $v_j \neq 0$ **then**
    $p_t^{(j)} \leftarrow p_t^{(j)} - \frac{v_j}{s_t}$
  end if
end for
$sk \leftarrow s$
$pk \leftarrow \mathcal{P}$

The Identification protocol One round of our threshold ring identification scheme works as follows:

1. Each of the t signers $P_i \in S$ (including the leader) chooses

$$r_0^{(i)}, t_0^{(i)} \in \mathbb{F}^n,\ e_0^{(i)} \in \mathbb{F}^m,$$

computes

$$r_1^{(i)} = s_i - r_0^{(i)},\ t_1^{(i)} = r_0^{(i)} - t_0^{(i)},\ e_1^{(i)} = \mathcal{P}_i(r_0^{(i)}) - e_0^{(i)},$$

$$\begin{aligned} c_0^{(i)} &= Com(r_1^{(i)}, \mathcal{G}_i(t_0^{(i)}, r_1^{(i)}) + e_0^{(i)}), \\ c_1^{(i)} &= Com(t_0^{(i)}, e_0^{(i)}), \\ c_2^{(i)} &= Com(t_1^{(i)}, e_1^{(i)}), \\ c_3^{(i)} &= Com(r_0^{(i)}), \\ c_4^{(i)} &= Com(r_1^{(i)}), \end{aligned}$$

and sends $c_0^{(i)},\ c_1^{(i)},\ c_2^{(i)},\ c_3^{(i)}$ and $c_4^{(i)}$ to the leader.

2. The leader computes $c_0^{(i)},\ c_1^{(i)},\ c_2^{(i)},\ c_3^{(i)}$ and $c_4^{(i)}$ for the $N - t$ non-signers $P_i \in U \setminus S$ (using 0 as "secret" s_i), chooses a random permutation $\Sigma \in S_N$, computes the master commitments

$$\begin{aligned} C_0 &= Com(c_0^{(1)}, \ldots, c_0^{(N)}) \\ C_1 &= Com(\Sigma, c_1^{(1)}, \ldots, c_1^{(N)}) \\ C_2 &= Com(c_2^{(1)}, \ldots, c_2^{(N)}) \\ C_3 &= Com(\Sigma(c_3^{(1)}, \ldots, c_3^{(N)})) \\ C_4 &= Com(\Sigma(c_4^{(1)}, \ldots, c_4^{(N)})) \end{aligned}$$

and sends $C_0,\ C_1,\ C_2,\ C_3$ and C_4 to the verifier.

3. The verifier chooses the challenge $Ch \in \{0, 1, 2, 3\}$ and sends it to the leader. If $Ch \in \{0, 1, 2\}$ the leader sends Ch to the $t - 1$ co-signers.

4. If $Ch \in \{0,1,2\}$, the t signers $P_i \in S$ (including the leader) compute their responses Rsp_i, namely

- If $Ch = 0$, $Rsp_i = (r_0^{(i)}, t_1^{(i)}, e_1^{(i)})$
- If $Ch = 1$, $Rsp_i = (r_1^{(i)}, t_1^{(i)}, e_1^{(i)})$
- If $Ch = 2$, $Rsp_i = (r_1^{(i)}, t_0^{(i)}, e_0^{(i)})$

and send Rsp_i to the leader.

5. The leader computes Rsp_i for the $N - t$ non-signers, computes the master response RSP

- If $Ch = 0$, $RSP = (\Sigma, Rsp_1, \ldots, Rsp_N)$
- If $Ch = 1$, $RSP = (Rsp_1, \ldots, Rsp_N)$
- If $Ch = 2$, $RSP = (\Sigma, Rsp_1, \ldots, Rsp_N)$
- If $Ch = 3$, $RSP = (\Sigma(c_3^{(1)}, \ldots, c_3^{(N)}), \Sigma(c_4^{(1)}, \ldots, c_4^{(N)}))$

and sends RSP to the verifier.

6. The verifier checks the correctness of the commitments

- If $Ch = 0$, he parses RSP into $\Sigma, r_0^{(1)},\ t_1^{(1)},\ e_1^{(1)}, \ldots, r_0^{(N)},\ t_1^{(N)},\ e_1^{(N)}$.
  For $i = 1, \ldots N$ he computes

$$\tilde{c}_1^{(i)} = Com(r_0^{(i)} - t_1^{(i)}, \mathcal{P}_i(r_0^{(i)}) - e_1^{(i)}),$$
$$\tilde{c}_2^{(i)} = Com(t_1^{(i)}, e_1^{(i)}) \text{ and}$$
$$\tilde{c}_3^{(i)} = Com(r_0^{(i)})$$

  and checks, if

$$C_1 \stackrel{?}{=} Com(\Sigma, \tilde{c}_1^{(1)}, \ldots, \tilde{c}_1^{(N)}),$$
$$C_2 \stackrel{?}{=} Com(\tilde{c}_2^{(1)}, \ldots, \tilde{c}_2^{(N)}) \text{ and}$$
$$C_3 \stackrel{?}{=} Com(\Sigma(\tilde{c}_3^{(1)}, \ldots, \tilde{c}_3^{(N)})).$$

- If $Ch = 1$, he parses RSP into $r_1^{(1)},\ t_1^{(1)},\ e_1^{(1)}, \ldots, r_1^{(N)},\ t_1^{(N)},\ e_1^{(N)}$.
  For $i = 1, \ldots N$ he computes

$$\tilde{c}_0^{(i)} = Com(r_1^{(i)}, -\mathcal{P}_i(r_1^{(i)}) - \mathcal{G}_i(t_1^{(i)}, r_1^{(i)}) - e_1^{(i)}) \text{ and}$$
$$\tilde{c}_2^{(i)} = Com(t_1^{(i)}, e_1^{(i)})$$

  and checks, if

$$C_0 \stackrel{?}{=} Com(\tilde{c}_0^{(1)}, \ldots, \tilde{c}_0^{(N)}) \text{ and}$$
$$C_2 \stackrel{?}{=} Com(\tilde{c}_2^{(1)}, \ldots, \tilde{c}_2^{(N)}).$$

- If $Ch = 2$, he parses RSP into $\Sigma, r_1^{(1)},\ t_0^{(1)},\ e_0^{(1)}, \ldots, r_1^{(N)},\ t_0^{(N)},\ e_0^{(N)}$.
  For $i = 1, \ldots N$ he computes

$$\tilde{c}_0^{(i)} = Com(r_1^{(i)}, \mathcal{G}_i(t_0^{(i)}, r_1^{(i)}) + e_0^{(i)}),$$
$$\tilde{c}_1^{(i)} = Com(t_0^{(i)}, e_0^{(i)}) \text{ and}$$
$$\tilde{c}_4^{(i)} = Com(r_1^{(i)})$$

  and checks, if

$$C_0 \stackrel{?}{=} Com(\tilde{c}_0^{(1)}, \ldots, \tilde{c}_0^{(N)}),$$
$$C_1 \stackrel{?}{=} Com(\Sigma, \tilde{c}_1^{(1)}, \ldots, \tilde{c}_1^{(N)}) \text{ and}$$
$$C_4 \stackrel{?}{=} Com(\Sigma(\tilde{c}_4^{(1)}, \ldots, \tilde{c}_4^{(N)})).$$

– If $Ch = 3$, he parses RSP into $c_3^{(\Sigma(1))}, \ldots, c_3^{(\Sigma(N))}, c_4^{(\Sigma(1))}, \ldots, c_4^{(\Sigma(N))}$.
He checks, whether

$$C_3 \stackrel{?}{=} Com(c_3^{(\Sigma(1))}, \ldots, c_3^{(\Sigma(N)}))$$

$$C_4 \stackrel{?}{=} Com(c_4^{(\Sigma(1))}, \ldots, c_4^{(\Sigma(N)}))$$

and that there are at least t indices $i \in \{1, \ldots, N\}$ with

$$c_3^{(\Sigma(i))} \neq c_4^{(\Sigma(i))}. \tag{7}$$

4.3 The Signature Scheme

By using the Fiat-Shamir paradigm we can transform this identification scheme into a threshold ring signature scheme.

The key generation process for the signature scheme works just as for the threshold identification scheme (see Algorithm 1).

To produce a threshold ring signature for a message m, the leader gathers the commitments of the signers (for all rounds) and creates the master commitments $C_0^{(1)}, C_1^{(1)}, \ldots, C_4^{(1)}, C_0^{(2)}, \ldots, C_4^{(\#\text{rounds})}$ (following step 2 of the threshold identification scheme). He then uses a hash function $\mathcal{H}$ to produce the challenge vector

$$Ch = \mathcal{H}(m \| C_0^{(1)} \| C_1^{(1)} \| \ldots \| C_4^{(1)} \| C_0^{(2)} \| \ldots \| C_4^{(\#\text{rounds})})|_{0-(2 \cdot \#rounds-1)}{}^{5}. \tag{8}$$

He sends the vector Ch to his co-signers which compute their responses. Finally the leader computes the master responses and creates the signature

$$\sigma = (C_0^{(1)} \| C_1^{(1)} \| \ldots \| C_4^{(1)} \| C_0^{(2)} \| \ldots \| C_4^{(\#\text{rounds})} || RSP_1 || RSP_2 || \ldots || RSP_{\#\text{rounds}}). \tag{9}$$

Since the challenge can be computed out of the message m and the commitments, it does not have to be part of the signature.

To verify the authenticity of a signature, the verifyer parses σ into $C_0^{(1)}$, $C_1^{(1)}$, ... , $C_4^{(1)}$, $C_0^{(2)}$, ..., $C_4^{(\#\text{rounds})}$, RSP_1, RSP_2, ... , $RSP_{\#\text{rounds}}$, computes the challenge vector (see equation (8)) and tests for each $i \in \{1, \ldots, \#\text{rounds}\}$ if RSP_i is a correct response to Ch_i according to $C_0^{(i)}, \ldots, C_4^{(i)}$.

5 Security

Theorem 1. *The scheme as described in Section 4 is a zero knowledge argument of knowledge, with a cheating probability of $\frac{3}{4}$, that the group of signers knows t vectors $s_{i_1}, \ldots, s_{i_t} \in \mathbb{F}^n \setminus \{0\}$ which fulfill $\mathcal{P}_{i_j}(s_{i_j}) = 0 \ \forall \ j = 1, \ldots, t$.*

Proof. We have to prove the three properties of completeness, soundness and zero knowledge. The completeness of the scheme follows directly from our description in the previous section. The proofs of soundness and zero-knowledge (see the next two lemmas) follow mostly the original proofs in [16]. □

Lemma 1. *(Soundness) An attacker, which is able to pass r rounds of our scheme without detection with probability $> \left(\frac{3}{4}\right)^r$, can either break the binding property of the commitment scheme or extract t vectors $s_{i_1}, \ldots, s_{i_t} \in \mathbb{F}^n \setminus \{0\}$ with $\mathcal{P}_{i_j}(s_{i_j}) = 0 \ \forall \ j = 1, \ldots t$.*

[5] For 193 rounds (corresponds to 80 bit security) the length of the hash value must be ≥ 386 bits.

Proof. Let's assume that an attacker is able to pass r rounds of the threshold ring identification scheme with probability $> \left(\frac{3}{4}\right)^r$. Then he must be able to answer all four challenges in at least one round correctly. Assuming the binding property of the commitment scheme we show that such an attacker is able to extract t vectors $s_{i_1}, \ldots, s_{i_t} \in \mathbb{F}^n \setminus \{0\}$ with $\mathcal{P}_{i_j}(s_{i_j}) = 0$ $(j = 1, \ldots, t)$.
Let's denote by $\tilde{c}_k^{(i,j)}$ the value of $\tilde{c}_k$ the verifier computes in step 6 of the protocol for the user i and the challenge j. Furthermore we have to cover the fact that in step 5 of the protocol an attacker might use different permutations to perturb the $c_3^{(i)}$ and $c_4^{(i)}$. We denote these permutations by $\Sigma^{(3)}$ and $\hat{\Sigma}^{(3)}$ respectively. Due to the binding property of the commitment scheme we get

$$\tilde{c}_0^{(1,1)} = \tilde{c}_0^{(1,2)}, \ldots, \tilde{c}_0^{(N,1)} = \tilde{c}_0^{(N,2)} \tag{10}$$

$$\Sigma^{(0)} = \Sigma^{(2)}, \tilde{c}_1^{(1,0)} = \tilde{c}_1^{(1,2)}, \ldots, \tilde{c}_1^{(N,0)} = \tilde{c}_1^{(N,2)} \tag{11}$$

$$\tilde{c}_2^{(1,0)} = \tilde{c}_2^{(1,1)}, \ldots, \tilde{c}_2^{(N,0)} = \tilde{c}_2^{(N,1)} \tag{12}$$

$$\Sigma^{(0)}(\tilde{c}_3^{(1,0)}, \ldots, \tilde{c}_3^{(N,0)}) = (\tilde{c}_3^{(\Sigma^{(3)}(1),3)}, \ldots, \tilde{c}_3^{(\Sigma^{(3)}(N),3)}) \tag{13}$$

$$\Sigma^{(2)}(\tilde{c}_4^{(1,2)}, \ldots, \tilde{c}_4^{(N,2)}) = (\tilde{c}_4^{(\hat{\Sigma}^{(3)}(1),3)}, \ldots, \tilde{c}_4^{(\hat{\Sigma}^{(3)}(N),3)}) \tag{14}$$

The equations (10), (11) and (12) yield (for all $i = 1, \ldots, N$)

$$(r_1^{(i,1)}, \mathcal{P}_i(r_1^{(i,1)}) - \mathcal{G}_i(t_1^{(i,1)}, r_1^{(i,1)} - e_1^{(i,1)})) = (r_1^{(i,2)}, \mathcal{G}_i(t_0^{(i,2)}, r_1^{(i,2)}) + e_0^{(i,2)}) \tag{15}$$

$$(r_0^{(i,0)} - t_0^{(i,0)}, \mathcal{P}_i(r_0^{(i,0)}) - e_1^{(i,0)}) = (t_0^{(i,2)}, e_0^{(i,2)}) \tag{16}$$

$$(t_1^{(i,0)}, e_1^{(i,0)}) = (t_1^{(i,1)}, e_1^{(i,1)}) \tag{17}$$

As shown in [16], these three equations lead, for every $i = 1, \ldots, N$, to a solution $\hat{s}_i = r_0^{(i,0)} + r_1^{(i,2)}$ of $\mathcal{P}_i(x) = 0$. We show that at least t of these solutions are $\neq 0$.
To pass challenge 3 of the protocol, the test (equation (7)) has to be fulfilled for at least t indices $i_1, \ldots, i_t$. With our notation this can be written as

$$\tilde{c}_3^{(\Sigma^{(3)}(i_j),3)} \neq \tilde{c}_4^{(\hat{\Sigma}^{(3)}(i_j),3)} \ \forall\ j = 1, \ldots, t.$$

Due to $\Sigma^{(0)} = \Sigma^{(2)} =: \Sigma$ (c.f. equation (11)) and equation (13) and (14) this is equivalent to

$$\tilde{c}_3^{(\Sigma(i_j),0)} \neq \tilde{c}_4^{(\Sigma(i_j),2)} \ \forall\ j = 1, \ldots, t,$$

which again is computationally equivalent to

$$r_0^{(\Sigma(i_j),0)} \neq r_1^{(\Sigma(i_j),2)} \ \forall\ j = 1, \ldots, t.$$

Finally, we see that this is equivalent to

$$\hat{s}_{\Sigma(i_j)} = r_0^{(\Sigma(i_j))} + r_1^{(\Sigma(i_j),2)} \neq 0 \ \forall\ j = 1, \ldots, t,$$

which means that the attacker has found t vectors $\hat{s}_{\Sigma(i_j)} \in \mathbb{F}^n \setminus \{0\}$ with $\mathcal{P}_{\Sigma(i_j)}(\hat{s}_{\Sigma(i_j)}) = 0$ $j = 1, \ldots, t$. □

Remark: Lemma 1 states that, in order to pass r rounds of the scheme with probability $\geq \left(\frac{3}{4}\right)^r$, an attacker has to break at least one instance of the MQ-Problem. The same holds for a group of less than t signers.

Lemma 2. *(Zero-knowledge) Our threshold ring identification scheme is statistically zero knowledge if the commitment scheme Com is statistically hiding.*

Proof. Let $\mathcal{S}$ be a simulator which knows all the public keys $\mathcal{P}_i$ $(i = 1, \ldots, N)$, but does not know t of the private keys. W.l.o.g. we can assume that $\mathcal{S}$ does not know any of the private keys. Therefore we can neglect the interactions between the signers and the leader L and $\mathcal{S}$ simulates just the interactions between the leader and a cheating verifier CV. At the beginning, $\mathcal{S}$ chooses a value $Ch^\star \in \{0, 1, 2, 3\}$. $Ch^\star$ is a prediction, of what challenge the verifier will not choose.
If $Ch^\star = 3$, the simulator sets all $\tilde{s}_i$ $(i = 1, \ldots, N)$ to zero, chooses randomly $\tilde{r}_0^{(i)}$, $\tilde{t}_0^{(i)} \in \mathbb{F}^n$ and $\tilde{e}_0^{(i)} \in \mathbb{F}^m$ and computes $\tilde{r}_1^{(i)} = \tilde{s}_i - \tilde{r}_0^{(i)}$, $\tilde{t}_1^{(i)} = \tilde{r}_0^{(i)} - \tilde{t}_1^{(i)}$ and $\tilde{e}_1^{(i)} = \mathcal{P}_i(\tilde{r}_0^{(i)}) - \tilde{e}_0^{(i)}$. After that he creates the commitments $\tilde{c}_0^{(i)}$, $\tilde{c}_1^{(i)}$, $\tilde{c}_2^{(i)}$, $\tilde{c}_3^{(i)}$ and $\tilde{c}_4^{(i)}$ $(i = 1, \ldots, N)$, chooses a random permutation $\Sigma \in S_N$ and creates the master commitments $\tilde{C}_0$, $\tilde{C}_1$, $\tilde{C}_2$, $\tilde{C}_3$ and $\tilde{C}_4$ as described in step 2 of the protocol. It is obvious that $\mathcal{S}$ can answer the challenges 0, 1 and 2 correctly.
If $Ch^\star \neq 3$, $\mathcal{S}$ chooses $\tilde{s}_i$, $\tilde{r}_0^{(i)}$, $\tilde{t}_0^{(i)} \in \mathbb{F}^n$ and $\tilde{e}_0^{(i)} \in \mathbb{F}^m$ at random and computes $\tilde{r}_1^{(i)} = \tilde{s}_i - \tilde{r}_0^{(i)}$ and $\tilde{t}_1^{(i)} = \tilde{r}_0^{(i)} - \tilde{t}_0^{(i)}$. If $Ch^\star = 0$, it computes $\tilde{e}_1^{(i)} = \mathcal{P}_i(\tilde{r}_0^{(i)}) + \mathcal{P}_i(\tilde{s}_i) - \tilde{e}_0^{(i)}$, otherwise $\tilde{e}_1^{(i)} = \mathcal{P}_i(\tilde{r}_0^{(i)})$. If $Ch^\star = 2$, $\mathcal{S}$ computes $\tilde{c}_0^{(i)} = Com(\tilde{r}_1^{(i)}, -\mathcal{P}_i(\tilde{r}_1^{(i)}) - \mathcal{G}_i(\tilde{t}_1^{(i)}, \tilde{r}_1^{(i)}) - \tilde{e}_1^{(i)})$, otherwise $\tilde{c}_0^{(i)} = Com(\tilde{r}_1^{(i)}, \mathcal{G}_i(\tilde{t}_0^{(i)}, \tilde{r}_1^{(i)}) + \tilde{e}_0^{(i)})$. It computes $\tilde{c}_1^{(i)} = Com(\tilde{t}_0^{(i)}, \tilde{e}_0^{(i)})$, $\tilde{c}_2^{(i)} = Com(\tilde{t}_1^{(i)}, \tilde{e}_1^{(i)})$, $\tilde{c}_3^{(i)} = Com(\tilde{r}_0^{(i)})$ and $\tilde{c}_4^{(i)} = Com(\tilde{r}_1^{(i)})$, chooses a random permutation Σ and creates the master commitments $\tilde{C}_0$, $\tilde{C}_1$, $\tilde{C}_2$, $\tilde{C}_3$ and $\tilde{C}_4$ following step 2 of the protocol.
If $Ch^\star \neq Ch$, $\mathcal{S}$ is able to answer Ch correctly.
Therefore, $\mathcal{S}$ outputs a valid transcript of one round of the threshold identification scheme with probability $\frac{3}{4}$. Furthermore, this output is statistically close to a real transcript of the scheme. □

Remark: It is also impossible for the leader to get information about the signers' private keys. The interactions between the leader and the signers follow the identification scheme of [16], which is zero knowledge. Since the commitment scheme is statistically hiding, the two additional commitments sent by each signer to the leader make no difference to this fact.

Corollary 1. *The resulting threshold ring signature scheme obtained from the application of the Fiat-Shamir paradigm on our threshold ring identification scheme (see Subsection 4.3) is existentially unforgeable under chosen message attacks (see Subsection 2.1) in the random oracle model.*

Proof. The proof runs straightforward as shown in [14]. □

Theorem 2. *The so obtained threshold ring signature scheme is unconditionally source hiding.*

Proof. For the challenges 0,1 and 2 the responses of both a signer and a non-signer are completely indistinguishable, since r_0, t_0 and e_0 are chosen uniformly at random and therefore the responses are random, too. So, the only possibility for the verifier to identify non-signers is challenge 3. In this case he checks whether the two values $Com(r_0)$ and $Com(r_1)$ are equal, which implies $r_0 = r_1$ and therefore $s = r_0 + r_1 = 0$. But in this case the possible signers are mixed by a random permutation Σ. Since the verifier has no access to this permutation, he is not able to identify non-signers. □

6 Parameters and Comparison

For our threshold ring identification (and signature) scheme we propose the same parameters as in [16], namely

$$\mathbb{F} = GF(2), \ (m, n) = (80, 84).$$

Therefore, the public key size of our scheme is $m \cdot \frac{n \cdot (n+3)}{2} \cdot N = 292,320 \cdot N$ bit or $35.7 \cdot N$ kB.

The communication cost of one round of our scheme depends on the challenge. Between the signers and the leader the communication cost per round lies between $800 \cdot (t-1)$ (for $Ch = 3$ we don't need a response) and $1050 \cdot (t-1)$ bit.

Between the leader and the verifier the communication cost is given by

$$802 + 248 \cdot N \text{ bit} \leq \text{ communication cost per round} \leq 802 + 320 \cdot N \text{ bit}.$$

The overall communication cost per round is on average

$$988 \cdot (t-1) + 866 + 266 \cdot N \text{ bit}.$$

6.1 Reducing the communication cost

To reduce the cost of communication, one can use the following trick:

In step 2 of the protocol, the leader sends $COM = Com(C_0, C_1, C_2, C_3, C_4)$ to the verifier. The responses in step 5 are changed as follows

- If $Ch = 0$, $RSP = (\Sigma, Rsp_1, \ldots, Rsp_N, C_0, C_4)$
- If $Ch = 1$, $RSP = (Rsp_1, \ldots, Rsp_N, C_1, C_3, C_4)$
- If $Ch = 2$, $RSP = (\Sigma, Rsp_1, \ldots, Rsp_N, C_2, C_3)$
- If $Ch = 3$, $RSP = (\Sigma(c_3^{(1)}, \ldots, c_3^{(N)}), \Sigma(c_4^{(1)}, \ldots, c_4^{(N)}), C_0, C_1, C_2)$

In step 6 of the protocol, the verifier computes the remaining C_i's and checks, if COM was created honestly. By doing so, one can reduce the communication cost per round by on average 240 bit. Furthermore it is possible to perform all 73 (or 193) rounds of the scheme on parallel. Therewith, the communication cost can be reduced further by $160 \cdot (\#rounds - 1)$ bit.
Table 1 shows the reduced communication cost of our scheme for different parameters.

cheating probability	2^{-30}	2^{-80}
(N, t)	73 rounds	193 rounds
$(50, 30)$	378 kB	998 kB
$(100, 50)$	672 kB	1777 kB

Table 1. communication cost

6.2 Reducing the signature length

The trick mentioned in the previous subsection can also be used to reduce the signature length of our scheme.
After having computed the master commitments $C_0^{(1)}, C_1^{(1)}, \ldots, C_4^{(1)}, C_0^{(2)}, \ldots, C_4^{(\#\text{rounds})}$ (see Subsection 4.3), the leader uses a hash function $\mathcal{H}$ to compute

$$COM = \mathcal{H}(C_0^{(1)} \| \, C_1^{(1)} \| \ldots \| C_4^{(1)} \| C_0^{(2)} \| \ldots \| \, C_4^{\#\text{rounds}}) \tag{18}$$

The challenge vector is then obtained by

$$Ch = \mathcal{H}(m \| COM) \tag{19}$$

for a message m. By doing so, the leader gets a signature of the form

$$\sigma = (COM || RSP_1 || RSP_2 || \ldots || RSP_{\#\text{rounds}}), \tag{20}$$

where the responses RSP_i are computed as shown in Subsection 6.1.
To verify the authenticity of a signature, the verifyer computes for each round the remaining C_i

and finally checks the correctness of COM.

The length of a so obtained signature is given by

$$|\sigma| = 160 + \#\text{rounds} \cdot (464 + 266 \cdot N) \text{ bits.} \tag{21}$$

In particular, the signature length is independent of t and linear in N. Table 2 shows the signature lengths of our scheme for different values of N. The signature lengths shown in Table 2 correspond

(N, t)	signature length
$(50, 30)$	324 kB
$(100, 50)$	638 kB

Table 2. Signature lengths

to 193 rounds of the scheme, which leads to a security level of 80 bit.

6.3 Computational Cost

The computationally most expensive operations in our scheme are the polynomial evaluations of the systems $\mathcal{P}_i$ and $\mathcal{G}_i$ of which the latter can be performed by three evaluations of $\mathcal{P}_i$. Therefore we have on average $\frac{23}{4} \cdot N$ evaluations of systems of m equations in n variables per round. For the whole scheme (193 rounds), we get a computational effort of $1110 \cdot N$ polynomial evaluations and $1448 \cdot (N + 1)$ hash function evaluations. In particular, the computational complexity of our scheme is independent of t and linear in N.

6.4 Comparison

Table 3 compares our scheme with other existing Post-Quantum threshold ring signature schemes.

Security	Scheme	TRSS-C [1]	TRSS-L [7]	Our scheme
2^{80}	hash length	160 bit	160 bit	160 bit
	rounds	140	80	193
	public key	1.5 MB[1]	7.8 MB[1]	3.5 MB[2]
	private key	700 bit	1280 bit	84 bit
	signature length	1.4 MB	14.8 MB	0.64 MB
2^{100}	hash length	224 bit	224 bit	224 bit
	rounds	190	100	256
	public key	2.2 MB[1]	17.0 MB[1]	6.9 MB [2]
	private key	850 bit	1728 bit	105 bit
	signature length	2.4 MB	26.7 MB	1.07 MB

[1] Using a PRNG these numbers can be reduced to several kB.
[2] By viewing the homogeneous quadratic part of the public key as system parameter (c.f. Subsection 4.2) the public key sizes can be reduced to 82 kB (80 bit security) or 117 kB (100 bit security) respectively.

Table 3. Comparison of different threshold ring signature schemes (for (N, t)=$(100, 50)$)

As Table 3 shows, our scheme produces shorter signatures than code- or lattice based constructions. The reason for this is that the responses RSP in our threshold identification scheme (see Section 4) are relatively short (on average $266 \cdot N$ bit compared to $828 \cdot N$ bit for the code-based scheme). This has a much greater influence on the signature length than the number of rounds, which is larger in our scheme.

Remark: It is also possible to extend the 5-pass version of the MQ-identification scheme of [16] to a threshold ring identification (and signature) scheme. But this does not lead to a more efficient solution.

7 Conclusion and Future Work

In this paper we proposed a new threshold ring identification and signature scheme, whose security is based solely on the MQ-problem. Our scheme is the first multivariate scheme of this kind and the first multivariate signature scheme with special properties. Furthermore it offers provable security, which is quite a rare fact in multivariate cryptography. Despite the fact that our scheme requires more rounds than other post-quantum threshold ring signature schemes, the signatures are significantly smaller. The scheme also enjoys smaller secret keys.

As Future work we plan to create other provable secure multivariate signature schemes with special properties (forward secure, identity-based, etc.)

8 Acknowledgements

We thank Pierre-Louis Cayrel and the anonymous referees of PKC 2012 for helpful comments. The first author thanks the Horst Görtz Foundation for financial support. The second author is funded by DFG grant BU 630/22-1.

References

[1] C. Aguilar, P.L. Cayrel, P. Gaborit and F. Laguillaumie: A New Efficient Threshold Ring Signature Scheme Based on Coding Theory. IEEE Transactions on Information Theory 57(7), pp. 4833-4842 (2011).
[2] D.J. Bernstein, J. Buchmann and E. Dahmen (eds.): Post Quantum Cryptography. Springer 2009.
[3] L. Bettale, J.C. Faugère and L. Perret: Hybrid approach for solving multivariate systems over finite fields. Journal of Math. Cryptology, pp. 177-197, 2009
[4] A. Bogdanov, T. Eisenbarth, A. Rupp, and C. Wolf. Time-area optimized public-key engines: -cryptosystems as replacement for elliptic curves? CHES, LNCS vol. 5154, pp. 45-61. Springer, 2008.
[5] X. Boyen: Mesh Signatures. EUROCRYPT 2007, LNCS vol. 4515, pp. 210-227, Springer 2007.
[6] E. Bresson, J. Stern and M. Szydlo: Threshold ring signatures and their application to ad-hoc groups. CRYPTO 2002, LNCS vol. 2442, pp. 465 - 480, Springer 2002
[7] P.L. Cayrel, R. Lindner, M. Rückert and R. Silva: A Lattice-Based Threshold Ring Signature Scheme. LATINCRYPT 2010, LNCS vol. 6212, pp. 255 - 272, Springer 2010.
[8] A.I.T. Chen, M.-S. Chen, T.-R. Chen, C.-M. Cheng, J. Ding, E. L.-H. Kuo, F. Y.-S. Lee, and B.-Y. Yang. SSE implementation of multivariate pkcs on modern x86 cpus. CHES 2009, LNCS vol. 5747, pp. 33–48. Springer 2009.
[9] J. Ding, J.E. Gower, D. Schmidt: Multivariate Public Key Cryptosystems. Springer 2006.
[10] J.C. Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). ISSAC 2002, pp. 75 - 83. ACM Press 2002.
[11] A. Fiat and A. Shamir: How to Prove Yourself. CRYPTO 1986, LNCS vol. 263, pp. 186 - 194, Springer 1986
[12] M. R. Garey and D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Company, 1979
[13] J.K. Liu, V.K. Wei and D.S. Wong: A Separable Threshold Ring Signature Scheme. ICISC 2003, LNCS vol. 2971, pp. 352 - 369, Springer 2003.
[14] P. Pointcheval and J. Stern: Security proofs for signature schemes. EUROCRYPT 96, LNCS vol. 1070, pp. 387 - 398, Springer 1996.
[15] R. Rivest, A. Shamir and Y. Tauman: How to leak a secret. ASIACRYPT 2001, LNCS vol. 2248, pp. 552 - 565, Springer 2001.

[16] K. Sakumoto, T. Shirai and H. Hiwatari: Public-Key Identification Schemes based on Multivariate Quadratic Polynomials. CRYPTO 2011, LNCS vol. 6841, pp. 706 - 723, Springer 2011.

[17] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal on Computing, 26(5):1484-1509, Oct. 1997.