Short Lattice-based One-out-of-Many Proofs and Applications to Ring Signatures*

Muhammed F. Esgin[1,2], Ron Steinfeld[1], Amin Sakzad[1], Joseph K. Liu[1], and Dongxi Liu[2]

[1] Faculty of Information Technology, Monash University, Australia
[2] Data61, CSIRO, Australia
{Muhammed.Esgin,Ron.Steinfeld,Amin.Sakzad,Joseph.Liu}@monash.edu
Dongxi.Liu@data61.csiro.au

Abstract. In this work, we construct a short one-out-of-many proof from (module) lattices, allowing one to prove knowledge of a secret associated with one of the public values in a set. The proof system builds on a combination of ideas from the efficient proposals in the discrete logarithm setting by Groth and Kohlweiss (EUROCRYPT '15) and Bootle et al. (ESORICS '15), can have logarithmic communication complexity in the set size and does not require a trusted setup.
Our work resolves an open problem mentioned by Libert et al. (EUROCRYPT '16) of how to efficiently extend the above discrete logarithm proof techniques to the lattice setting. To achieve our result, we introduce new technical tools for design and analysis of algebraic lattice-based zero-knowledge proofs, which may be of independent interest.
Using our proof system as a building block, we design a short ring signature scheme, whose security relies on "post-quantum" lattice assumptions. Even for a very large ring size such as 1 billion, our ring signature size is only 3 MB for 128-bit security level compared to 216 MB in the best existing lattice-based result by Libert et al. (EUROCRYPT '16).

Keywords: lattice-based cryptography, zero-knowledge proof, one-out-of-many proof, ring signature

1 Introduction

In the last decade, lattice-based cryptography has seen a great interest with many new applications developed rapidly. Although it offers solutions even to problems which long seemed elusive, there is still a gap in some areas where lattice-based cryptographic proposals are not efficient enough for practical use and even fall far behind their number theoretic counterparts in terms of efficiency. One important example for such a case is zero-knowledge proofs (ZKPs). It seems that lattice-based cryptography does not agree well with ZKPs and extending the existing number theoretic proposals to the lattice setting is quite challenging.

* This is the full version of the paper accepted to ACNS 2019.

A particular example is one-out-of-many proofs where the prover's goal is to prove knowledge of an opening of a commitment within a set of commitments without revealing which one he has. Groth and Kohlweiss [15] and Bootle et al. [8] gave very efficient constructions with logarithmic (log) communication complexity in the size of the set of commitments based on decisional Diffie-Hellman assumption. Their protocols also lead to very efficient ring signatures without trusted setup[3], where a signatory signs a message on behalf of a group of users (referred as a *ring*). The idea behind obtaining a ring signature from a one-out-of-many proof works as follows. Users commit to their secret keys, resulting in the users' public keys. Then, the signatory proves (in a non-interactive fashion using Fiat-Shamir heuristic) that he knows an opening (i.e., the secret key) of one of the commitments (i.e., the corresponding public keys) used to create the ring signature. Ring signatures are important tools used in e-voting systems and cryptocurrencies to provide anonymity. Especially in the case of cryptocurrencies, an important aspect is the ring signature size, which makes the schemes in [15, 8] very attractive on a large scale. However, these proposals in [15, 8] do not offer post-quantum security as they are in the discrete logarithm (DL) setting.

On the side of lattice-based cryptography, a promising candidate for post-quantum security, efficient designs targeting the same problems do not currently exist. There has not been a successful extension of the ideas in [15, 8] to the lattice setting, and other approaches proposed so far resulted in very inefficient schemes that are far from offering practical usability. To illustrate, while [8] gives constructions in the order of a few KB even for very large ring sizes, the current shortest log-sized ring signature from lattices by Libert et al. [19] results in a ring signature of size 75MB for around a thousand ring members and a security level of 128 bits. It is therefore tempting to realise the ideas in [15, 8] using lattice-based techniques, but, as we discuss next, this is far from trivial. In this work, we tackle this problem and design short one-out-of-many proofs and ring signatures from (module) lattices by introducing new tools for the design and analysis of algebraic lattice-based ZKPs (see Section 3).

For some practical applications, one requires *linkability* between ring signatures generated using the same secret key. This is often referred to as a *linkable ring signature* [21], which is useful in e-voting systems (e.g., see [12]) and blockchain confidential transactions (e.g., see [27, 31, 32]). Our ring signature can be extended to provide linkability using the same technique as in [35, 4].

1.1 Technical difficulties

The starting point of our protocol is the works by Groth and Kohlweiss [15] and Bootle et al. [8], instantiated using Pedersen commitment as a core ingredient. As also noted in [19] and [4], it is not straightforward to design lattice-based one-out-of-many proofs and ring signatures from the ideas in [15, 8]. One can see [35] for an attempt to design a (linkable) lattice-based ring signature based

[3] There are some constructions of ring signatures that give a constant size signature but require a trusted setup.

on [15]. The authors of [35] claim that the anonymity and unforgeability of their scheme follow from the framework of [15], provided that a perfectly hiding and computationally binding commitment scheme is used. However, as we show here, there are many issues to be addressed if one aims to use the ideas from [15, 8] in the lattice setting, whereas [35] did not go into details of how these issues are to be solved. To begin with, the *valid* input space of lattice-based commitment schemes is a proper subset of $\mathbb{Z}_q^v$ for some $v \geq 1$ (or the underlying polynomial ring $R_q^v = \mathbb{Z}_q[X]/(X^d + 1)$ in the case of ring variants) consisting of vectors of *small* elements unlike their number-theoretic counterparts such as Pedersen commitment accepting any element in $\mathbb{Z}_q^v$. This restriction prevents straightforward adaptation of number-theoretic results, and in fact there is a crucial difference between the relations of the lattice-based and DDH-based one-out-of-many proofs (see Remark 1 in Section 4). Furthermore, extending [15] alone does not yield *efficient* lattice-based ring signatures even if the security issues in the lattice setting are addressed properly.

Let us briefly discuss the main technical difficulties our new techniques enable us to overcome in extending [15, 8] to the lattice setting. We denote the public set size for one-out-of-many proof (or the ring size for the ring signature) by N, and $C = \mathrm{Com}_{ck}(m\ ;\ r)$ as a commitment to a message m with randomness r using a commitment key ck. A pair of *acceptable* values (m', r') such that $C = \mathrm{Com}_{ck}(m'\ ;\ r')$ is called an opening of C. The reader unfamiliar with the general concepts of Σ-protocols is referred to Section 2.4.

1. **Growth of extracted witness size:** As mentioned previously, lattice-based commitment schemes accept only elements of bounded size as valid openings. We show that the sizes of extracted witnesses, which will be openings of some commitments, grow rapidly with the size of challenge difference inverses in the framework of [15, 8] (see Section 3.2). In particular, we show that if one works over a ring $R_q = \mathbb{Z}_q[X]/(X^d + 1)$, the growth can be made to be of the form $\Gamma = d^{\log N}$ (see Section 5). Letting $d = 2^{10}$ with $N = 2^{20}$ users, Γ (and, in turn, q) reaches 200 bits without any additional considerations.
2. **(Small) challenge space size:** In connection with the above difficulty, we need to find a challenge space where the sizes of challenge difference inverses are guaranteed to be small. Unfortunately, we cannot find such a space with exponentially many elements, restricting us to a small challenge space. A simple (commonly used) possible option is to use binary challenges. However, the scheme presented in [8] requires at least 3 distinct challenges to extract a witness, making that option ineligible. In fact, the main protocols in [15, 8] even require up to $\log_2 N + 1$ challenges for witness extraction, which means that several *forkings* are required in the unforgeability proof of the ring signature. This fact combined with a small challenge space causes major issues in the unforgeability proof (see proof of Theorem 3). For example, one cannot simply rely on a commonly used Forking Lemma from [10].
3. **Proof of commitment to a binary value over R_q:** When working over the ring $R_q = \mathbb{Z}_q[X]/(X^d + 1)$, the following statement, which is typically used to prove that a value is binary, does not necessarily hold: $x(x - 1) = 0$

Table 1: Comparison of ring signature sizes at $\lambda = 128$-bit security with N ring members. For [19], we use the same system parameters given in [19] for 80-bit security, but only increase the number of protocol repetitions to reach 2^{-128} soundness error. See Section 6.1 for detailed parameter setting.

N	2^6	2^8	2^{10}	2^{12}	2^{16}	2^{20}	2^{30}
[19] (sign. size in KB)	47294	61438	75582	89726	118014	146303	217023
Our Work (sign. size in KB)	774	881	1021	1178	1487	1862	3006

$\implies x \in \{0, 1\}$. This is because there exist zero divisors in R_q (unlike the field $\mathbb{Z}_q$ used in DL-based schemes). Hence, straightforward proofs of $x(x-1) = 0$ does not guarantee that x is binary (see Section 3.1).

4. **Soundness gap:** In common with some other lattice-based proofs, our protocol has the so-called *soundness gap* unlike DL-based schemes. That is, the extractor recovers the openings of $\gamma \cdot \mathrm{Com}_{ck}(m\ ;\ r)$ instead of the actual commitments of the form $\mathrm{Com}_{ck}(m\ ;\ r)$. This makes things more complicated in the soundness proofs (see the proofs of Theorems 1 and 2) and requires one to be careful in protocol's application to a ring signature as the extractor is never guaranteed to recover the openings of the actual commitments used in the protocol (see the proof of Theorem 3).

1.2 Our contributions

New technical tools for algebraic protocols and design of short lattice-based one-out-of-many proofs and ring signatures. By now, it is clear that extending the works [15, 8] to the lattice setting is far from being trivial, which was indeed stated as an open problem in [19, 4]. Our main contributions in this work are the introduction of new technical tools for the design and analysis of algebraic protocols from lattices (Section 3) and the design of short (sublinear-sized) one-out-of-many proofs and ring signature schemes from (module) lattices (Sections 5 and 6). It is worth emphasising that our proposal is not a direct adaptation of either [15] or [8], but rather carefully combines ideas from both in a way suitable in the lattice scenario, and also that the technical difficulties mentioned in Section 1.1 do not allow straightforward extension of [15] or [8].

As shown in Table 1, our ring signature achieves a dramatic improvement in terms of length over the shortest existing log-sized result from lattices by Libert et al. [19], where the improvement is almost two orders of magnitude.[4] Moreover, an important feature of our constructions is that a modulus q of a special form (such as $q \equiv 17 \bmod 32$ as in [13]) is not required, which allows the use of fast computation algorithms such as Number Theoretic Transform (NTT).

A series of previous proposals of group/ring signatures (e.g., [19, 18, 20]) rely on combinatorial Stern-like protocols [30]. Even though these protocols offer a

[4] Our scheme, like [19], is only analyzed in the classical random oracle model (ROM) (rather than quantum ROM). Also, note that the linear-sized ring signature schemes are inherently long for large ring sizes.

range of functionalities, all of them have very long signature sizes that seem too large for practical use. Our new technical tools developed in Section 3 introduce new directions for efficient applications of algebraic lattice-based techniques to areas where lattice-based proposals fall behind their number-theoretic counterparts. The protocol structure, for whose construction our new tools provide efficient techniques, is also involved in advanced ZKPs such as arithmetic circuit arguments [9] and Bulletproofs [11]. Hence, our new tools may be of independent interest, especially for the extension of other advanced ZKPs in the DL setting to the lattice setting. In fact, the issues in [35] can be fixed using our techniques, but the revised scheme is unlikely to be more efficient than our work.

Exploiting module variants of standard lattice assumptions for efficiency purposes. Another contribution of our work is to show that the use of Module-SIS (M-SIS) problem [17] (over SIS or Ring-SIS) opens the door for significant efficiency improvements by allowing us to tradeoff extracted witness size growth (and hence signature length) against computational efficiency. To the best of our knowledge, this is the first time a lattice-based ZKP has been instantiated based on M-SIS to gain such an efficiency improvement.[5]

In the Ring-LWE setting, *monomial challenges*, $X^i \in R = \mathbb{Z}[X]/(X^d + 1)$, was introduced in [6] to enable a challenge space of size $2d$ with the property that the doubled inverse of the difference of such challenges have small norm. The three methods introduced in Section 3 provide an in-depth analysis of the use of monomial challenges in a more generalized setting of $(k+1)$-special sound protocols. We believe that the combination of using monomial challenges together with M-SIS to fine-tune the parameters for efficiency purposes holds great potential to be investigated through further research in lattice-based cryptography.

Paper Organization. Section 2 discusses some preliminaries. Our new tools for the design and analysis of algebraic protocols from lattices are introduced in Section 3. Section 4 and Section 5 cover our binary proof and one-out-of-many proof, respectively. Our compact lattice-based ring signature is then provided in Section 6. We provide a brief discussion on the issues in [35] in Appendix A, more detailed related work in Appendix B, and rigorous definitions of ring signatures in Appendix D. Some of the proofs of our new results are deferred to Appendix E.

2 Preliminaries

2.1 Notation

Throughout the manuscript, bold-face lower-case letters like $\boldsymbol{x}$ are used to denote column vectors and bold-face capital letters like $\boldsymbol{A}$ to denote matrices with $\boldsymbol{I}_n$ being the n-dimensional identity matrix. $(\boldsymbol{x}, \boldsymbol{y})$ denotes appending the vector $\boldsymbol{y}$ to the vector $\boldsymbol{x}$. $\mathbb{Z}_q = \mathbb{Z}/q\mathbb{Z}$ denotes the ring of integers modulo q represented

[5] M-SIS is used usually (e.g. in [13]) to fix the ring dimension d and to avoid the need for a change of it to accommodate new security parameters. It does not have a significant effect on efficiency due to extracted witness norm unlike in our case.

by the range $\left[-\frac{q-1}{2}, \frac{q-1}{2}\right]$ where q is an odd positive integer. We usually work with the Euclidean norm denoted by $\|\cdot\|$ unless otherwise stated. For a vector $\boldsymbol{x} = (x_0, \ldots, x_{n-1})$ and a polynomial $p(X) = a_0 + a_1 X + \cdots + a_{n-1}X^{n-1}$ in variable X, the Euclidean norm is defined as $\|\boldsymbol{x}\| = \sqrt{\sum_{i=0}^{n-1} x_i^2}$ and $\|p\| = \sqrt{\sum_{i=0}^{n-1} a_i^2}$, respectively. The infinity norm of p is $\|p\|_\infty = \max_i |a_i|$. For a vector $\boldsymbol{p} = (p_0, \ldots, p_{m-1})$ of polynomials, $\|\boldsymbol{p}\| = \sqrt{\sum_{i=0}^{m-1} \|p_i\|^2}$. Also, we denote the main security parameter by λ and adapt $\lambda = 128$ when instantiating parameters. $a \leftarrow \mathcal{Z}$ means a is chosen uniformly from a set $\mathcal{Z}$. We use the same notation to sample a from a distribution $\mathcal{Z}$. In the case that $\mathcal{Z}$ is an algorithm, the same notation is used to denote that the algorithm outputs a. Logarithms are base 2 unless explicitly specified otherwise. S_c defines the set of all polynomials in R with infinity norm at most $c \in \mathbb{Z}^+$. We write $\boldsymbol{p} \leftarrow S^{md}$ to indicate that $\boldsymbol{p} \in R^m$ is a vector of m polynomials where each coefficient is sampled from a set S (i.e., md coefficients are sampled in total). We say that a function $\nu(\lambda)$ is negligible (denoted by $\nu = \text{negl}(\lambda)$) if $\nu(\lambda) < 1/\lambda^c$ for any $c > 0$ and all sufficiently large λ. $[a, b]$ denotes the set of integers $\{a, a+1, \ldots, b-1, b\}$.

2.2 Module-SIS, Module-LWE problems and commitment scheme

In our schemes, we work over a ring R_q and rely on the hardness of Module-SIS (M-SIS)[6] and Module-LWE (M-LWE) problems [17] defined below.

Definition 1 (M-SIS$_{n,m,q,\theta}$). *Let $R_q = \mathbb{Z}_q[X]/(X^d+1)$. Given $\boldsymbol{A} = [\,\boldsymbol{I}_n \,\|\, \boldsymbol{A'}\,] \in R_q^{n\times m}$ where each component of $\boldsymbol{A'}$ is chosen independently from the uniform distribution, find $\boldsymbol{z} \in R_q^m$ such that $\boldsymbol{A}\boldsymbol{z} = \boldsymbol{0} \bmod q$ and $0 < \|\boldsymbol{z}\| \leq \theta$.*

For simplicity, we consider a special case of M-LWE problem where each error and secret key coefficient is sampled uniformly from $\{-\mathcal{B}, \ldots, \mathcal{B}\}$ for some $\mathcal{B} \in \mathbb{Z}^+$. A more special case of $\mathcal{B} = 1$ is commonly practised in recent lattice-based proposals [3, 23, 14], and our results can be easily extended to a case with a discrete Gaussian distribution.

Definition 2 (M-LWE$_{n,m,q,\mathcal{B}}$). *Let $R_q = \mathbb{Z}_q[X]/(X^d+1)$ and $\boldsymbol{s} \leftarrow S_\mathcal{B}^n$ be a secret key. Define $\texttt{LWE}_{q,\boldsymbol{s}}$ as the distribution obtained by sampling $e \leftarrow S_\mathcal{B}$, $\boldsymbol{a} \leftarrow R_q^n$ and returning $(\boldsymbol{a}, \langle \boldsymbol{a}, \boldsymbol{s}\rangle + e)$. Given m samples from either $\texttt{LWE}_{q,\boldsymbol{s}}$ or $\mathcal{U}(R_q^n, R_q)$, the problem asks to distinguish which is the case.*

We use the following lattice-based commitment scheme that allows commitment to multiple messages, and is additively homomorphic. Following the standard notions, *hiding* property requires that it is hard to distinguish between commitments to two distinct message-randomness pairs, and *strong binding* property (which is stronger than the standard binding property) dictates that it is hard to find two distinct *valid* openings (message-randomness pairs) of a commitment. In common with similar lattice-based commitment schemes (see, e.g., [3]

[6] As in [3], we define M-SIS in "Hermite normal form", which is equivalent to M-SIS with completely random $\boldsymbol{A}$.

for more discussion), the opening algorithm of the commitment scheme has an additional input $y \in R_q$, called the relaxation factor, and the opening message-randomness pair is required to have a bounded norm. The latter is needed to relate the binding property to the M-SIS problem (as given in Lemma 1). Thus, we introduce a parameter $T_{\text{com}} \in \mathbb{R}^+$ and say T_{com}-binding where T_{com} serves as an upperbound on the norm of a valid opening message-randomness pair.

- $\mathsf{CKeygen}(1^\lambda)$: Pick $\boldsymbol{G}'_r \leftarrow R_q^{n\times(m-n)}$, $\boldsymbol{G}_m \leftarrow R_q^{n\times v}$ and set $\boldsymbol{G}_r = [\,\boldsymbol{I}_n \,\|\, \boldsymbol{G}'_r\,]$. Output $ck = \boldsymbol{G} = [\,\boldsymbol{G}_r \,\|\, \boldsymbol{G}_m\,] \in R_q^{n\times(m+v)}$.
- $\mathsf{Commit}_{ck}(\boldsymbol{m})$: Pick $\boldsymbol{r} \leftarrow S_{\mathcal{B}}^m$. Output $\text{Com}_{ck}(\boldsymbol{m}\ ;\ \boldsymbol{r}) = \boldsymbol{G}\cdot(\boldsymbol{r},\boldsymbol{m}) = \boldsymbol{G}_r\cdot\boldsymbol{r} + \boldsymbol{G}_m\cdot\boldsymbol{m}$.
- $\mathsf{Open}_{ck}(C,(y,\boldsymbol{m}',\boldsymbol{r}'))$: For $y \in R_q$, if $\text{Com}_{ck}(\boldsymbol{m}'\ ;\ \boldsymbol{r}') = yC$ and $\|(\boldsymbol{r}',\boldsymbol{m}')\| \le T_{\text{com}}$, return 1. Otherwise, return 0.

Lemma 1. *The commitment scheme defined above is computationally hiding if M-LWE$_{m-n,n,q,\mathcal{B}}$ is hard. It is also computationally strong T_{com}-binding with respect to the same relaxation factor y if M-SIS$_{n,m+v,q,2T_{com}}$ problem is hard.*

The proof is deferred to Appendix E, and more discussion about the commitment scheme is given in Appendix C.

2.3 Technical definitions and general lemmas

Singular Values. For a rank-n matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, there exists orthogonal matrices $\boldsymbol{U},\boldsymbol{V}$ and a diagonal matrix $\boldsymbol{\Lambda}$ with the non-negative diagonal entries $\sigma_1 \ge \cdots \ge \sigma_n$ such that $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^\top$. The values $\sigma_1(\boldsymbol{A})$ and $\sigma_n(\boldsymbol{A})$ are called the largest and the least singular values of $\boldsymbol{A}$, respectively.

Fact 1 *For square matrices $\boldsymbol{A},\boldsymbol{A}_1,\ldots,\boldsymbol{A}_s \in \mathbb{R}^{n\times n}$, $s \ge 1$, and $c \in \mathbb{R}$, the following holds*

- $\sigma_1(\boldsymbol{A}_1\cdots\boldsymbol{A}_s) \le \sigma_1(\boldsymbol{A}_1)\cdots\sigma_1(\boldsymbol{A}_s)$, *and* $\sigma_n(\boldsymbol{A}_1\cdots\boldsymbol{A}_s) \ge \sigma_n(\boldsymbol{A}_1)\cdots\sigma_n(\boldsymbol{A}_s)$,
- $\sigma_1(c\boldsymbol{A}) = |c|\cdot\sigma_1(\boldsymbol{A})$ *and* $\sigma_n(c\boldsymbol{A}) = |c|\cdot\sigma_n(\boldsymbol{A})$,
- $\sigma_1(\boldsymbol{A}\otimes\boldsymbol{I}_m) = \sigma_1(\boldsymbol{A})$ *and* $\sigma_n(\boldsymbol{A}\otimes\boldsymbol{I}_m) = \sigma_n(\boldsymbol{A})$ *for any $m \ge 1$ where $\otimes$ denotes the Kronecker product,*
- $\boldsymbol{A}$ *and* $\boldsymbol{A}^\top$ *have the same singular values.*

Discrete Gaussian Distribution. In this work, we always consider Gaussian distributions centered at zero, and thus we restrict our definitions to that case. Let $\boldsymbol{S} \in \mathbb{R}^{m\times n}$ be a rank-n matrix. Define the *ellipsoid Gaussian function* on $\mathbb{R}^n$ centered at zero with parameter $\boldsymbol{S}$ (and covariance matrix $\boldsymbol{S}^\top\boldsymbol{S}$) as $\rho_{\boldsymbol{S}}(\boldsymbol{x}) = e^{-\pi\boldsymbol{x}^\top(\boldsymbol{S}^\top\boldsymbol{S})^{-1}\boldsymbol{x}}$ for all $\boldsymbol{x} \in \mathbb{R}^n$. The *ellipsoid discrete Gaussian distribution* over $\mathbb{Z}^n$ centered at zero with parameter $\boldsymbol{S}$ is then defined by the probability mass function $\mathcal{D}^n_{\boldsymbol{S}}(\boldsymbol{x}) = \rho_{\boldsymbol{S}}(\boldsymbol{x})/\rho_{\boldsymbol{S}}(\mathbb{Z}^n)$ where $\rho_{\boldsymbol{S}}(\mathbb{Z}^n) = \sum_{\boldsymbol{z}\in\mathbb{Z}^n}\rho_{\boldsymbol{S}}(\boldsymbol{z})$ is a normalisation factor. If the parameter $\boldsymbol{S} = s\boldsymbol{I}_n$ for $s \in \mathbb{R}^+$, then we obtain the *spherical* discrete Gaussian distribution, denoted by $\mathcal{D}^n_s$. We denote by D^n_σ the discrete normal distribution with standard deviation σ, defined as $\mathcal{D}^n_s$ with $s = \sigma\sqrt{2\pi}$.

Fact 2 (A result of [1, Fact 2]) *For an invertible $n \times n$ matrix $\boldsymbol{X}$, $\boldsymbol{X} \cdot \mathcal{D}^n_{\boldsymbol{S}} = \mathcal{D}^n_{\boldsymbol{S}\boldsymbol{X}^\top}$. That is, the distribution induced by sampling $\boldsymbol{v} \leftarrow \mathcal{D}^n_{\boldsymbol{S}}$ and outputting $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{v}$ is the same as $\mathcal{D}^n_{\boldsymbol{S}\boldsymbol{X}^\top}$.*

As defined in [25], for a lattice L and real $\epsilon > 0$, the *smoothing parameter*, $\eta_\epsilon(L)$, of L is the smallest s such that $\rho_{1/s}(L^* \setminus \{\boldsymbol{0}\}) \leq \epsilon$ where L^* is the "dual lattice". We skip the details, but for our purposes the following facts are enough.

Fact 3 ([25, Lemma 3.3]) *$\eta_\epsilon(\mathbb{Z}^n) < 6$ for $\epsilon = 2^{-128}$ and any $1 \leq n \leq 2^{32}$.*

Lemma 2 ([1, Lemma 3]). *Let $\sigma_1(\boldsymbol{S})$ and $\sigma_n(\boldsymbol{S})$ be the largest and the least singular values of a rank-n matrix $\boldsymbol{S}$, respectively. If $\sigma_n(\boldsymbol{S}) \geq \eta_\epsilon(\mathbb{Z}^n)$,*

$$\Pr_{\boldsymbol{v} \leftarrow \mathcal{D}^n_{\boldsymbol{S}}}\left[\, \|\boldsymbol{v}\| \geq \sigma_1(\boldsymbol{S})\sqrt{n} \,\right] \leq \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n}.$$

We also summarise some known results related to the norms in the next lemma, which is followed by another lemma used for concrete parameter setting.

Lemma 3. *For $a, b \in R_q = \mathbb{Z}_q[X]/(X^d + 1)$, we have the following relations*

$$\|a\| \leq \sqrt{d} \cdot \|a\|_\infty, \quad \|a \cdot b\| \leq \sqrt{d} \cdot \|a\| \cdot \|b\|, \text{ and } \quad \|a \cdot b\|_\infty \leq \|a\| \cdot \|b\|.$$

Lemma 4 ([22, Lemma 4.4]).

1. *For any $\alpha > 0$, $\Pr[|z| > \alpha \cdot \sigma \ : \ z \leftarrow D_\sigma] \leq 2 \cdot \exp(-\frac{\alpha^2}{2})$,*
2. *For any $\alpha > 1$, $\Pr[\|\boldsymbol{z}\| > \alpha\sigma\sqrt{t} \ : \ \boldsymbol{z} \leftarrow D^t_\sigma] < \alpha^t e^{\frac{1-\alpha^2}{2}t}$. In particular,*

- $\Pr[|z| > 12\sigma \ : \ z \leftarrow D_\sigma] < 2^{-100}$,
- $\Pr[\|\boldsymbol{z}\| > 2\sigma\sqrt{t} \ : \ \boldsymbol{z} \leftarrow D^t_\sigma] < 2^{-100}$ *if* $t \geq 86$, *and*
- $\Pr[\|\boldsymbol{z}\| > 5\sigma\sqrt{t} \ : \ \boldsymbol{z} \leftarrow D^t_\sigma] < 2^{-100}$ *if* $t \geq 7$.

The lemma below recalls the norm bound of monomial challenge differences.

Lemma 5 ([6, Lemma 3.1]). *For $0 \leq i, j \leq 2d-1$, all the coefficients of $2(X^i - X^j)^{-1} \in R$ are in $\{-1, 0, 1\}$. This implies that $\left\|2(X^i - X^j)^{-1}\right\| \leq \sqrt{d}$.*

Rejection sampling. Algorithm 1 summarizes the result of the rejection sampling technique from [22] with its result given in Lemma 6.

Lemma 6 ([22]). *Let $\boldsymbol{c} \leftarrow h$ and $\phi > 0$ for a probability distribution h over $V \subseteq \mathbb{Z}^s$ ($s \geq 1$) where all the elements have norm less than K. Consider a procedure $\mathcal{F}$ that samples $\boldsymbol{y} \leftarrow D^s_\sigma$ and outputs Rej($\boldsymbol{z}, \boldsymbol{c}, \phi, K$) (Algorithm 1) for $\boldsymbol{z} = \boldsymbol{y} + \boldsymbol{c}$. Then, for $\mu(\phi) = e^{12/\phi + 1/(2\phi^2)}$, the probability of $\mathcal{F}$ outputting 1 is within 2^{-100} of $1/\mu(\phi)$, and conditioned on the output being 1, $\boldsymbol{z}$ is distributed within the statistical distance at most 2^{-100} of D^s_σ.*

Representative matrices. For a vector $\boldsymbol{p} = (p_0, \ldots, p_{m-1})$ of polynomials in R with $m \geq 1$, we denote the vector of all coefficients in $\boldsymbol{p}$ by $\mathrm{Coeff}(\boldsymbol{p}) \in \mathbb{Z}^{md}$. For any $f, g \in R$, there exists a matrix $\mathrm{Rot}(f)$, called the *Rot matrix*, of f such that $\mathrm{Rot}(f) \cdot \mathrm{Coeff}(g) = \mathrm{Coeff}(f \cdot g)$. This notion generalizes to the case where $(\mathrm{Rot}(f) \otimes \boldsymbol{I}_m) \cdot \mathrm{Coeff}(\boldsymbol{p}) = \mathrm{Coeff}(f \cdot \boldsymbol{p})$ for $f \in R$ and $\boldsymbol{p} \in R^m$ for $m \geq 1$ where $\otimes$ denotes the Kronecker product.

Algorithm 1 $\mathrm{Rej}(\boldsymbol{z}, \boldsymbol{c}, \phi, K)$

1: $\sigma = \phi K$; $\mu(\phi) = e^{12/\phi + 1/(2\phi^2)}$; $u \leftarrow [0, 1)$
2: **if** $u > (\frac{1}{\mu(\phi)}) \cdot \exp\left(\frac{-2\langle \boldsymbol{z}, \boldsymbol{c}\rangle + \|\mathbf{c}\|^2}{2\sigma^2}\right)$ **then return** 0 ▷ indicates 'abort protocol'.
3: **else return** 1

2.4 Σ-protocols

Σ-protocols are a type of zero-knowledge proofs between two parties: the prover and the verifier. A language $\mathcal{L} \subseteq \{0,1\}^*$ is said to have a witness relationship $\mathcal{R} \subseteq \{0,1\}^* \times \{0,1\}^*$ provided $v \in \mathcal{L}$ if and only if there exists $w \in \{0,1\}^*$ such that $(v, w) \in \mathcal{R}$. The quantity w is referred to as a witness for v. The definition of Σ-protocols from [6] generalises the well-known notion of Σ-protocols. We further extend it to allow $(k+1)$-special soundness as in [15, 8].

Definition 3 (Extension of Definition 2.5 in [6]). *Let $(\mathcal{P}, \mathcal{V})$ be a two-party protocol where $\mathcal{V}$ is a PPT algorithm, and $\mathcal{L}, \mathcal{L}'$ be languages with witness relations $\mathcal{R}, \mathcal{R}'$ with $\mathcal{R} \subseteq \mathcal{R}'$. Then, $(\mathcal{P}, \mathcal{V})$ is called a Σ-protocol for $\mathcal{R}, \mathcal{R}'$ with completeness error α, a challenge set $\mathcal{C}$, public input v and private input w, if it satisfies the following conditions:*

- ***Three-move form:*** *The protocol has the following form. On input (v, w), $\mathcal{P}$ computes initial commitment t and sends it to $\mathcal{V}$. On input v, $\mathcal{V}$ draws a challenge $x \leftarrow \mathcal{C}$ and sends it to $\mathcal{P}$. The prover sends a response s to $\mathcal{V}$. The verifier accepts or rejects depending on the protocol transcript (t, x, s). The transcript (t, x, s) is called accepting if the verifier accepts the protocol run.*
- ***Completeness:*** *Whenever $(v, w) \in \mathcal{R}$, the honest verifier accepts with probability at least $1 - \alpha$ when interacting with an honest prover.*
- $(k+1)$***-special soundness:*** *There exists a PPT algorithm $\mathcal{E}$ (called the extractor) which takes $(k+1)$ accepting transcripts $(t, x_0, s_0), \ldots, (t, x_k, s_k)$ with pairwise distinct x_i's $(0 \leq i \leq k)$ as inputs, and outputs w' satisfying $(v, w') \in \mathcal{R}'$. We call this procedure* witness extraction*, and say that the protocol has a soundness error $\frac{k}{|C|}$.*[7]
- ***Special honest-verifier zero-knowledge (SHVZK):*** *There exists a PPT algorithm $\mathcal{S}$ (called the simulator) that takes $v \in \mathcal{L}$ and $x \in \mathcal{C}$ as inputs, and outputs (t, s) such that (t, x, s) is indistinguishable from an accepting protocol transcript generated by a real protocol run.*

3 New Technical Tools for Lattice-based Proofs

In this section, we present a collection of technical tools we use in our constructions. These new tools may be of independent interest for future works on algebraic lattice-based zero-knowledge proofs and signatures.

[7] We refer to Section 2.2 of [7] for further discussion on soundness error.

3.1 Proving a value binary in R_q

We show a lemma (see Appendix E for the proof) that, in particular, enables us to guarantee that $b \in R_q$ is a bit when the equation $b \cdot (1 - b) = 0$ holds over R_q. Our lemma does not put any additional assumption on q but its size, which enables one to use fast computation algorithms such as the number-theoretic transform (NTT) with $q \equiv 1 \bmod 2d$. In particular, we do not need number theoretic conditions on q that makes NTT less efficient. For example, such a condition is imposed in [13] to ensure the invertibility of small elements in R_q.

Lemma 7. *For $b \in R_q$, if $b \cdot (\alpha - b) = 0$ over R_q for some positive integer α, and $\|b\| + \alpha < \sqrt{q}$, then $b \in \{0, \alpha\}$.*

3.2 Bounding the extracted witness norm for monomial challenges

Consider a Σ-protocol where the prover's initial commitments are $A_0, A_1, \ldots, A_k$ ($k \geq 1$), and he responds with $(\boldsymbol{f}_x, \boldsymbol{r}_x)$ for a given challenge x by the verifier. Then, the verifier checks whether $A_0 + A_1 x + A_2 x^2 + \cdots + A_k x^k = \mathrm{Com}(\boldsymbol{f}_x; \boldsymbol{r}_x)$ holds where Com is a homomorphic commitment scheme. Now, suppose A_k is the commitment of prover's witness and that the extractor obtains $k + 1$ accepting protocol transcripts for the same initial commitments, represented as follows.

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^k \\ 1 & x_1 & x_1^2 & \cdots & x_1^k \\ : & : & : & : & : \\ 1 & x_k & x_k^2 & \cdots & x_k^k \end{pmatrix} \cdot \begin{pmatrix} A_0 \\ A_1 \\ : \\ A_k \end{pmatrix} = \begin{pmatrix} \mathrm{Com}(\boldsymbol{f}_{x_0}; \boldsymbol{r}_{x_0}) \\ \mathrm{Com}(\boldsymbol{f}_{x_1}; \boldsymbol{r}_{x_1}) \\ : \\ \mathrm{Com}(\boldsymbol{f}_{x_k}; \boldsymbol{r}_{x_k}) \end{pmatrix}.$$

Here, the matrix on the very left is a Vandermonde matrix $\boldsymbol{V}$, and the extractor can recover a *possible* opening of A_k via multiplying both sides by $\boldsymbol{V}^{-1}$, if exists, due to the homomorphic properties of the commitment scheme. We observe from [33] that the inverse matrix $\boldsymbol{V}^{-1}$ has the following form:

$$\begin{pmatrix} \frac{*}{(x_0-x_1)(x_0-x_2)\cdots(x_0-x_k)} & \frac{*}{(x_0-x_1)(x_1-x_2)\cdots(x_1-x_k)} & \cdots & \frac{*}{(x_0-x_k)(x_1-x_k)\cdots(x_{k-1}-x_k)} \\ \frac{*}{(x_0-x_1)(x_0-x_2)\cdots(x_0-x_k)} & \frac{*}{(x_0-x_1)(x_1-x_2)\cdots(x_1-x_k)} & \cdots & \frac{*}{(x_0-x_k)(x_1-x_k)\cdots(x_{k-1}-x_k)} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{1}{(x_0-x_1)(x_0-x_2)\cdots(x_0-x_k)} & \frac{-1}{(x_0-x_1)(x_1-x_2)\cdots(x_1-x_k)} & \cdots & \frac{(-1)^k}{(x_0-x_k)(x_1-x_k)\cdots(x_{k-1}-x_k)} \end{pmatrix}, \quad (1)$$

where $*$ denotes some element in the domain. Our protocol as well as the protocols in [15, 8] have this structure and, therefore, the Vandermonde matrix inverse plays a crucial role in the witness extraction. In particular, if we denote the entries in the last row of $\boldsymbol{V}^{-1}$ by $\alpha_0, \ldots, \alpha_k$ (from left to right), we have

$$A_k = \sum_{j=0}^{k} \alpha_j \mathrm{Com}(\boldsymbol{f}_{x_j}; \boldsymbol{r}_{x_j}) = \mathrm{Com}\left(\sum_{j=0}^{k} \alpha_j \boldsymbol{f}_{x_j}; \sum_{j=0}^{k} \alpha_j \boldsymbol{r}_{x_j} \right) =: \mathrm{Com}(\boldsymbol{m}_{\mathrm{ext}}; \boldsymbol{r}_{\mathrm{ext}}). \quad (2)$$

These arguments tell us that we need to make sure $\boldsymbol{V}^{-1}$ exists in the first place, which follows from the invertibility of pairwise differences of challenges. What is

more important in the case of lattice-based proofs is that α_j's (and, in general, the entries in $\boldsymbol{V}^{-1}$) must have small norm so that extracted witness (particularly, $(\boldsymbol{m}_{\text{ext}}; \boldsymbol{r}_{\text{ext}})$) is a *valid* opening (of A_k). To that end, we can make use of Lemma 5 to bound the entries in $\boldsymbol{V}^{-1}$, which brings us to our first method below. In the rest, we focus on the last row of $\boldsymbol{V}^{-1}$, which is enough for our purposes, but our results can be extended to the cases related to the other entries of $\boldsymbol{V}^{-1}$.

Method 1. Taking the first entry α_0 as an example, we have

$$2^k\alpha_0 = \frac{2^k}{(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_k)} = \frac{2}{x_0 - x_1} \cdot \frac{2}{x_0 - x_2} \cdots \frac{2}{x_0 - x_k}.$$

For monomial challenges, using Lemma 3 and Lemma 5, we get

$$\left\|2^k\alpha_0\right\| = \left\|\prod_{i=1}^{k} \frac{2}{x_0 - x_i}\right\| \leq \left(\sqrt{d}\right)^{k-1} \prod_{i=1}^{k} \|2(x_0 - x_i)^{-1}\| \leq \left(\sqrt{d}\right)^{k-1} \left(\sqrt{d}\right)^{k} = d^{k-0.5}.$$

Since all the entries in the last row have a similar form and the bound does not depend on the particular choice of monomials, the same bound holds for all entries in the last row of $\boldsymbol{V}^{-1}$. Note that $\boldsymbol{V}^{-1}$ exists over R_q for odd q (though may not have small entries) since 2 is invertible for such q. We summarise these results in the following lemma, whose proof follows from the above discussion.

Lemma 8. *For $k \in \mathbb{Z}^+$, let $x_i = X^{\omega_i} \in R = \mathbb{Z}[X]/(X^d + 1)$ for $0 \leq \omega_i \leq 2d - 1$ and $0 \leq i \leq k$. Define the Vandermonde matrix $\boldsymbol{V}$ of dimension $k+1$ where i-th row is the vector $(1, x_i, x_i^2, \ldots, x_i^k)$. Then, $\boldsymbol{V}$ is invertible over R_q for odd q, and for any entry α_j $(0 \leq j \leq k)$ in the last row of $\boldsymbol{V}^{-1}$, we have $\left\|2^k\alpha_j\right\| \leq d^{k-0.5}$.*

Using Lemma 8, we can now summarize the main result of Method 1.

Lemma 9. *For the extracted opening $(\boldsymbol{m}_{\text{ext}}, \boldsymbol{r}_{\text{ext}})$ of A_k in (2), we have*

$$\left\|2^k\boldsymbol{r}_{\text{ext}}\right\| \leq (k+1) \cdot d^k \cdot \max_{0 \leq j \leq k} \left\|\boldsymbol{r}_{x_j}\right\| \quad \textit{and} \quad \left\|2^k\boldsymbol{m}_{\text{ext}}\right\| \leq (k+1) \cdot d^k \cdot \max_{0 \leq j \leq k} \left\|\boldsymbol{f}_{x_j}\right\|.$$

The proof is deferred to Appendix E. This initial attempt succeeds, but the result may not be optimal. Thus, we deepen our analysis to get a tighter bound.

Method 2. We observe that all entries in $\boldsymbol{V}^{-1}$ are constructed by challenge values, which are public. Therefore, independent of a protocol run, anyone can take a set of challenges and compute, in particular, $\left\|2^k\alpha_j\right\|$ for any entry α_j in the last row of $\boldsymbol{V}^{-1}$. The important part here is that one can indeed iterate through all the possible challenge sets (to be used in witness extraction) *if* the challenge space size and k are not too large. This means anyone can compute a global bound $\mathbb{B}_{d,k}$ on $\left\|2^k\alpha_j\right\|$ for any given k and d independent of the index j and the challenges used in the witness extraction.

Observing from (1), the total search space will be of size at most $(k+1) \cdot |\mathcal{C}|^{k+1}$ where $|\mathcal{C}| = 2d$ denotes the monomial challenge space size. However, note that, assuming w.l.o.g. $i > j$,

$$\left\|(X^i - X^j)^{-1}\right\| = \left\|X^{2d-i}(1 - X^{j-i})^{-1}\right\| = \left\|(1 - X^{j-i})^{-1}\right\| \tag{3}$$

since multiplication by a monomial in R simply performs a nega-cyclic rotation of the coefficients. Therefore, for any given k, it is enough to iterate through all subsets of $\{1, \ldots, 2d-1\}$ of size k, and compute $\left\|\prod_{\omega \in U_k} 2(1-X^\omega)^{-1}\right\|$ for such a given subset U_k. As a result, the search space size is reduced to $\binom{|\mathcal{C}|-1}{k}$. In our parameter setting for practical ring sizes of $N \leq 2^{20}$, we have $k \leq 3$. Therefore, for example, for $d = 64$ and $k = 3$, this requires only $\binom{127}{3} < 2^{18.4}$ iterations to be performed only ever once. Below is the result of Method 2 where the proof follows by replacing $\max_{0 \leq j \leq k} \left\|2^k \alpha_j\right\|$ in (13) in the proof of Lemma 9 by $\mathbb{B}_{d,k}$.

Lemma 10. *For the extracted opening $(\boldsymbol{m}_{\text{ext}}, \boldsymbol{r}_{\text{ext}})$ of A_k in (2), and any given d and k, there exists a constant $\mathbb{B}_{d,k} \leq d^{k-0.5}$ and an algorithm to compute $\mathbb{B}_{d,k}$ with a running time at most $(k-1) \cdot \binom{2d-1}{k}$ polynomial multiplications in R_q and $\binom{2d-1}{k}$ Euclidean norm computations of degree d polynomials such that*

$$\left\|2^k \boldsymbol{r}_{\text{ext}}\right\| \leq (k+1) \cdot \sqrt{d} \cdot \mathbb{B}_{d,k} \cdot \max_{0 \leq j \leq k} \left\|\boldsymbol{r}_{x_j}\right\|, \text{ and} \tag{4}$$

$$\left\|2^k \boldsymbol{m}_{\text{ext}}\right\| \leq (k+1) \cdot \sqrt{d} \cdot \mathbb{B}_{d,k} \cdot \max_{0 \leq j \leq k} \left\|\boldsymbol{f}_{x_j}\right\|. \tag{5}$$

Method 3. The above two methods give us ways to bound the extracted witness length independent of a protocol run. The question now is "How much additional information can we use from a protocol run?"

Assume that the prover's response follows a discrete Gaussian distribution, i.e., $\boldsymbol{r}_x \leftarrow \mathcal{D}_s^{md}$ for some $s \in \mathbb{R}^+, m \in \mathbb{Z}^+$. Instead of bounding $\left\|2^k \alpha_j\right\|$, we bound $\left\|2^k \alpha_j \boldsymbol{r}_{x_j}\right\|$ for all j's. The product $2^k \alpha_j \boldsymbol{r}_{x_j}$ can be represented as $(\text{Rot}(2^k \alpha_j) \otimes \boldsymbol{I}_m) \cdot \text{Coeff}(\boldsymbol{r}_{x_j}) = \text{Coeff}(2^k \alpha_j \boldsymbol{r}_{x_j})$ where $\otimes$ denotes the Kronecker product. Let us denote $\boldsymbol{R}_j = \text{Rot}(2^k \alpha_j) \otimes \boldsymbol{I}_m$. Since $\text{Coeff}(\boldsymbol{r}_{x_j}) \leftarrow \mathcal{D}_s^{md}$, by Fact 2, we have $\boldsymbol{R}_j \cdot \text{Coeff}(\boldsymbol{r}_{x_j}) \in \mathcal{D}_{s\boldsymbol{R}_j^\top}^{md}$. Hence, by Lemma 2, with high probability, we get

$$\left\|\text{Coeff}(2^k \alpha_j \boldsymbol{r}_{x_j})\right\| = \left\|\boldsymbol{R}_j \cdot \text{Coeff}(\boldsymbol{r}_{x_j})\right\| \leq \sigma_1(s\boldsymbol{R}_j^\top)\sqrt{md} = \sigma_1(\boldsymbol{R}_j) s\sqrt{md}, \tag{6}$$

if $\sigma_n(s\boldsymbol{R}_j^\top) \geq \eta_\epsilon(\mathbb{Z}^{md})$, which can be easily satisfied as shown in the proof of Lemma 11. We can now summarize the main result of Method 3 as below, where the proof is given in Appendix E.

Lemma 11. *Let $\boldsymbol{r}_{\text{ext}} = \sum_{j=0}^{k} \alpha_j \boldsymbol{r}_{x_j}$ be the randomness opening of A_k as in (2). Assume that $s \geq 6$, $d \in \{4, 8, \ldots, 512\}$ and $md \leq 2^{32}$. If $\boldsymbol{r}_{x_j} \leftarrow \mathcal{D}_s^{md}$ for all $0 \leq j \leq k$, then with probability at least $1 - \frac{1+\epsilon}{1-\epsilon} 2^{-md}$ for $\epsilon = 2^{-128}$,*

$$\left\|2^k \boldsymbol{r}_{\text{ext}}\right\| \leq (k+1) \cdot \max_{0 \leq j \leq k} \sigma_1(\boldsymbol{S}_j) \cdot s\sqrt{md}, \tag{7}$$

where $\boldsymbol{S}_j = \text{Rot}(2^k \alpha_j)$ for $j = 0, \ldots, k$.

Similar to the idea in Method 2, one can iterate through all $\boldsymbol{S}_j$'s and compute a global bound $\mathbb{S}_{d,k}$ on possible $\sigma_1(\boldsymbol{S}_j)$'s for a given d and k. When $\boldsymbol{r}_{x_j} \leftarrow \mathcal{D}_s^{md}$, we have $\left\|\boldsymbol{r}_{x_j}\right\| \leq s\sqrt{md}$ (up to a small constant factor) by Lemma 2. As a result,

Table 2: Comparison of Method 1, Method 2 and Method 3. * indicates that only a subset of the whole search space has been iterated through.

	$k=2$			$k=3$			$k=4$		
d	$\log(d^k)$	$\log(\mathbb{B}'_{d,k})$	$\log(\mathbb{S}_{d,k})$	$\log(d^k)$	$\log(\mathbb{B}'_{d,k})$	$\log(\mathbb{S}_{d,k})$	$\log(d^k)$	$\log(\mathbb{B}'_{d,k})$	$\log(\mathbb{S}_{d,k})$
16	8	7.21	6.70	12	9.56	9.06	16	11.92	11.42
32	10	9.21	8.70	15	12.55	12.05	20	15.90	15.40
64	12	11.21	10.70	18	15.55	15.05	24	19.90	19.40
128	14	13.21	12.70	21	18.55	18.05^*	28	23.90^*	23.40^*
256	16	15.21	14.70	24	21.55	21.05^*	32	-	-

we may reduce the comparison of the three methods to the comparison of the values d^k (Method 1), $\mathbb{B}'_{d,k} = \sqrt{d} \cdot \mathbb{B}_{d,k}$ (Method 2) and $\mathbb{S}_{d,k}$ (Method 3).

However, there is an important detail in Method 3: it only works when the prover's response follows a discrete Gaussian distribution and the verifier cannot simply check if that is the case. To solve this problem, we introduce a new tool called, Pseudo Witness Extraction in Algorithm 2. If Algorithm 2 is used in protocol's verification with an input bound β, then $\left\|2^k \boldsymbol{r}_{\text{ext}}\right\| \leq (k+1)\beta$ must hold. Hence, when the prover's responses $\boldsymbol{r}_{x_j}$'s are from $\mathcal{D}_s^{md}$, setting $\beta = \mathbb{S}_{d,k} s\sqrt{md}$ ensures both that an honest prover's proof will be accepted and also that the extracted randomness will satisfy the norm-bound as in Lemma 11.

Algorithm 2 Pseudo-witness-extraction

1: **Input:** a vector $\boldsymbol{r}$; a challenge $x_0 \in \mathcal{C}$; an integer $k \geq 1$; a norm bound $\beta \in \mathbb{R}^+$
2: **for** each k-tuple $(x_1, \ldots, x_k) \in \mathcal{C}^k$ s.t. $x_0 \neq x_1 \neq \cdots \neq x_k$ **do**
3: $\quad \boldsymbol{r}_{\text{p-ext}} = \left[\prod_{j=1}^{k} 2(x_0 - x_j)^{-1}\right] \cdot \boldsymbol{r}$
4: $\quad$ **if** $\|\boldsymbol{r}_{\text{p-ext}}\| > \beta$ **then return** False
5: **end for**
6: **return** True

In Table 2, we provide a comparison between the three methods introduced. As can be seen from the table, as k increases, the advantage of Method 2 and Method 3 over Method 1 grows larger. There are also obvious patterns that can be observed from the table such as $\mathbb{S}_{d,k}/\mathbb{B}'_{d,k} \approx \sqrt{2}$ for any d and k. We leave the investigation of these behaviours as an open problem. For larger values of k, for which it is infeasible to search the whole space, one can use Lemma 3 to upper-bound $\mathbb{B}_{d,k}$ (as $\mathbb{B}_{d,k}$ is an upperbound on the norm of a product of polynomials) and Fact 1 to upper-bound $\mathbb{S}_{d,k}$ (as $\mathbb{S}_{d,k}$ is an upperbound on the singular value of a product of matrices). These still give better results over Method 1.

4 Σ-protocol for Commitment to a Sequence of Bits

In this section, we describe a lattice-based Σ-protocol showing that a commitment B opens to sequences of binary values where the Hamming weight of each sequence is exactly one. Let $N = \beta^k > 1$ and $\boldsymbol{r}, \hat{\boldsymbol{r}} \in R_q^m$, and define the relations to be proved in Definition 4.

Definition 4. *For positive real numbers $\mathcal{T}$ and $\hat{\mathcal{T}}$, we define the following relations to be used in Protocol 1.*

$$\mathcal{R}_{\text{bin}}(\mathcal{T}) = \left\{ \begin{array}{l} ((ck, B), (b_{0,0}, \ldots, b_{k-1,\beta-1}, \boldsymbol{r})) \ : \ \|\boldsymbol{r}\| \leq \mathcal{T} \ \wedge \ (b_{j,i} \in \{0,1\} \ \forall j, i) \\ \quad \wedge \ B = \text{Com}_{ck}(b_{0,0}, \ldots, b_{k-1,\beta-1} \ ; \ \boldsymbol{r}) \ \wedge \ (\sum_{i=0}^{\beta-1} b_{j,i} = 1 \ \forall j) \end{array} \right\}.$$

$$\mathcal{R}'_{\text{bin}}(\hat{\mathcal{T}}) = \left\{ \begin{array}{l} ((ck, B), (b_{0,0}, \ldots, b_{k-1,\beta-1}, \hat{\boldsymbol{r}})) \ : \ \|\hat{\boldsymbol{r}}\| \leq \hat{\mathcal{T}} \ \wedge \ (b_{j,i} \in \{0,1\} \ \forall j, i) \\ \quad \wedge \ 2B = \text{Com}_{ck}(2b_{0,0}, \ldots, 2b_{k-1,\beta-1} \ ; \ \hat{\boldsymbol{r}}) \ \wedge \ (\sum_{i=0}^{\beta-1} b_{j,i} = 1 \ \forall j) \end{array} \right\}.$$

Remark 1. The conditions on the norms of $\boldsymbol{r}$ and $\hat{\boldsymbol{r}}$ in the relations $\mathcal{R}_{\text{bin}}$ and $\mathcal{R}'_{\text{bin}}$ play a very crucial role, and is one of the main differences of a lattice-based zero-knowledge proof over its number-theoretic counterpart. Without that control, one cannot easily tie the security of the protocol to a hard lattice problem.

In the protocol, we first prove that each value in the sequences is binary, and then that the sum of each sequence equals one. This guarantees that there is only a single 1 in each sequence. The idea behind proving a value binary works as follows. Let b be the value we want to prove binary. Given a challenge x, the value b is multiplied by x and the resulting value is masked by a as $f = x \cdot b + a$ in the protocol (Step 10 in Protocol 1). Now observe that $f \cdot (x - f) = b(1-b) \cdot x^2 + a(1-2b) \cdot x - a^2$ and proving that the coefficient of x^2 is zero implies that $b(1-b) = 0$. Then, using Lemma 7, for a sufficiently large q, this statement over R_q implies that b is binary.

Similar to [6], we make use of an auxiliary commitment scheme aCom (which is assumed to be hiding and binding) in order to be able to simulate aborts in the proof of zero-knowledge property.[8] One can treat aCom as a random oracle. However, if aCom is computationally binding, then the soundness of the protocol holds under the respective assumption and similarly if it is computationally hiding [6]. The protocol is described in Protocol 1, which will later be used in the one-out-of-many proof. The parameters ϕ_1, ϕ_2 control the acceptance rate of two-step rejection sampling and can be adjusted as desired. The following summarizes the result of Protocol 1, and its proof is given in Appendix E.

Theorem 1. *For $T = (2d+2)\left(5^4\phi_1^4 d^3 k^3 \beta(\beta-1) + 12\phi_2^2 \mathcal{B}^2 m^2 d^2\right)^{1/2}$, assume that the commitment scheme is T-binding and also hiding (i.e., M-LWE$_{m-n,n,q,\mathcal{B}}$ is hard). Let $d \geq 7$, $md \geq 86$, and $q > (10\phi_1 d\sqrt{kd(\beta-1)} + 2)^2$. Then, Protocol 1 is a 3-special sound Σ-protocol (as in Definition 3) for relations $\mathcal{R}_{\text{bin}}(\mathcal{B}\sqrt{md})$ and $\mathcal{R}'_{\text{bin}}(4\sqrt{2}\phi_2 \mathcal{B} md^2)$ with soundness error $1/d$ and a completeness error $1 - 1/(\mu(\phi_1)\mu(\phi_2))$.*

[8] In protocol's application to a ring signature (and for other applications in general), simulation of aborts is not needed as the protocol is made non-interactive.

$\mathcal{P}_{\text{bin}}(ck, B, (\{b_{j,i}\}_{j,i=0}^{k-1,\beta-1}; \boldsymbol{r}))$		$\mathcal{V}_{\text{bin}}(ck, B)$
$a_{0,1}, \ldots, a_{k-1,\beta-1} \leftarrow D^d_{\phi_1\sqrt{k}}$		
$\boldsymbol{r}_c \leftarrow \{-\mathcal{B}, \ldots, \mathcal{B}\}^{md}$		
$\boldsymbol{r}_a, \boldsymbol{r}_d \leftarrow D^{md}_{\phi_2\mathcal{B}\sqrt{2md}}$		
for $j = 0, \ldots, k-1$ **do**		
$\quad a_{j,0} = -\sum_{i=1}^{\beta-1} a_{j,i}$		
$A = \text{Com}_{ck}(a_{0,0}, \ldots, a_{k-1,\beta-1}\ ;\ \boldsymbol{r}_a)$		
$C = \text{Com}_{ck}(\{a_{j,i}(1-2b_{j,i})\}_{j,i=0}^{k-1,\beta-1}\ ;\ \boldsymbol{r}_c)$		
$D = \text{Com}_{ck}(-a^2_{0,0}, \ldots, -a^2_{k-1,\beta-1}\ ;\ \boldsymbol{r}_d)$		
$(c_a, d_a) = \text{aCom}(A, C, D)$		
	$\xrightarrow{\quad c_a \quad}$	
	$\xleftarrow{\quad x := X^\omega \quad}$	$\omega \leftarrow \{0, \ldots, 2d-1\}$
$f_{j,i} = x \cdot b_{j,i} + a_{j,i}\ \forall j, \forall i \neq 0$		
$\boldsymbol{f}_1 := (f_{0,1}, \ldots, f_{k-1,\beta-1}), \boldsymbol{b}_1 := (b_{0,1}, \ldots, b_{k-1,\beta-1})$		
$\text{Rej}(\boldsymbol{f}_1, x\boldsymbol{b}_1, \phi_1, \sqrt{k})$		
$\boldsymbol{z}_b = x \cdot \boldsymbol{r} + \boldsymbol{r}_a$		
$\boldsymbol{z}_c = x \cdot \boldsymbol{r}_c + \boldsymbol{r}_d$		
$\text{Rej}((\boldsymbol{z}_b, \boldsymbol{z}_c), x(\boldsymbol{r}, \boldsymbol{r}_c), \phi_2, \mathcal{B}\sqrt{2md})$		
Return $\perp$ if aborted.	$\xrightarrow{\quad f_{0,1}, \ldots, f_{k-1,\beta-1}, d_a, A, C, D, \boldsymbol{z}_b, \boldsymbol{z}_c \quad}$	
		for $j = 0, \ldots, k-1$ **do**
		$\quad f_{j,0} = x - \sum_{i=1}^{\beta-1} f_{j,i}$
		$(c_a, d_a) \stackrel{?}{=} \text{aCom}(A, C, D)$
		$\|f_{j,i}\| \stackrel{?}{\leq} 5\phi_1\sqrt{dk}\ \ \forall j, \forall i \neq 0$
		$\|f_{j,0}\| \stackrel{?}{\leq} 5\phi_1\sqrt{dk(\beta-1)}\ \ \forall j$
		$\|\boldsymbol{z}_b\|, \|\boldsymbol{z}_c\| \stackrel{?}{\leq} 2\sqrt{2}\phi_2\mathcal{B}md$
		$\boldsymbol{f} := (f_{0,0}, \ldots, f_{k-1,\beta-1})$
		$\boldsymbol{g} := \{f_{j,i}(x - f_{j,i})\}_{j,i=0}^{k-1,\beta-1}$
		$xB + A \stackrel{?}{=} \text{Com}_{ck}(\boldsymbol{f}\ ;\ \boldsymbol{z}_b)$
		$xC + D \stackrel{?}{=} \text{Com}_{ck}(\boldsymbol{g}\ ;\ \boldsymbol{z}_c)$

Protocol 1: Lattice-based Σ-protocol for $\mathcal{R}_{\text{bin}}$ and $\mathcal{R}'_{\text{bin}}$.

Remark 2. The way the rejection sampling is done in Protocol 1 allows us to sample $f_{j,i}$'s from a narrower distribution, and to make their norm smaller. This as a result weakens the condition on the size of q.

5 Lattice-based One-out-of-Many Protocol

We are now ready to describe our main protocol. Let $\delta_{j,i}$ denote the Kronecker's delta such that $\delta_{j,i} = 1$ if $j = i$, and $\delta_{j,i} = 0$ otherwise. The prover's goal in the protocol is to show that he knows the randomness within a commitment to zero among a list of N commitments. (Note that the commitments other than the prover's need not be commitments to zero, i.e., there is no need to assume that they are well-formed). Similar to the previous works [15, 8], we assume that the number of commitments satisfy $N = \beta^k$, which can be realised by using the same commitment multiple times until such an N is reached. Let c_ℓ be the prover's commitment for $0 \leq \ell \leq N-1$, and $L = \{c_0, \ldots, c_{N-1}\}$ be the list of all commitments. The main idea is to prove knowledge of the index ℓ such that $\sum_{i=0}^{N-1} \delta_{\ell,i} c_i$ is a commitment to zero. Note that $\delta_{\ell,i} = \prod_{j=0}^{k-1} \delta_{\ell_j,i_j}$ where $\ell = (\ell_0, \ldots, \ell_{k-1})$ and $i = (i_0, \ldots, i_{k-1})$ are representations in base β. The relations for the protocol are given in Definition 5.

Definition 5. *For positive real numbers $\mathcal{T}$ and $\hat{\mathcal{T}}$, we define the following relations to be used in Protocol 2.*

$$\mathcal{R}_{1/\mathrm{N}}(\mathcal{T}) = \left\{ \begin{array}{c} ((ck, (c_0, \ldots, c_{N-1})), (\ell, \boldsymbol{r})) \ : \ (c_i \in R_q^n \ \ \forall i \in [0, N-1]) \wedge \\ \ell \in \{0, \ldots, N-1\} \ \wedge \ \|\boldsymbol{r}\| \leq \mathcal{T} \ \wedge \ c_\ell = \mathrm{Com}_{ck}(\mathbf{0} \ ; \ \boldsymbol{r}) \end{array} \right\}.$$

$$\mathcal{R}'_{1/\mathrm{N}}(\hat{\mathcal{T}}) = \left\{ \begin{array}{c} ((ck, (c_0, \ldots, c_{N-1})), (\ell, \hat{\boldsymbol{r}})) \ : \ (c_i \in R_q^n \ \ \forall i \in [0, N-1]) \wedge \\ \ell \in \{0, \ldots, N-1\} \ \wedge \ \|\hat{\boldsymbol{r}}\| \leq \hat{\mathcal{T}} \ \wedge \ 2^k c_\ell = \mathrm{Com}_{ck}(\mathbf{0} \ ; \ \hat{\boldsymbol{r}}) \end{array} \right\}.$$

For each $0 \leq j \leq k-1$, the prover commits to a sequence $(\delta_{\ell_j,0}, \ldots, \delta_{\ell_j,\beta-1})$ and proves that it is a binary sequence with Hamming weight one using Protocol 1. As given in Protocol 1, the prover responds with $f_{j,i} = x \cdot \delta_{\ell_j,i} + a_{j,i}$ upon receiving a challenge x. Now, let us concentrate on the product $\prod_{j=0}^{k-1} f_{j,i_j} =: p_i(x)$. Observe that for all $i \in \{0, \ldots, N-1\}$,

$$p_i(x) = \prod_{j=0}^{k-1} \left(x\delta_{\ell_j,i_j} + a_{j,i_j} \right) = \prod_{j=0}^{k-1} x\delta_{\ell_j,i_j} + \sum_{j=0}^{k-1} p_{i,j} x^j = \delta_{\ell,i} x^k + \sum_{j=0}^{k-1} p_{i,j} x^j, \quad (8)$$

for some coefficients $p_{i,j}$'s depending on ℓ and $a_{j,i}$, which means that $p_{i,j}$'s can be computed by the prover before receiving a challenge. Now, since $\delta_{\ell,i} = 1$ if and only if $i = \ell$, the only p_i of degree k is p_ℓ. Then, the idea is to send some E_j's in the initial message, which will later be used by the verifier to cancel out the coefficients of low order terms $1, x, \ldots, x^{k-1}$, and the coefficient of x^k will be $\sum_{i=0}^{N-1} \delta_{\ell,i} c_i = c_\ell$, which corresponds to the prover's commitment. The full protocol is described in Protocol 2. We summarize the results of Protocol 2 below, and defer its proof to Appendix E.

$\mathcal{P}(ck, (c_0, \ldots, c_{N-1}), (\ell, \boldsymbol{r}))$	$\mathcal{V}(ck, (c_0, \ldots, c_{N-1}))$
1: $\boldsymbol{r}_b \leftarrow \{-\mathcal{B}, \ldots, \mathcal{B}\}^{md}$	
2: $\boldsymbol{\delta} = (\delta_{\ell_{0,0}}, \ldots, \delta_{\ell_{k-1,\beta-1}})$	
3: $B = \mathrm{Com}_{ck}(\boldsymbol{\delta}\ ;\ \boldsymbol{r}_b)$	
4: $A, C, D, \boldsymbol{r}_c \leftarrow \mathcal{P}_{\mathrm{bin}}(ck, B, (\boldsymbol{\delta}, \boldsymbol{r}_b))[1-8]$	
5: **for** $j = 0, \ldots, k-1$ **do**	
6: $\quad \boldsymbol{\rho}_j \leftarrow D^{md}_{\phi_2 \mathcal{B}\sqrt{3md/k}}$	
7: $\quad E_j = \sum_{i=0}^{N-1} p_{i,j} c_i + \mathrm{Com}_{ck}(\mathbf{0}\ ;\ \boldsymbol{\rho}_j)$	
using $p_{i,j}$'s from (8)	
8: $(c_a, d_a) = \mathrm{aCom}(A, B, C, D, \{E_j\})$	
$\xrightarrow{\quad c_a \quad}$	
$\xleftarrow{\quad x = X^{\omega} \quad}$	$\omega \leftarrow \{0, \ldots, 2d-1\}$
9: $\boldsymbol{f}_1, \boldsymbol{z}_b, \boldsymbol{z}_c \leftarrow \mathcal{P}_{\mathrm{bin}}(x)[10-13]$	
10: $\boldsymbol{z} = x^k \cdot \boldsymbol{r} - \sum_{j=0}^{k-1} x^j \cdot \boldsymbol{\rho}_j$	
11: $\mathrm{Rej}((\boldsymbol{z}, \boldsymbol{z}_b, \boldsymbol{z}_c), (x^k \boldsymbol{r}, x\boldsymbol{r}_b, x\boldsymbol{r}_c), \phi_2, \mathcal{B}\sqrt{3md})$	
Return $\perp$ if aborted.	$\xrightarrow{\quad d_a, \boldsymbol{f}_1, B, \boldsymbol{z}, \{E_j\}_{j=0}^{k-1}, \boldsymbol{R} := (A, C, D, \boldsymbol{z}_b, \boldsymbol{z}_c) \quad}$
	1: $\mathcal{V}_{\mathrm{bin}}(ck, B, x, \boldsymbol{f}_1, \boldsymbol{R})[1,2,6,7] \stackrel{?}{=} 1$
	2: $(c_a, d_a) \stackrel{?}{=} \mathrm{aCom}(A, B, C, D, \{E_j\})$
	3: $\lVert f_{j,i} \rVert \stackrel{?}{\leq} 5\phi_1\sqrt{dk}\ \ \forall j, \forall i \neq 0$
	4: $\lVert f_{j,0} \rVert \stackrel{?}{\leq} 5\phi_1\sqrt{dk(\beta-1)}\ \ \forall j$
	5: $\lVert \boldsymbol{z} \rVert, \lVert \boldsymbol{z}_b \rVert, \lVert \boldsymbol{z}_c \rVert \stackrel{?}{\leq} 2\sqrt{3}\phi_2 \mathcal{B} md$
	6: $\sum_{i=0}^{N-1} \left(\prod_{j=0}^{k-1} f_{j,i_j} \right) c_i - \sum_{j=0}^{k-1} E_j x^j \stackrel{?}{=} \mathrm{Com}_{ck}(\mathbf{0}\ ;\ \boldsymbol{z})$ for $i = (i_0, \ldots, i_{k-1})$.

Protocol 2: Lattice-based Σ-protocol for $\mathcal{R}_{1/\mathrm{N}}$ and $\mathcal{R}'_{1/\mathrm{N}}$.
$\mathcal{P}_{\mathrm{bin}}(ck, B, (\boldsymbol{\delta}, \boldsymbol{r}_b))[1-8]$ denotes running the same steps from 1 to 8 done by $\mathcal{P}_{\mathrm{bin}}$ in Protocol 1. Similar notation is used for $\mathcal{V}_{\mathrm{bin}}$. $\boldsymbol{r}_a$ and $\boldsymbol{r}_d$ in $\mathcal{P}_{\mathrm{bin}}(ck, B, (\boldsymbol{\delta}, \boldsymbol{r}_b))[1-8]$ are drawn from $D^{md}_{\phi_2 \mathcal{B}\sqrt{3md}}$ instead of $D^{md}_{\phi_2 \mathcal{B}\sqrt{2md}}$ as the rejection sampling is now done on a $(3md)$-dimensional vector.

Theorem 2. *For* $T = (2d+2)\left(5^4\phi_1^4 d^3 k^3 \beta(\beta-1) + 12\phi_2^2 \mathcal{B}^2 m^2 d^2\right)^{1/2}$*, assume that the commitment scheme is* T*-binding and also hiding (i.e.,* M-LWE$_{m-n,n,q,\mathcal{B}}$ *is hard). Let* $d \geq 7$*,* $md \geq 86$*, and* $q > (10\phi_1 d\sqrt{dk(\beta-1)}+2)^2$*. Then, Protocol 2 is a* $(k'+1)$*-special sound* Σ*-protocol (as in Definition 3) for the relations* $\mathcal{R}_{1/\mathrm{N}}(\mathcal{B}\sqrt{md})$ *and* $\mathcal{R}'_{1/\mathrm{N}}(2\sqrt{3}\phi_2\mathcal{B}md\cdot(k+1)\cdot d^k)$ *with a soundness error* $\frac{k'}{2d}$ *and a completeness error* $1-1/(\mu(\phi_1)\mu(\phi_2))$ *where* $k' = \max\{2,k\}$*.*

Proof (Theorem 2). Completeness and SHVZK proofs are given in Appendix E.

$(k+1)$-special soundness: Given $(k+1)$ distinct challenges $x_0,\ldots,x_k$, by the binding property of aCom, we have $(k+1)$ accepting responses with the same $(A,B,C,D,\{E_j\})$. Suppose that $((f_{j,i}^{(0)},\boldsymbol{z}^{(0)}),\ldots,(f_{j,i}^{(k)},\boldsymbol{z}^{(k)}))$ are produced and $k>1$. We first use 3-special soundness of Protocol 1 to extract openings $\hat{b}_{j,i}$ and $\hat{a}_{j,i}$ of $2B$ and $2A$, respectively. We can also obtain $b_{j,i}$ such that $\hat{b}_{j,i} = 2b_{j,i}$, and it is guaranteed that $b_{j,i} \in \{0,1\}$ and $\sum_{i=0}^{\beta-1} b_{j,i} = 1$. From here, we can obtain the digits ℓ_j by choosing $\ell_j = i^*$ for which $b_{j,i^*} = 1$. Then, we construct the index ℓ as $\ell = \sum_{j=0}^{k-1}\beta^j\ell_j$.

Using $b_{j,i}$ and $\hat{a}_{j,i}$, we can compute $\hat{p}_i(x) = 2^k\prod_{j=0}^{k-1} f_{j,i_j} = \prod_{j=0}^{k-1} 2f_{j,i_j} = \prod_{j=0}^{k-1}(x\cdot 2b_{j,i_j} + \hat{a}_{j,i_j})$. Note that $\hat{p}_\ell(x)$ is the only such polynomial of degree k in x by the construction of ℓ. Thus, the last verification step, when both sides are multiplied by 2^k, can be rewritten as $\sum_{i=0}^{N-1}\hat{p}_i(x)c_i - \sum_{j=0}^{k-1} 2^k E_j x^j = \mathrm{Com}_{ck}(\mathbf{0}\,;\,2^k\boldsymbol{z})$. Separating the term of degree k with respect to x, we get

$$x^k\cdot 2^k c_\ell + \sum_{j=0}^{k-1}\tilde{E}_j x^j = \mathrm{Com}_{ck}(\mathbf{0}\,;\,2^k\boldsymbol{z}), \tag{9}$$

where $\tilde{E}_j$'s are the coefficients of the monomials x^j of degree strictly less than k. Now, we know that (9) holds for distinct challenges $x_0,\ldots,x_k$, which can be represented as a system of equations where $x_0,\ldots,x_k$ form a Vandermonde matrix $\boldsymbol{V}$ as in Section 3.2. From the discussion in Section 3.2, $\boldsymbol{V}$ is invertible and we can obtain a linear combination $\alpha_0,\ldots,\alpha_k$ of copies of (9) with respect to different challenges that produces the vector $(0,\ldots,0,1)$. This gives

$$2^k c_\ell = \sum_{e=0}^{k}\alpha_e\left(x_e^k\cdot 2^k c_\ell + \sum_{j=0}^{k-1}\tilde{E}_j x_e^j\right) = \mathrm{Com}_{ck}(\mathbf{0}\,;\,2^k\sum_{e=0}^{k}\alpha_e\boldsymbol{z}^{(e)}). \tag{10}$$

An opening of $2^k c_\ell$ to the message $\mathbf{0}$ with randomness $\boldsymbol{r}_{\mathrm{ext}} = 2^k\sum_{e=0}^{k}\alpha_e\boldsymbol{z}^{(e)}$ is obtained. The bound on the norm of $\boldsymbol{r}_{\mathrm{ext}}$ for $\mathcal{R}'_{1/\mathrm{N}}$ follows easily by Lemma 9.

Finally, we assumed that $k>1$. If $k=1$, then we still need at least 3 challenges to be able to prove special soundness due to the 3-special soundness of Protocol 1. Thus, Protocol 2 is k'-special sound for $k' = \max\{2,k\}$, and since $|\mathcal{C}| = 2d$, the soundness error is $k'/2d$. □

It is easy to see from the definition of $\mathcal{R}'_{1/\mathrm{N}}$ that the norm of the extracted randomness, and thus the size of q, grows with $d^k = d^{\log_\beta N}$. If one is to rely on

Ring-SIS and use a base $\beta = 2$, then this growth would be very rapid, yielding a very inefficient scheme. This justifies our choice of working with M-SIS problem and choosing large base values β as given in Section 6.1. As discussed in Section 3.2, the bound on $\|\boldsymbol{r}_{\text{ext}}\|$ can be tightened using Method 2 or Method 3.

6 Lattice-based Ring Signature

Let $N = \beta^k$ for $2 \leq \beta \leq N$, and n, m be fixed positive integers. As a single run of Protocol 2 does not provide a small enough soundness error, suppose that r non-aborting executions of Protocol 2 gives negligible soundness error of $2^{-\lambda}$.

Recall that a single run of Protocol 2 produces an accepting transcript with probability $1/(\mu(\phi_1)\mu(\phi_2))$. Therefore, when it is repeated r times, the overall acceptance rate reduces to $1/(\mu(\phi_1)\mu(\phi_2))^r$, which is too small. Therefore, we introduce the tweaks below to Protocol 2 in order to get an overall completeness error of $1 - 1/(\mu(\phi_1)\mu(\phi_2))$ for the r-repeated protocol.

Tweaks for r-repeated protocol. First, we apply the rejection sampling to r-concatenated vectors at once. That is, it is applied on $(\boldsymbol{f}_1^1, \ldots, \boldsymbol{f}_1^r)$ and $(\boldsymbol{z}^1, \boldsymbol{z}_b^1, \boldsymbol{z}_c^1, \ldots, \boldsymbol{z}^r, \boldsymbol{z}_b^r, \boldsymbol{z}_c^r)$. Thus, we need to sample $f_{j,i} \leftarrow D^d_{12\sqrt{kr}}$ $(i \neq 0)$ and $\boldsymbol{z}, \boldsymbol{z}_b, \boldsymbol{z}_c \leftarrow D^{md}_{12\mathcal{B}\sqrt{3mdr}}$, and hence require $q > (10\phi_1 d\sqrt{dkr(\beta-1)} + 2)^2$ as in Assumption 1. Furthermore, since the extracted randomness norm will be larger, the relation $\mathcal{R}'_{1/\mathrm{N}}$ becomes $\mathcal{R}'_{1/\mathrm{N}}(24\sqrt{3r}\mathcal{B}md \cdot (k+1) \cdot d^k)$ and the commitment scheme is required to be binding in a larger domain. Therefore, the commitment scheme is set to be T_1-binding for $T_1 = (2d+2)\left(5^4\phi_1^4 d^3 k^3 \beta(\beta-1)r^2 + 12\phi_2^2\mathcal{B}^2 m^2 d^2 r\right)^{1/2}$.

Note that these tweaks do not affect the soundness error of individual protocol runs as the extraction still works with $k+1$ accepting transcripts. Only the extracted witness norm is increased since the bound on $\|\boldsymbol{z}\|$ changes from $24\sqrt{3}\mathcal{B}md$ to $24\sqrt{3r}\mathcal{B}md$ in Protocol 2.

Construction. We now describe our lattice-based ring signature, which similarly builds on the one-out-of-many proof as in [15, 8]. First, we summarise the assumptions on the parameters, and also let $CMT = (A, B, C, D, \{E_j\}_{j=0}^{k-1})$ and $RSP = (\{f_{j,i}\}_{j=0,i=1}^{k-1,\beta-1}, \boldsymbol{z}, \boldsymbol{z}_b, \boldsymbol{z}_c)$ be the corresponding values from Protocol 2.

Assumption 1. *Assume $d \geq 7$, $md \geq 86$ and $q > (10\phi_1 d\sqrt{dkr(\beta-1)} + 2)^2$.*

- **RSetup(1^λ):** Run $\boldsymbol{G} \leftarrow \mathsf{CKeygen}(1^\lambda)$ and pick a hash function $H : \{0,1\}^* \to \mathcal{C}^r$ for $\mathcal{C} = \{X^\omega \ : \ \omega \in [0, 2d-1]\}$. Return $ck = \boldsymbol{G}$ and H as $pp = (ck, H)$.
- **RKeygen(pp):** Run $\boldsymbol{r} \leftarrow S_{\mathcal{B}}^m$, $c = \mathrm{Com}_{ck}(\boldsymbol{0} \ ; \ \boldsymbol{r})$ and return $(pk, sk) = (c, \boldsymbol{r})$.
- **RSign$_{pp,sk}(\mathcal{M}, L)$:** Parse $L = (c_0, \ldots, c_{N-1})$ with $c_\ell = \mathrm{Com}_{ck}(\boldsymbol{0} \ ; \ sk)$ where $\ell \in \{0, \ldots, N-1\}$. Continue as follows.
  1. Generate $(CMT_1, \ldots, CMT_r)$ by running $\mathcal{P}(ck, (c_0, \ldots, c_{N-1}), (\ell, sk))[1-7]$ r-times in parallel with the described modifications.
  2. Compute $\boldsymbol{x} = (x_1, \ldots, x_r) = H(ck, \mathcal{M}, L, (CMT_1, \ldots, CMT_r))$.
  3. Compute RSP_i by running $\mathcal{P}(x_i)[9-11]$ with CMT_i for all $i \in \{1, \ldots, r\}$.
  4. If $RSP_i \neq \perp$ for all $i \in \{1, \ldots, r\}$, return $\sigma = (\{CMT_i\}_{i=1}^r, \boldsymbol{x}, \{RSP_i\}_{i=1}^r)$.
  5. Otherwise go to Step 1.

– **RVerify**$_{pp}(\mathcal{M}, L, \sigma)$**:** Parse $\sigma = (\{CMT_i\}_{i=1}^r, \boldsymbol{x}, \{RSP_i\}_{i=1}^r)$, $\boldsymbol{x} = (x_1, \ldots, x_r)$ and $L = (c_0, \ldots, c_{N-1})$. Proceed as follows.
  1. If $\boldsymbol{x} \neq H(ck, \mathcal{M}, L, (CMT_1, \ldots, CMT_r))$, return 0.
  2. For each $i \in \{1, \ldots, r\}$:
     (a) Run Protocol 2's verification with CMT_i, x_i and RSP_i except Step 2.
     (b) If verification fails, return 0.
  3. Return 1.

We can remove A, D, E_0 from the signature as they are uniquely determined by the remaining components, and Step 1 in **RVerify** ensures the relevant protocol verification steps hold. This is a standard technique and we skip the details.

The correctness and anonymity properties of the ring signature follow from the completeness and zero-knowledge properties of Protocol 2, respectively. In particular, the expected number of iterations in **RSign** is $\mu(\phi_1)\mu(\phi_2)$, which is upper-bounded by 3 in the parameter setting. However, the unforgeability proof of the ring signature is not straightforward due to the small challenge space and soundness gap issues. A detailed proof is given in Appendix E.

Theorem 3. *If Assumption 1 holds and the commitment scheme defined in Section 2.2 is T'-binding where $T' = \max\{T_1, \sqrt{\left(24\sqrt{3r} \cdot m\mathcal{B}(k+1)d^{k+1}\right)^2 + 2^{2k}}\}$ for T_1 described with the tweaks, then the ring signature scheme described is unforgeable with respect to insider corruption in the random oracle model.*

6.1 Parameter setting

First of all, we set $\phi_1 = \phi_2 = 22$ to get an acceptance rate of more than 1/3 for the two-step rejection sampling. Such an acceptance rate is greater than or equal to the most commonly used ones such as those in [13, 22, 4, 3] and the expected number of iterations in **RSign** is 3 in this case. Also, we need to ensure that the commitment scheme T'-binding as in Theorem 3. Thus, from the discussion in Appendix C, to make M-SIS secure against lattice attacks, we ensure the following holds

$$\min\left\{q,\ 2^{2\sqrt{n \cdot d \cdot \log q \log \delta}}\right\} > \max\left\{2T_1,\ 2 \cdot 24\sqrt{3r}\mathcal{B}md \cdot (k+1) \cdot \mathbb{B}_{d,k}\right\}. \quad (11)$$

That is, we use Method 2 to bound the extracted witness norm, which does not require the use of Algorithm 2 in the protocol's verification. For the set of (d, k) pairs used in Table 3, the exact value of $\mathbb{B}_{d,k}$ is computed by iterating through the whole search space.

We also set $\mathcal{B} = 1$ as in previous works [3, 23, 14], and make sure that M-LWE$_{m-n,n,q,1}$ is hard using Albrecht et al.'s estimator [2]. The root Hermite factor δ is at most 1.0045 for both M-SIS and M-LWE security estimations. Finally, Assumption 1 is ensured to hold.

Table 3 shows several instances with respect to different ring sizes where the soundness error of the underlying (r-repeated) protocol is $2^{-\lambda}$ and we restrict $\log q \leq 64$. The calculations are done as given in Table 4 in Appendix D. Note

Table 3: Parameters and sizes of our lattice-based ring signature for a root Hermite factor $\delta \leq 1.0045$. The signature sizes are rounded to the nearest integer.

N	64	256	1024	4096	$\sim 2^{16}$	$\sim 2^{20}$	2^{30}
(n, m)	$(5, 13)$	$(5, 13)$	$(11, 25)$	$(21, 50)$	$(20, 51)$	$(40, 101)$	$(41, 106)$
$(d, \log q)$	$(256, 50)$	$(256, 53)$	$(128, 46)$	$(64, 47)$	$(64, 50)$	$(32, 49)$	$(32, 52)$
(k, β)	$(2, 8)$	$(2, 16)$	$(2, 32)$	$(2, 64)$	$(3, 41)$	$(3, 102)$	$(5, 64)$
r	16	16	19	22	24	29	35
λ	128.0	128.0	133.0	132.0	129.96	128.04	128.73
Signature Size (KB)	774	881	1021	1178	1487	1862	3006
User PK Size (KB)	7.81	8.28	7.91	7.71	7.81	7.66	8.33
User SK Size (KB)	0.81	0.81	0.78	0.78	0.80	0.79	0.83

that since r is rounded up, the security parameter λ may be slightly larger than 128. Also, the results from Lemma 4 used to bound the Euclidean norm of a discrete normal vector can be adjusted with respect to the vector dimension. In particular, the constant 5^4 in T_1 (and also T) can be reduced depending on the choice of d. We take this optimization into consideration when setting the parameters.

Acknowledgements. The work of Ron Steinfeld and Amin Sakzad was supported in part by ARC grant DP150100285. Ron Steinfeld and Joseph K. Liu were also supported in part by ARC grant DP180102199.

References

[1] S. Agrawal, C. Gentry, S. Halevi, and A. Sahai. Discrete gaussian leftover hash lemma over infinite domains. In *ASIACRYPT*, pages 97–116. Springer, 2013.

[2] M. R. Albrecht, R. Player, and S. Scott. On the concrete hardness of learning with errors. *Journal of Mathematical Cryptology*, 9(3):169–203, 2015.

[3] C. Baum, I. Damgård, V. Lyubashevsky, S. Oechsner, and C. Peikert. More efficient commitments from structured lattice assumptions. In *SCN*, pages 368–385. Springer, 2018.

[4] C. Baum, H. Lin, and S. Oechsner. Towards practical lattice-based one-time linkable ring signatures. In *ICICS*, volume 11149 of *LNCS*, pages 303–322. Springer, 2018.

[5] A. Bender, J. Katz, and R. Morselli. Ring signatures: Stronger definitions, and constructions without random oracles. *Journal of Cryptology*, 22(1):114–138, 2009.

[6] F. Benhamouda, J. Camenisch, S. Krenn, V. Lyubashevsky, and G. Neven. Better zero-knowledge proofs for lattice encryption and their application to group signatures. In *ASIACRYPT*, pages 551–572. Springer, 2014.

[7] F. Benhamouda, S. Krenn, V. Lyubashevsky, and K. Pietrzak. Efficient zero-knowledge proofs for commitments from learning with errors over rings. In *ESORICS*, pages 305–325. Springer, 2015.

[8] J. Bootle, A. Cerulli, P. Chaidos, E. Ghadafi, J. Groth, and C. Petit. Short accountable ring signatures based on DDH. In *ESORICS*, pages 243–265. Springer, 2015.

[9] J. Bootle, A. Cerulli, P. Chaidos, J. Groth, and C. Petit. Efficient zero-knowledge arguments for arithmetic circuits in the discrete log setting. In *EUROCRYPT*, pages 327–357. Springer, 2016.

[10] E. Brickell, D. Pointcheval, S. Vaudenay, and M. Yung. Design validations for discrete logarithm based signature schemes. In *PKC*, pages 276–292. Springer, 2000.

[11] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, and G. Maxwell. Bulletproofs: Short proofs for confidential transactions and more. In *S&P*. IEEE, 2018.

[12] S. S. M. Chow, J. K. Liu, and D. S. Wong. Robust receipt-free election system with ballot secrecy and verifiability. In *NDSS*. The Internet Society, 2008.

[13] R. del Pino, V. Lyubashevsky, G. Neven, and G. Seiler. Practical quantum-safe voting from lattices. In *CCS*, pages 1565–1581. ACM, 2017.

[14] R. del Pino, V. Lyubashevsky, and G. Seiler. Lattice-based group signatures and zero-knowledge proofs of automorphism stability. In *CCS*, pages 574–591. ACM, 2018.

[15] J. Groth and M. Kohlweiss. One-out-of-many proofs: Or how to leak a secret and spend a coin. In *EUROCRYPT*, volume 9057, pages 253–280. Springer, 2015.

[16] A. Kawachi, K. Tanaka, and K. Xagawa. Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In *ASIACRYPT*, volume 5350, pages 372–389. Springer, 2008.

[17] A. Langlois and D. Stehlé. Worst-case to average-case reductions for module lattices. *Designs, Codes and Cryptography*, 75(3):565–599, 2015.

[18] B. Libert, S. Ling, F. Mouhartem, K. Nguyen, and H. Wang. Signature schemes with efficient protocols and dynamic group signatures from lattice assumptions. In *ASIACRYPT 2016*, pages 373–403. Springer, 2016.

[19] B. Libert, S. Ling, K. Nguyen, and H. Wang. Zero-knowledge arguments for lattice-based accumulators: logarithmic-size ring signatures and group signatures without trapdoors. In *EUROCRYPT*, pages 1–31. Springer, 2016.

[20] S. Ling, K. Nguyen, H. Wang, and Y. Xu. Lattice-based group signatures: Achieving full dynamicity with ease. In *ACNS*, pages 293–312. Springer, 2017.

[21] J. K. Liu, V. K. Wei, and D. S. Wong. Linkable spontaneous anonymous group signature for ad hoc groups (extended abstract). In *ACISP*, volume 3108 of *LNCS*, pages 325–335. Springer, 2004.

[22] V. Lyubashevsky. Lattice signatures without trapdoors. In *EUROCRYPT*, pages 738–755. Springer, 2012. (Full version).

[23] V. Lyubashevsky and G. Neven. One-shot verifiable encryption from lattices. In *EUROCRYPT*, pages 293–323. Springer, 2017.

[24] D. Micciancio and C. Peikert. Hardness of sis and lwe with small parameters. In *Advances in Cryptology–CRYPTO 2013*, pages 21–39. Springer, 2013.

[25] D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM Journal on Computing*, 37(1):267–302, 2007. Preliminary version in FOCS 2004.

[26] D. Micciancio and O. Regev. Lattice-based cryptography. In *Post-quantum cryptography*, pages 147–191. Springer, 2009.

[27] S. Noether. Ring signature confidential transactions for monero. Cryptology ePrint Archive, Report 2015/1098, 2015. https://eprint.iacr.org/2015/1098.

[28] D. Pointcheval and J. Stern. Security arguments for digital signatures and blind signatures. *Journal of cryptology*, 13(3):361–396, 2000.

[29] R. Rivest, A. Shamir, and Y. Tauman. How to leak a secret. *ASIACRYPT*, pages 552–565, 2001.

[30] J. Stern. A new paradigm for public key identification. *IEEE Transactions on Information Theory*, 42(6):1757–1768, 1996.
[31] S. Sun, M. H. Au, J. K. Liu, and T. H. Yuen. Ringct 2.0: A compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency monero. In *ESORICS*, volume 10493 of *LNCS*, pages 456–474. Springer, 2017.
[32] W. A. A. Torres, R. Steinfeld, A. Sakzad, J. K. Liu, V. Kuchta, N. Bhattacharjee, M. H. Au, and J. Cheng. Post-quantum one-time linkable ring signature and application to ring confidential transactions in blockchain (lattice ringct v1. 0). In *ACISP*, pages 558–576. Springer, 2018.
[33] L. R. Turner. Inverse of the vandermonde matrix with applications. Technical Report NASA-TN-D-3547, Lewis Research Center, NASA, 1966.
[34] R. Yang, M. H. Au, J. Lai, Q. Xu, and Z. Yu. Lattice-based techniques for accountable anonymity: Composition of abstract sterns protocols and weak prf with efficient protocols from lwr. Cryptology ePrint Archive, Report 2017/781, 2017. https://eprint.iacr.org/2017/781.
[35] H. Zhang, F. Zhang, H. Tian, and M. H. Au. Anonymous post-quantum cryptocash. Cryptology ePrint Archive, Report 2017/716, 2017. https://eprint.iacr.org/2017/716 (To appear in FC 2018).

A A Brief Discussion on the Protocol in [35]

In this section, we list some of the issues in the security proofs given in [35]. In an earlier version of [35], it was claimed without providing an explicit definition of the underlying protocol that the anonymity and unforgeability of their ring signature scheme followed directly from the results of [15], provided that a perfectly hiding and computationally binding commitment scheme is used. Later, the authors revised their claims, and provided an explicit one-out-of-many protocol. However, we see that still not all security issues are addressed properly. Since the issues are related to the details of the protocol used as a building block for the ring signature, we focus on Appendix B of [35] (01-Apr-2018 version on IACR's eprint archive).

The authors first claim that for a prime q and a power-of-two n, $X^n + 1 \in \mathbb{Z}_q[X]$ is irreducible and $\mathbb{Z}_q[X]/(X^n + 1)$ is a field, which clearly does not hold. Furthermore, the distribution of the prover's responses in their protocol depends on secret witness values (similar to our case). It is not mentioned at all how this issue is tackled as the simulator is unaware of these values and straightforward simulation (without using a technique such as rejection sampling) does not work.

Moreover, the authors also claim that the invertibility of a Vandermonde matrix formed by challenges $x_0, \ldots, x_k$ over R_q follows when x_i's are distinct and invertible in R_q. As we have clearly shown, we need the *differences* of challenges, $(x_i - x_j)$, to be invertible in R_q. The authors do not consider anything about the invertibility of the challenge differences with respect to the challenge space they use. In addition, in the special soundness proof for their one-out-of-many protocol, the authors assume that the accepting transcripts used for witness extraction are well-formed (as it would happen in an honest run of the protocol). No assumption on how the accepting transcripts are generated can be made as they are provided by a (possibly) cheating prover.

B Additional Related work

A ring signature enables one to sign a message on behalf of an ad hoc group, called *ring*, of users without revealing the actual signatory. The ring is formed by gathering public keys and no consent is required from the other users. Ring signatures were introduced by Rivest, Shamir and Tauman-Kalai [29] and the rigorous security notions were established in the work of Bender, Katz and Morselli [5]. The work in this area is relatively scarce and currently, the only log-size (in the number of ring members) ring signatures based on number-theoretic assumptions are due to [15] and [8], where the main ideas in the latter are borrowed from the former. In [15, 8], the authors first describe efficient (in terms of communication complexity) one-out-of-many proofs, which then enables them to design short ring signatures in the DL setting. On the side of lattice setting, most of the existing ring signature schemes (e.g., [16, 32, 4]) have linear size.

As mentioned earlier, [35] attempts to extend Groth-Kohlweiss' scheme [15] by replacing Pedersen commitment with a lattice-based commitment scheme. It is claimed that the security requirements for the instantiation with this lattice-based commitment follows from the results of [15]. We found that this does not hold true without addressing the issues detailed in Section 1.1, which is also hinted in the works [19, 4] by noting that Groth-Kohlweiss' scheme does not easily extend to the lattice setting. Moreover, even if the security issues were to be solved, [35] leads to inefficient parameters without our techniques.

This leaves us with the work of Libert et al. [19] (and a follow-up by [34], adding linkability to [19]) as the only log-sized ring signature from lattices. In [19], the authors first design an accumulator through a Merkle tree using SIS-based hash function. Zero-knowledge membership arguments are then built for this accumulator. Having these building blocks, the authors propose ring and group signatures, both of which are log-sized in the number of users involved. We therefore focus on [19] for efficiency comparison purposes.

Most of the existing lattice-based zero-knowledge proofs requiring an extraction of a small witness make use of either binary challenges or use Stern-type protocols [30], providing soundness errors of $1/2$ and $2/3$, respectively. These approaches inherently require more than 100 repetitions to achieve a negligible soundness error, say 2^{-100}. Benhamouda et al. [6] introduced a different challenge space in the Ring-LWE setting consisting of *monomial challenges* of the form $X^i \in R = \mathbb{Z}[X]/(X^d + 1)$ and proved that the (Euclidean) norm of the doubled inverse differences of such challenges is at most $\sqrt{d}$ (i.e., $\left\|2(X^i - X^j)^{-1}\right\| \leq \sqrt{d}$). If we consider a ring dimension $d = 2^{10}$, this approach requires only 10 repetitions to achieve the same soundness error of 2^{-100}.

C More Discussion about the Commitment Scheme

It is clear that as θ gets smaller, M-SIS problem becomes harder. That's why if the commitment scheme is T_{com}-binding, it is also T-binding for any $T \leq T_{\text{com}}$.

It is also well known that M-LWE problem gets harder if the error is sampled from a wider distribution (i.e., as the standard deviation of the distribution gets

larger). In our instantiation, we will measure the security of M-LWE with respect to the easiest case when $e \leftarrow S_{\mathcal{B}}$, and this ensures that the commitment scheme is also hiding with respect to the harder cases when the error is sampled from a wider distribution.

For estimating M-LWE hardness, we use the well-known LWE estimator by Albrecht et al. [2]. For the computational hardness of M-SIS, similar to previous works [22, 3], we use the results of [26]. For $n_L = n \cdot d$, they show that state-of-the-art lattice reduction algorithms find a non-zero vector of length

$$\min\left\{q, 2^{2\sqrt{n_L \log q \log \delta}}\right\}, \tag{12}$$

where δ is the root Hermite factor depending on the quality of the lattice reduction algorithm. Albrecht et al.'s LWE estimator (when run using both sieving and enumeration techniques) shows that a root Hermite factor of $\delta = 1.0045$ provides 128-bit post-quantum security. Hence, we adapt $\lambda = 128$ and $\delta = 1.0045$ for both M-SIS and M-LWE. By Lemma 1, to ensure T-binding property for $T \in \mathbb{R}^+$, we make sure that $2T$ is strictly smaller than (12). Finally, the following homomorphic properties hold: $\mathrm{Com}_{ck}(\boldsymbol{a}\,;\,\boldsymbol{r}_1) + \mathrm{Com}_{ck}(\boldsymbol{b}\,;\,\boldsymbol{r}_2) = \mathrm{Com}_{ck}(\boldsymbol{a}+\boldsymbol{b}\,;\,\boldsymbol{r}_1+\boldsymbol{r}_2)$ and $\gamma \cdot \mathrm{Com}_{ck}(\boldsymbol{a}\,;\,\boldsymbol{r}) = \mathrm{Com}_{ck}(\gamma\cdot\boldsymbol{a}\,;\,\gamma\cdot\boldsymbol{r})$ for any $\gamma \in R_q$.

D Additional Material about Ring Signature

Table 4: Calculation of parameters and sizes for the ring signature in Section 6.

	Notation/Formula	**Notes**
Security parameter	λ	
Dimension of randomness vector	m	Chosen based on LWE estimator of [2]
Soundness error	$\eta = \frac{\max\{2,\, \log_\beta N\}}{2d}$	Recall that $k = \log_\beta N$
Number of protocol repetitions	$r = \lceil -\frac{\lambda}{\log \eta} \rceil$	
Number of commitments	$N_c = k + 1$	$B, C, E_1, \ldots, E_{k-1}$
Number of $f_{j,i}$ values	$N_f = k \cdot (\beta - 1)$	$f_{0,1}, \ldots, f_{k-1,\beta-1} \in D_{\phi_1\sqrt{kr}}$
Number of randomness	$N_R = 3$	$\boldsymbol{z}, \boldsymbol{z}_b, \boldsymbol{z}_c \in D^{md}_{\phi_2\mathcal{B}\sqrt{3mdr}}$
Ring Signature size	$r \cdot [N_c \cdot (nd \log q)$ $+N_f \cdot d \cdot \log(12\phi_1\sqrt{kr})$ $+N_R \cdot (md \cdot \log(12\phi_2\mathcal{B}\sqrt{3mdr}))]$	
User public key size	$nd \log q$	A commitment in $R_q^{n\times 1}$
User secret key size	$md \cdot \log(2\mathcal{B}+1)$	A randomness vector in $\{-\mathcal{B}, \ldots, \mathcal{B}\}^{md}$. For $\mathcal{B} = 1$, we simply take $\log(2\mathcal{B}+1) = 2$.

D.1 Definitions

We recall the standard definitions and properties of a ring signature, which consists of four algorithms (**RSetup**, **RKeygen**, **RSign**, **RVerify**) as follows.

- $pp \leftarrow$**RSetup**(1^λ)**:** On input a security parameter λ, generates the public parameters pp, which are assumed to be made available to everyone.
- $(pk, sk) \leftarrow$**RKeygen**(pp)**:** Given pp, outputs a public-secret key pair (pk, sk).
- $\sigma \leftarrow$**RSign**$_{pp,sk}(\mathcal{M}, L)$**:** On input a message $\mathcal{M}$ and a ring L of public keys, outputs a signature σ on $\mathcal{M}$ with respect to L. It is required that sk is generated by **RKeygen**(pp), and the corresponding public key pk is in L.
- $\{0,1\} \leftarrow$**RVerify**$_{pp}(\mathcal{M}, L, \sigma)$**:** On input a purported signature σ on a message $\mathcal{M}$ with respect to a ring L, checks the validity of σ. If it is valid, outputs 1 or outputs 0 otherwise.

Definition 6 (Correctness). *A ring signature (**RSetup**, **RKeygen**, **RSign**, **RVerify**) provides statistical correctness if for any $pp \leftarrow$ **RSetup**, any $(pk, sk) \leftarrow$**RKeygen**(pp), any L such that $pk \in L$, and any $\mathcal{M} \in \{0,1\}^*$, the following is satisfied*

$$\Pr[\ \mathbf{RVerify}_{pp}(\mathcal{M}, L, \mathbf{RSign}_{pp,sk}(\mathcal{M}, L)) = 1\ ] = 1 - \mathrm{negl}(\lambda).$$

Definition 7 (Anonymity). *A ring signature (**RSetup**, **RKeygen**, **RSign**, **RVerify**) provides statistical anonymity if for any (possibly unbounded) adversary $\mathcal{A}$*

$$\Pr\left[\begin{matrix} pp \leftarrow \mathbf{RSetup}(1^\lambda); (\mathcal{M}, i_0, i_1, L) \leftarrow \mathcal{A}^{\mathbf{RKeygen}(pp)} \\ b \leftarrow \{0,1\};\ \sigma \leftarrow \mathbf{RSign}_{pp}(sk_{i_b}, \mathcal{M}, L) \end{matrix} : \mathcal{A}(\sigma) = b\right] = \frac{1}{2} + \mathrm{negl}(\lambda),$$

*where $pk_{i_0}, pk_{i_1} \in L$ and $(pk_{i_0}, sk_{i_0}), (pk_{i_1}, sk_{i_1}) \leftarrow$ **RKeygen**(pp).*

Definition 8 (Unforgeability w.r.t. insider corruption). *A ring signature (**RSetup**, **RKeygen**, **RSign**, **RVerify**) is unforgeable with respect to insider collusion if for all PPT adversary $\mathcal{A}$*

$$\Pr\left[\begin{matrix} pp \leftarrow \mathbf{RSetup}(1^\lambda); \\ (\mathcal{M}, L, \sigma) \leftarrow \mathcal{A}^{\mathrm{PKGen,Sign,Corrupt}}(pp) \end{matrix} : \mathbf{RVerify}(\mathcal{M}, L, \sigma) = 1\right] = \mathrm{negl}(\lambda),$$

where

- PKGen*: on the i-th query, picks a randomness ρ_i, runs $(pk_i, sk_i) \leftarrow$* **RKeygen**$(pp; \rho_i)$ *and returns pk_i.*
- Sign*(i, M, L): returns $\sigma \leftarrow$* **RSign**$_{pp,sk_i}(\mathcal{M}, L)$*, provided (pk_i, sk_i) has been generated by* PKGen*.*
- Corrupt*(i): returns ρ_i (enabling the computation of sk_i), provided (pk_i, sk_i) has been generated by* PKGen*.*
- *$\mathcal{A}$ outputs $(\mathcal{M}, L, \sigma)$ such that* Sign*$(\cdot, \mathcal{M}, L)$ has not been queried and L only contains pk_i's generated by* PKGen *where* Corrupt*(i) has not been queried.*

E Proofs of Theorems and Lemmas

Proof (Lemma 1). Let $(y, \boldsymbol{m}, \boldsymbol{r})$ and $(y, \boldsymbol{m}', \boldsymbol{r}')$ with $(\boldsymbol{r}, \boldsymbol{m}) \neq (\boldsymbol{r}', \boldsymbol{m}')$ be two valid openings of a commitment C. That is,

$$yC = \mathrm{Com}_{ck}(\boldsymbol{m}\ ;\ \boldsymbol{r}) = \mathrm{Com}_{ck}(\boldsymbol{m}'\ ;\ \boldsymbol{r}') \quad \text{and} \quad \|(\boldsymbol{r}, \boldsymbol{m})\|, \|(\boldsymbol{r}', \boldsymbol{m}')\| \leq T_{\mathrm{com}}.$$

Therefore, we have $yC = \boldsymbol{G} \cdot (\boldsymbol{r}, \boldsymbol{m}) = \boldsymbol{G} \cdot (\boldsymbol{r}', \boldsymbol{m}')$, which implies $\boldsymbol{G} \cdot (\boldsymbol{r} - \boldsymbol{r}', \boldsymbol{m} - \boldsymbol{m}') = 0$. Hence, $(\boldsymbol{r} - \boldsymbol{r}', \boldsymbol{m} - \boldsymbol{m}')$ is a solution to M-SIS$_{n,m+v,q,2T_{\mathrm{com}}}$ problem (in Hermite normal form). This proves the computational strong binding with respect to the same relaxation factor y.

For the hiding property, we can write $\mathrm{Com}_{ck}(\boldsymbol{m}\ ;\ \boldsymbol{r}) = \boldsymbol{G}_r \cdot \boldsymbol{r} + \boldsymbol{G}_m \cdot \boldsymbol{m} = \boldsymbol{r}_0 + \boldsymbol{G}'_r \cdot \boldsymbol{r}_1 + \boldsymbol{G}_m \cdot \boldsymbol{m}$ where $\boldsymbol{r} = (\boldsymbol{r}_0, \boldsymbol{r}_1)$ since $\boldsymbol{G}_r = [\, \boldsymbol{I}_n \,\|\, \boldsymbol{G}'_r \,]$. The result of the computation $\boldsymbol{r}_0 + \boldsymbol{G}'_r \cdot \boldsymbol{r}_1$ gives n M-LWE samples with $\boldsymbol{r}_1 \in S_{\mathcal{B}}^{m-n}$ as the secret key. Therefore, if M-LWE$_{m-n,n,q,\mathcal{B}}$ is hard, $\boldsymbol{r}_0 + \boldsymbol{G}'_r \cdot \boldsymbol{r}_1$ looks uniformly random in R_q^n and so does commitments to any message. □

Proof (Lemma 7). Since $\|b\| + \alpha < \sqrt{q}$, we have $\|b\| < \sqrt{q}$. Then, we get

$$\|b \cdot (\alpha - b)\|_\infty \leq \|b\| \cdot \|\alpha - b\| \leq \|b\| \cdot (\|b\| + \alpha) < \sqrt{q} \cdot \sqrt{q} = q.$$

Therefore, $b \cdot (\alpha - b) = 0$ holds over R. Since $X^d + 1$ is irreducible over $\mathbb{Q}$, we get $b \in \{0, \alpha\}$. □

Proof (Lemma 9).

$$\begin{aligned}\left\|2^k \boldsymbol{r}_{\mathrm{ext}}\right\| = \left\|\sum_{j=0}^{k} 2^k \alpha_j \boldsymbol{r}_{x_j}\right\| &\leq (k+1) \cdot \max_{0 \leq j \leq k} \left\|2^k \alpha_j \boldsymbol{r}_{x_j}\right\| \\ &\leq (k+1)\sqrt{d} \max_{0 \leq j \leq k} \left\|2^k \alpha_j\right\| \max_{0 \leq j \leq k} \left\|\boldsymbol{r}_{x_j}\right\| \leq (k+1) \cdot d^k \max_{0 \leq j \leq k} \left\|\boldsymbol{r}_{x_j}\right\|. \quad (13)\end{aligned}$$

A similar result follows analogously for $\boldsymbol{m}_{\mathrm{ext}}$. □

Proof (Lemma 11). By Fact 1, $\sigma_n(\boldsymbol{R}_j^\top) = \sigma_n(\boldsymbol{R}_j) = \sigma_n(\boldsymbol{S}_j \otimes \boldsymbol{I}_m) = \sigma_n(\boldsymbol{S}_j)$ for any $0 \leq j \leq k$. Again, by Fact 1, we have

$$\sigma_n(\boldsymbol{S}_j) = \sigma_n\left(\prod_{i=0, i \neq j}^{k} \mathrm{Rot}\left(\frac{2}{x_j - x_i}\right)\right) \geq \prod_{i=0, i \neq j}^{k} \sigma_n\left(\mathrm{Rot}\left(\frac{2}{x_j - x_i}\right)\right).$$

We have verified by computation that $\sigma_n\left(\mathrm{Rot}\left(\frac{2}{x_j - x_i}\right)\right) \geq 1$ for any pair of monomial challenges x_j, x_i and any $d \in \{4, 8, \ldots, 512\}$. As a result, $\sigma_n(\boldsymbol{R}_j^\top) \geq 1$ is always satisfied with the given assumptions. Thus, using Fact 1 and Fact 3, we have

$$\sigma_n(s\boldsymbol{R}_j^\top) \geq s \cdot \sigma_n(\boldsymbol{R}_j^\top) \geq 6 > \eta_\epsilon(\mathbb{Z}^{md}).$$

Since $\sigma_1(\boldsymbol{S}_j) = \sigma_1(\boldsymbol{R}_j)$ by Fact 1, the rest follows from Lemma 2 as sketched in Method 3. □

Proof (Theorem 1). **Completeness:** By Lemma 6, prover responds with probability statistically close to $1/(\mu(\phi_1)\mu(\phi_2))$, and distributions of $f_{j,i}$'s ($i \neq 0$) are statistically close to $D^d_{\phi_1\sqrt{k}}$ and that of $\boldsymbol{z}_b, \boldsymbol{z}_c$ are statistically close to $D^{md}_{\phi_2\mathcal{B}\sqrt{2md}}$ since

$$\|(x \cdot b_{0,1}, \ldots, x \cdot b_{k-1,\beta-1})\| \leq \sqrt{k}, \quad \text{and} \quad \|(x \cdot \boldsymbol{r}, x \cdot \boldsymbol{r}_c)\| \leq \mathcal{B}\sqrt{2md}.$$

Since the standard deviation of all sampled discrete normal coefficients are much larger than 6, the sum of discrete normal samples behave as in the continuous case by Fact 3 and [24, Theorem 3.3]. That is, the distribution of $\sum_{i=1}^{\beta-1} f_{j,i}$ is statistically close to $D^d_{\phi_1\sqrt{k(\beta-1)}}$. Therefore, if the prover does not abort, and since $d \geq 7$ and $md \geq 86$, by Lemma 4 except with probability at most 2^{-100}, we have,

$$\|f_{j,i}\| \leq 5 \cdot \phi_1\sqrt{k} \cdot \sqrt{d} = 5\phi_1\sqrt{dk}, \qquad \forall j \in [0, k-1], \forall i \in [1, \beta-1],$$
$$\|f_{j,0}\| = \|x - \sum_{i=1}^{\beta-1} f_{j,i}\| \leq 5 \cdot \phi_1\sqrt{k(\beta-1)} \cdot \sqrt{d} = 5\phi_1\sqrt{dk(\beta-1)}, \ \ \forall j \in [0, k-1],$$

and $\|\boldsymbol{z}_b\|, \|\boldsymbol{z}_c\| \leq 2 \cdot \phi_2\mathcal{B}\sqrt{2md} \cdot \sqrt{md} = 2\phi_2\sqrt{2}\mathcal{B}md$, proving the bounds on the norms. The other verification steps follow via straightforward investigation.
SHVZK: Given a challenge x, the simulator outputs $(\mathrm{aCom}(0), x, \perp)$ indicating an abort with probability $1 - 1/(\mu(\phi_1)\mu(\phi_2))$. Otherwise, it picks $C \leftarrow R_q^n$, $f_{j,i} \leftarrow D^d_{\phi_1\sqrt{k}}$ for all $0 \leq j \leq k-1$ and $1 \leq i \leq \beta-1$, and also $\boldsymbol{z}_b, \boldsymbol{z}_c \leftarrow D^{md}_{\phi_2\mathcal{B}\sqrt{2md}}$. Then, it sets $f_{j,0} = x - \sum_{i=1}^{\beta-1} f_{j,i}$ for all $j = 0, \ldots, k-1$. Finally, it computes $A = \mathrm{Com}_{ck}(\boldsymbol{f}\ ;\ \boldsymbol{z}_b) - xB$, $D = \mathrm{Com}_{ck}(\{f_{j,i}(x - f_{j,i})\}_{j,i}\ ;\ \boldsymbol{z}_c) - xC$ and $(c_a, d_a) = \mathrm{aCom}(A, C, D)$ where $\boldsymbol{f} = (f_{0,0}, \ldots, f_{k-1,\beta-1})$. It outputs the simulated transcript $(c_a, x, (d_a, \{f_{j,i}\}_{j=0,i=1}^{k-1,\beta-1}, A, C, D, \boldsymbol{z}_b, \boldsymbol{z}_c))$.

Note that the narrowest distribution where a randomness coefficient is sampled from is $\mathcal{U}(\{-\mathcal{B}, \ldots, \mathcal{B}\})$ and M-LWE$_{m-n,n,q,\mathcal{B}}$ is assumed to be hard. Therefore, by Lemma 1, all of the commitments are computationally indistinguishable from uniformly random elements in R_q^n. Hence, if the protocol is not aborted, the real and simulated transcripts are indistinguishable by Lemma 6 and the hiding property of the commitment scheme. If an abort occurs, then the indistinguishability is satisfied due to hiding property of aCom and the fact that the probability of having an abort is the same for all x.
3-special soundness: Given 3 accepting transcripts, by the binding property of aCom, we have the tuples $(A, C, D, x, f_{0,1}, \ldots, f_{k-1,\beta-1}, \boldsymbol{z}_b, \boldsymbol{z}_c)$, $(A, C, D, x', f'_{0,1}, \ldots, f'_{k-1,\beta-1}, \boldsymbol{z}'_b, \boldsymbol{z}'_c)$, $(A, C, D, x'', f''_{0,1}, \ldots, f''_{k-1,\beta-1}, \boldsymbol{z}''_b, \boldsymbol{z}''_c)$. Let $\boldsymbol{f} = (f_{0,0}, \ldots, f_{k-1,\beta-1})$, $\boldsymbol{f}' = (f'_{0,0}, \ldots, f'_{k-1,\beta-1})$, $\boldsymbol{f}'' = (f''_{0,0}, \ldots, f''_{k-1,\beta-1})$ where $f_{j,0}, f'_{j,0}, f''_{j,0}$'s are computed as in the verification. Then, by Step 7 in the verification, we have $xB + A = \mathrm{Com}_{ck}(\boldsymbol{f}\ ;\ \boldsymbol{z}_b)$ and $x'B + A = \mathrm{Com}_{ck}(\boldsymbol{f}'\ ;\ \boldsymbol{z}'_b)$. By subtracting the equations and multiplying both sides by $2(x - x')^{-1}$, we get

$$2B = \mathrm{Com}_{ck}(2(x - x')^{-1}(\boldsymbol{f} - \boldsymbol{f}')\ ;\ 2(x - x')^{-1}(\boldsymbol{z}_b - \boldsymbol{z}'_b)) =: \mathrm{Com}_{ck}(\hat{\boldsymbol{b}}\ ;\ \hat{\boldsymbol{r}}_b).$$

This gives us openings of $2B$ as $\hat{\boldsymbol{b}} = (\hat{b}_{0,0}, \ldots, \hat{b}_{k-1,\beta-1})$ and $\hat{\boldsymbol{r}}_b$. Note that

$$\begin{aligned}\|\hat{\boldsymbol{r}}_b\| = \left\|2(x-x')^{-1}(\boldsymbol{z}_b - \boldsymbol{z}'_b)\right\| &\leq \sqrt{d} \cdot \left\|2(x-x')^{-1}\right\| \cdot \|(\boldsymbol{z}_b - \boldsymbol{z}'_b)\| \\ &\leq d \cdot \|(\boldsymbol{z}_b - \boldsymbol{z}'_b)\| \leq d \cdot 2 \cdot 2\sqrt{2}\phi_2 \mathcal{B} m d = 4\sqrt{2}\phi_2 \mathcal{B} m d^2,\end{aligned}$$

which proves the required norm-bound on the extracted randomness for $\mathcal{R}'_{\text{bin}}$.

We can also recover openings of $2A$ by computing $\hat{a}_{j,i} = 2f_{j,i} - x \cdot \hat{b}_{j,i}$ and $\hat{\boldsymbol{r}}_a = 2\boldsymbol{z}_b - x \cdot \hat{\boldsymbol{r}}_b$. Similarly, by Step 8 of the verification, we get openings $\hat{c}_{j,i}$ and $\hat{d}_{j,i}$ of $2C$ and $2D$, respectively, such that $2g_{j,i} = x\hat{c}_{j,i} + \hat{d}_{j,i}$ and $g_{j,i} = f_{j,i}(x - f_{j,i})$. From here, by multiplying the former by 2, we get

$$\begin{aligned}2 \cdot \left(x \cdot \hat{c}_{j,i} + \hat{d}_{j,i}\right) = 2 \cdot 2g_{j,i} &= 2 \cdot (2f_{j,i}(x - f_{j,i})) = 2f_{j,i}(2x - 2f_{j,i}) \\ &= x^2 \left[\hat{b}_{j,i}(2 - \hat{b}_{j,i})\right] + x \left[2\hat{a}_{j,i}(1 - \hat{b}_{j,i})\right] - \hat{a}^2_{j,i},\end{aligned}$$

which implies

$$x^2 \left[\hat{b}_{j,i}(2 - \hat{b}_{j,i})\right] + x \left[2\hat{a}_{j,i}(1 - \hat{b}_{j,i}) - 2\hat{c}_{j,i}\right] - \hat{a}^2_{j,i} - 2\hat{d}_{j,i} = 0. \tag{14}$$

By Lemma 13 in the appendices, norms of the openings of $2A, 2B, 2C, 2D$ are all smaller than T. By the T-binding property of the commitment scheme, PPT prover cannot know other openings of $2A, 2B, 2C$ or $2D$. Thus, (14) also holds for the other challenges x' and x'' with the same $\hat{a}_{j,i}, \hat{b}_{j,i}, \hat{c}_{j,i}, \hat{d}_{j,i}$'s. Then, we can write this system of equations as

$$\begin{pmatrix} 1 & x & x^2 \\ 1 & x' & x'^2 \\ 1 & x'' & x''^2 \end{pmatrix} \cdot \begin{pmatrix} -\hat{a}^2_{j,i} - 2\hat{d}_{j,i} \\ 2\hat{a}_{j,i}(1 - \hat{b}_{j,i}) - 2\hat{c}_{j,i} \\ \hat{b}_{j,i}(2 - \hat{b}_{j,i}) \end{pmatrix} = \mathbf{0} \qquad \text{over } R_q.$$

The left-most matrix is a Vandermonde matrix $\boldsymbol{V}$, which is invertible by Lemma 8. Therefore, we get $\hat{b}_{j,i}(2 - \hat{b}_{j,i}) = 0$ over R_q. Further, we have

$$\begin{aligned}\|\hat{b}_{j,i}\| = \|2(x-x')^{-1}(f_{j,i} - f'_{j,i})\| &\leq \sqrt{d} \cdot \left\|2(x-x')^{-1}\right\| \cdot \left\|f_{j,i} - f'_{j,i}\right\| \\ &\leq d \cdot \left\|f_{j,i} - f'_{j,i}\right\| \leq d \cdot 2 \cdot (5\phi_1\sqrt{dk(\beta-1)}) = 10\phi_1 d\sqrt{dk(\beta-1)}.\end{aligned}$$

Since $q > \left(10\phi_1 d\sqrt{dk(\beta-1)} + 2\right)^2 \geq (\|\hat{b}_{j,i}\| + 2)^2$, we have $\hat{b}_{j,i} = 2b_{j,i}$ for $b_{j,i} \in \{0, 1\}$ by Lemma 7. Moreover, by construction, for all $j = 0, \ldots, k-1$,

$$2x = \sum_{i=0}^{\beta-1} 2f_{j,i} = x \cdot \sum_{i=0}^{\beta-1} 2b_{j,i} + \sum_{i=0}^{\beta-1} \hat{a}_{j,i} = 2x \cdot \sum_{i=0}^{\beta-1} b_{j,i} + \sum_{i=0}^{\beta-1} \hat{a}_{j,i}.$$

If this is true for 2 distinct challenges x and x', then $\sum_{i=0}^{\beta-1} b_{j,i} = 1$ for all $j = 0, \ldots, k-1$ as desired. Finally, since the protocol is 3-special sound and $|\mathcal{C}| = 2d$, the soundness error is $2/(2d) = 1/d$. □

Proof (Theorem 2). **Completeness:** Note that multiplication by x in R_q simply performs a nega-cyclic rotation of the coefficients of a polynomial and thus the distribution of $\sum_{j=0}^{k-1} x^j \boldsymbol{\rho}_j$ is statistically close to $D^{md}_{\phi_2 \mathcal{B}\sqrt{3md}}$ due to the fact that sum of independent discrete normal variables behave as in the continuous case as discussed in the proof of Theorem 1. From here the bounds on the norms of each component follow similar to the completeness proof of Theorem 1.

All the remaining but the last verification steps also follow straightforwardly. To prove that the last verification step holds for honestly generated values, we have, for $c_\ell = \mathrm{Com}_{ck}(\boldsymbol{m}_\ell \; ; \; \boldsymbol{r})$,

$$\begin{aligned}
&\sum_{i=0}^{N-1} \left(\prod_{j=0}^{k-1} f_{j,i_j} \right) c_i - \sum_{j=0}^{k-1} E_j x^j = \sum_{i=0}^{N-1} p_i(x) c_i - \sum_{j=0}^{k-1} \left(\sum_{i=0}^{N-1} p_{i,j} c_i + \mathrm{Com}_{ck}(\mathbf{0} \; ; \; \boldsymbol{\rho}_j) \right) x^j \\
&= \sum_{i=0}^{N-1} p_i(x) c_i - \sum_{j=0}^{k-1} \sum_{i=0}^{N-1} p_{i,j} c_i x^j - \sum_{j=0}^{k-1} x^j \cdot \mathrm{Com}_{ck}(\mathbf{0} \; ; \; \boldsymbol{\rho}_j) \\
&= \sum_{i=0}^{N-1} c_i \left(p_i(x) - \sum_{j=0}^{k-1} p_{i,j} x^j \right) - \sum_{j=0}^{k-1} x^j \cdot \mathrm{Com}_{ck}(\mathbf{0} \; ; \; \boldsymbol{\rho}_j) \\
&= \sum_{i=0}^{N-1} c_i \delta_{\ell,i} x^k - \sum_{j=0}^{k-1} x^j \cdot \mathrm{Com}_{ck}(\mathbf{0} \; ; \; \boldsymbol{\rho}_j) = x^k \cdot c_\ell - \sum_{j=0}^{k-1} x^j \cdot \mathrm{Com}_{ck}(\mathbf{0} \; ; \; \boldsymbol{\rho}_j) \\
&= \mathrm{Com}_{ck}(x^k \boldsymbol{m}_\ell \; ; \; x^k \boldsymbol{r} - \sum_{j=0}^{k-1} x^j \boldsymbol{\rho}_j) = \mathrm{Com}_{ck}(x^k \boldsymbol{m}_\ell \; ; \; \boldsymbol{z}) = \mathrm{Com}_{ck}(\mathbf{0} \; ; \; \boldsymbol{z}) \text{ if } \boldsymbol{m}_\ell = \mathbf{0}.
\end{aligned}$$

SHVZK: Given a challenge x, the simulator outputs $(\mathrm{aCom}(0), x, \perp)$ indicating an abort with probability $1 - \frac{1}{\mu(\phi_1)\mu(\phi_2)}$. Otherwise, it picks $B, C, E_1, \ldots, E_{k-1} \leftarrow R_q^n$ and $f_{j,i} \leftarrow D^d_{\phi_1\sqrt{k}}$ for all $0 \le j \le k-1$ and $1 \le i \le \beta - 1$, and also picks $\boldsymbol{z}, \boldsymbol{z}_b, \boldsymbol{z}_c \leftarrow D^{md}_{\phi_2 \mathcal{B}\sqrt{3md}}$. Then, it calculates $f_{j,0} = x - \sum_{i=1}^{\beta-1} f_{j,i}$ for all $0 \le j \le k-1$, and computes E_0 so as to ensure that the last verification equation is satisfied. Similarly, it computes A and D so that the corresponding verification equations are satisfied. Then, it calculates $(c_a, d_a) = \mathrm{aCom}(A, B, C, D, \{E_j\}_{j=0}^{k-1})$ and outputs the simulated transcript

$$(c_a, x, (d_a, \{f_{j,i}\}_{i \neq 0}, A, B, C, D, \{E_j\}_{j=0}^{k-1}, \boldsymbol{z}, \boldsymbol{z}_b, \boldsymbol{z}_c)).$$

Note that the narrowest distribution where a randomness coefficient is sampled from is $\mathcal{U}(\{-\mathcal{B}, \ldots, \mathcal{B}\})$ and M-LWE$_{m-n,n,q,\mathcal{B}}$ is assumed to be hard. Therefore, by Lemma 1, all of the commitments are computationally indistinguishable from uniformly random elements in R_q^n. Hence, if the protocol is not aborted, the real and simulated transcripts are indistinguishable by Lemma 6, the hiding property of the commitment scheme and the fact that A, D, E_0 are uniquely determined by the verification equations given all the other components in both the real proof and the simulation. If an abort occurs, then the indistinguishability is satisfied due to hiding property of aCom and the fact that the probability of having an abort is the same for all x. □

Proof (Theorem 3). We prove the unforgeability by showing if there exists a PPT forger with a polynomial running time and a non-negligible success probability, then one can break the binding property of the commitment scheme for message and randomness of maximum Euclidean norms 2^k and $24\sqrt{3r} \cdot m\mathcal{B}(k+1)d^{k+1}$, respectively. This implies that one can find a solution to Module-SIS$_{n,m+k\beta,q,\theta}$ problem for $\theta = 2\sqrt{\left(24\sqrt{3r} \cdot m\mathcal{B}(k+1)d^{k+1}\right)^2 + 2^{2k}}$ by Lemma 1.

Let $\mathcal{C}^r$ be the range of H (i.e., each output component of H is in $\mathcal{C}$), Ψ be the set of all random tapes that could be used by a PPT adversary $\mathcal{A}$, and Φ be the set of all random tapes defining the random oracle H. Let $\boldsymbol{x}_j = (\boldsymbol{x}_{j,1}, \ldots, \boldsymbol{x}_{j,r})$ be the output of j-th random oracle query. We partition Φ into Φ_{j_-}, $\boldsymbol{x}_j$ and Φ_{j_+} so that Φ_{j_-}, Φ_{j_+} represent the sets of random tapes defining the random oracle outputs up to j-th query (i.e., $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{j-1}$) and after j-th query (i.e., $\boldsymbol{x}_{j+1}, \ldots, \boldsymbol{x}_Q$), respectively. Therefore, the tuple $(\phi_{j_-}, \boldsymbol{x}_j, \phi_{j_+})$ defines all the random oracle outputs. Further, assume that $\mathcal{A}$ makes q_P, q_S, q_H queries to PKGen, Sign and the random oracle, respectively. Hence, $\mathcal{A}$ makes at most $Q = q_S + q_H$ random oracle queries in total. Suppose that $\mathcal{A}$ has running time $T_A = \text{poly}(\lambda)$ and a probability $\varepsilon = 1/\text{poly}(\lambda) > 4Q\eta$ of generating a successful forgery where $\eta = (k/|\mathcal{C}|)^r$.

We construct an adversary $\mathcal{D}$ against the binding property of the commitment scheme with a running time $T_B = \text{poly}(\lambda)$ and non-negligible success probability $\varepsilon_B = 1/\text{poly}(\lambda)$. On input a commitment key ck, $\mathcal{D}$ works as follows.

1. Pick $t \leftarrow \{1, \ldots, q_P\}$.
2. Set $pk_t = \text{Com}_{ck}(\mathbf{1} \; ; \; \boldsymbol{r}_t)$ for some randomness $\boldsymbol{r}_t \in \{-\mathcal{B}, \ldots, \mathcal{B}\}^{md}$ where $\mathbf{1} = (1, 0, \ldots, 0) \in \{0,1\}^k$ (observe that $\|\boldsymbol{r}_t\| \leq \mathcal{B}\sqrt{md}$).
3. Pick $j \leftarrow \{1, \ldots, Q\}$.
4. Pick $\psi \leftarrow \Psi$.
5. Pick $(\phi_{j_-}, \boldsymbol{x}_j, \phi_{j_+}) \leftarrow \Phi_{j_-} \times \mathcal{C} \times \Phi_{j_+}$
6. Run 0: run $\mathcal{A}(\psi, \phi_{j_-}, \boldsymbol{x}_j, \phi_{j_+})$ with access to the oracles PKGen, Sign, Corrupt and the random oracle $H(\phi_{j_-}, \boldsymbol{x}_j, \phi_{j_+})$ simulated as follows. Whenever $\mathcal{A}$ queries PKGen, $\mathcal{D}$ answers as in the real case except for t-th query where pk_t is returned. If $\mathcal{A}$ ever queries Corrupt(t), $\mathcal{D}$ aborts (abort Type I). If $\mathcal{A}$ queries Sign$(t, \mathcal{M}, L)$, it picks a random challenge vector $\boldsymbol{x}$ and uses SHVZK simulator of Protocol 2 to simulate the proof $(\{CMT_i\}_{i=1}^r, \{RSP_i\}_{i=1}^r)$ (note that only the simulation of non-aborted protocols is used here). Then, the random oracle is programmed as $H(ck, \mathcal{M}, L, \{CMT_i\}_{i=1}^r) = \boldsymbol{x}$, except if $(ck, \mathcal{M}, L, \{CMT_i\}_{i=1}^r)$ has been queried before (abort Type II).
   (a) If $\mathcal{A}$ outputs a forgery σ^0 using j-th random oracle query output $\boldsymbol{x}_j^0$, fix ψ and ϕ_{j_-}.
   (b) Otherwise, abort.
7. Pick $\phi'_1, \ldots, \phi'_{\mathcal{N}} \leftarrow \Phi_{j_+}$
8. Run i (for $i \in \{1, \ldots, \mathcal{N}\}$ where $\mathcal{N}$ is defined below in the analysis): run $\mathcal{A}(\psi, \phi_{j_-}, \boldsymbol{x}_j^i, \phi'_i)$ with access to the oracles PKGen, Sign, Corrupt and the random oracle $H(\phi_{j_-}, \boldsymbol{x}_j^i, \phi'_i)$ where $\boldsymbol{x}_j^i$ is the response of the j-th random oracle query at iteration i.

(a) $\mathcal{A}$ outputs a forgery σ^i. We say that Run i is j-successful if σ^i was forged with respect to $\boldsymbol{x}_j^i$.

9. If there exists $i^* \in [1, r]$ and $S^* \subseteq \{0, \ldots, \mathcal{N}\}$ with $|S^*| = k + 1$ such that $\mathcal{G}^* := \{x_{j,i^*}^u \ : \ u \in S^*\}$ contains $k + 1$ distinct challenges and σ^u is j-successful for all $u \in S^*$, then run $(k + 1)$-special soundness extractor $\mathcal{E}$ of Protocol 2 on input $\{\sigma^u\}_{u \in S^*}$ to extract an opening of $2^k pk_{t'}$ to $(\mathbf{0}, \boldsymbol{s}_{t'})$ for some $1 \leq t' \leq q_P$ where $\|\boldsymbol{s}_{t'}\| \leq 24\sqrt{3r} \cdot m\mathcal{B} \cdot (k + 1) \cdot d^{k+1}$.
10. If $t = t'$, return $((2^k \cdot \mathbf{1}, 2^k \cdot \boldsymbol{r}_t), (\mathbf{0}, \boldsymbol{s}_{t'}))$ as a binding collision pair for the commitment scheme. Note that multiplication of $(\mathbf{1}, \boldsymbol{r}_t)$ by 2^k gives a valid opening of $2^k pk_t$, because $d^{k+1} > 2^k$ since $d \geq 7$.
11. Otherwise, abort.

Note that when $\mathcal{D}$ returns a binding collision, there cannot be Type I aborts as the forged signature must be for a ring comprised only of uncorrupted users.

Now, let us analyse this procedure in more details and denote $\varepsilon_{\text{LWE}} = O(2^{-\lambda})$ as the advantage of solving M-LWE problem. First, we observe that in each run of $\mathcal{A}$, the view of $\mathcal{A}$ is simulated by $\mathcal{D}$ with the same distribution as in the real attack except for:

- pk_t is a commitment to $\mathbf{1}$ in the simulation by $\mathcal{D}$ whereas it is a commitment to $\mathbf{0}$ in the real attack. By the hiding property of the commitment scheme, this reduces the success probability of $\mathcal{A}$ by at most ε_{LWE}.
- There is a statistical distance of at most $O(q_S \cdot 2^{-\lambda})$ between the distribution of signing oracle simulator and that of the real signing oracle.
- A Type II abort occurs during a signing oracle query with probability at most $Q \cdot 2^{-\lambda}$.

By the simulation statistical distance argument above, each run of $\mathcal{A}$ with pk_t and signing oracle simulated by $\mathcal{D}$ succeeds with probability $\tilde{\varepsilon} \geq \varepsilon - O(Q \cdot 2^{-\lambda})$. We say that $(\psi, \phi_{j_-}, \boldsymbol{x}_j, \phi_{j_+}, j)$ is 'winning' if $\mathcal{A}(\psi, \phi_{j_-}, \boldsymbol{x}_j, \phi_{j_+})$ outputs a valid forgery using $\boldsymbol{x}_j$ after Q random oracle queries. Note that there exists a $j^* \in \{1, \ldots, Q\}$ such that $\Pr[\ (\psi, \phi_{j_-^*}, \boldsymbol{x}_{j^*}, \phi_{j_+^*}, j^*) \text{ winning }] \geq \tilde{\varepsilon}/Q$. By the Splitting Lemma (Lemma 7 of [28]), there exists a subset $S \subseteq \Psi \times \Phi_{j_-^*}$ such that

$$\Pr_{\psi \in \Psi, \phi_{j_-^*} \in \Phi_{j_-^*}}[(\psi, \phi_{j_-^*}) \in S] \geq \tilde{\varepsilon}/(2Q), \text{ and}$$

$$\varepsilon' := \Pr_{\boldsymbol{x}_{j^*} \in \mathcal{C}, \phi_{j_+^*} \in \Phi_{j_+^*}}[(\psi, \phi_{j_-^*}, \boldsymbol{x}_{j^*}, \phi_{j_+^*}, j^*) \text{ winning }] \geq \tilde{\varepsilon}/(2Q) \qquad \forall (\psi, \phi_{j_-^*}) \in S.$$

Now, for $(\psi, \phi_{j_-^*}) \in S$, $c \in \mathcal{C}$ and $1 \leq i \leq r$, define $p_i(c)$ as the probability with respect to $\boldsymbol{x}_{j^*} \in \mathcal{C}$ and $\phi_{j_+^*} \in \Phi_{j_+^*}$ that $(\psi, \phi_{j_-^*}, \boldsymbol{x}_{j^*}, \phi_{j_+^*}, j^*)$ is winning and $\boldsymbol{x}_{j^*} = (x_{j^*,1}, \ldots, x_{j^*,r})$ with $x_{j^*,i} = c$.

Claim 1 *If $\varepsilon' > (k/|\mathcal{C}|)^r$, then there exists an $i^* \in [1, r]$ and $\mathcal{G} \subseteq \mathcal{C}$ with $|\mathcal{G}| = k + 1$ such that*

$$p_{i^*}(c) \geq \frac{\varepsilon' - (k/|\mathcal{C}|)^r}{(|\mathcal{C}| - k) \cdot r} =: p \qquad \forall c \in \mathcal{G}.$$

If the claim holds, then a sample of $\mathcal{N} := (k+1)\cdot p^{-1}$ independent and identically distributed winning tuples $(\psi, \phi_{j_-}, \boldsymbol{x}_j, \phi_{j_+}, j)$ will yield a set $\left\{\boldsymbol{x}_j^1, \ldots, \boldsymbol{x}_j^{k+1}\right\}$ such that $\mathcal{G} = \{x_{j,i^*}^1, \ldots, x_{j,i*}^{k+1}\}$ with a probability at least $1-(k+1)e^{-(k+1)}$, which is greater than $7/10$ for $k \geq 1$ (this comes from the fact that the probability that $\mathcal{N}$ samples do not contain c for some $c \in \mathcal{G}$ is at most $(k+1)\cdot(1-p)^{\mathcal{N}}$). That is, after $\mathcal{N}/\varepsilon'$ rewindings, we obtain a set of $(k+1)$ distinct challenge values of Protocol 2 with respect to the same initial commitment with a high probability.

Now, $\mathcal{N} = \text{poly}(\lambda)$ if $k, |\mathcal{C}|, r = \text{poly}(\lambda)$ and $\left(\varepsilon' - (k/|\mathcal{C}|)^r\right)^{-1} \leq \text{poly}(\lambda)$. It is easy to see that the first requirement holds since $|\mathcal{C}| = 2d$, $r = \frac{\lambda}{\log(2d)-\log k}$ and k is a small constant. For the second requirement, we have

$$\left(\varepsilon' - (k/|\mathcal{C}|)^r\right)^{-1} = (\varepsilon' - \eta)^{-1} \leq (\varepsilon' - \varepsilon'/2)^{-1} = 2/\varepsilon' \leq \text{poly}(\lambda),$$

where the first inequality holds since $\varepsilon' > 2\eta$. Now, by $(k+1)$-special soundness of Protocol 2, we can use the set $\mathcal{G}$ to extract an opening of $2^k pk_{t'}$ to $(\mathbf{0}, \boldsymbol{s}_{t'})$ for some $t' \in \{1, \ldots, q_P\}$. By the hiding property of the commitment scheme, $t' = t$ with probability at least $\frac{1}{q_P} - \varepsilon_{\text{LWE}}$. Also, $j = j^*$ with probability $\frac{1}{Q}$. Hence, $\mathcal{D}$ succeeds to output a binding collision pair with probability

$$\begin{aligned}\Pr[j = j^*] \cdot \Pr[(\psi, \phi_{j_-}) \in S] \cdot \Pr\left[\begin{matrix}\mathcal{N} \text{ runs contain } k+1 \\ j\text{-successful distinct challenges}\end{matrix}\right] \cdot \Pr[t = t'] \\ \geq \frac{1}{Q} \cdot \frac{\tilde{\varepsilon}}{2Q} \cdot \frac{7}{10} \cdot \left(\frac{1}{q_P} - \varepsilon_{\text{LWE}}\right) = \frac{1}{\text{poly}(\lambda)}.\end{aligned}$$

This leaves us with the proof of the claim, which is based on a pigeonhole argument. For each $i \in [1, r]$, let M_i with $|M_i| = k$ be the set of $c \in \mathcal{C}$ such that $p_i(c') \leq p_i(c)$ for all $c' \notin M_i$ and all $c \in M_i$. Further, let B be the set of $(\boldsymbol{x}_j, \phi_{j_+}) \in \mathcal{C}^r \times \boldsymbol{\Phi}_{j_+}$ for $\boldsymbol{x}_j = (x_{j,1}, \ldots, x_{j,r})$ such that $x_{j,i} \in M_i$ for all $i \in [1, r]$. Since $|M_i| = k$,

$$\Pr[(\boldsymbol{x}_j, \phi_{j_+}) \in B] \leq \Pr[x_{j,i} \in M_i \quad \forall i \in [1, r]] \leq (k/|\mathcal{C}|)^r .$$

For each $(\boldsymbol{x}_j, \phi_{j_+}) \in S \setminus B$, there exists $i \in [1, r]$ and $c \in \mathcal{C} \setminus M_i$ such that $x_{j,i} = c$. This implies that

$$\begin{aligned}\sum_{i=1}^{r} \sum_{c \in \mathcal{C} \setminus M_i} p_i(c) \geq \Pr[(\boldsymbol{x}_j, \phi_{j_+}) \in S \setminus B] \geq \Pr[(\boldsymbol{x}_j, \phi_{j_+}) \in S] - \Pr[(\boldsymbol{x}_j, \phi_{j_+}) \in B] \\ \geq \varepsilon' - (k/|\mathcal{C}|)^r .\end{aligned}$$

From here, we can deduce that there exists $i^* \in [1, r]$ and $c^* \in \mathcal{C} \setminus M_i$ such that $p_{i^*}(c^*) \geq \frac{\varepsilon' - (k/|\mathcal{C}|)^r}{(|\mathcal{C}|-k)\cdot r}$. Hence, for all $c \in \mathcal{G} := M_{i^*} \cup \{c^*\}$, $p_{i^*}(c) \geq \frac{\varepsilon' - (k/|\mathcal{C}|)^r}{(|\mathcal{C}|-k)\cdot r}$, proving the claim. □

Lemma 12. *The vector $\boldsymbol{g}$ defined in the verification of Protocol 1 satisfy the following $\|\boldsymbol{g}\|^2 \leq 5^4 \phi_1^4 d^3 k^3 \beta(\beta - 1)$.*

Proof. Since x is a monomial, we simply upper-bound $\|x - f_{j,i}\|$ by $\|f_{j,i}\|$ below.

$$\begin{aligned}
\|\boldsymbol{g}\|^2 &= \sum_{j=0}^{k-1}\sum_{i=0}^{\beta-1} \|f_{j,i}(x-f_{j,i})\|^2 \leq \sum_{j=0}^{k-1}\sum_{i=0}^{\beta-1} d\,\|f_{j,i}\|^2\,\|x-f_{j,i}\|^2 \\
&= \sum_{j=0}^{k-1}\sum_{i=1}^{\beta-1} d\,\|f_{j,i}\|^2\,\|x-f_{j,i}\|^2 + \sum_{j=0}^{k-1} d\,\|f_{j,0}\|^2\,\|x-f_{j,0}\|^2 \\
&\leq \sum_{j=0}^{k-1}\sum_{i=1}^{\beta-1} d\left(5\phi_1\sqrt{dk}\right)^2\left(5\phi_1\sqrt{dk}\right)^2 + \sum_{j=0}^{k-1} d\left(5\phi_1\sqrt{dk(\beta-1)}\right)^2\left(5\phi_1\sqrt{dk(\beta-1)}\right)^2 \\
&\leq dk(\beta-1)\left(5\phi_1\sqrt{dk}\right)^4 + dk\left(5\phi_1\sqrt{dk(\beta-1)}\right)^4 \\
&= 5^4\phi_1^4 d^3k^3(\beta-1) + 5^4\phi_1^4 d^3k^3(\beta-1)^2 \\
&= 5^4\phi_1^4 d^3k^3\beta(\beta-1)
\end{aligned}$$

□

Lemma 13. *The opening $(\hat{\boldsymbol{d}}, \hat{\boldsymbol{r}}_d)$ of $2D$ in the special soundness proof of Protocol 1 satisfy the following*

$$\left\|(\hat{\boldsymbol{d}}, \hat{\boldsymbol{r}}_d)\right\| \leq (2d+2)\left(5^4\phi_1^4 d^3k^3\beta(\beta-1) + 12\phi_2^2\mathcal{B}^2m^2d^2\right)^{1/2}.$$

Furthermore, the same bound applies to the openings of $2A, 2B$ and $2C$.

Proof. For distinct challenges x and x', recall the opening $(\hat{\boldsymbol{b}}, \hat{\boldsymbol{r}}_b)$ of $2B$ in the special soundness proof of Protocol 1. We have

$$\hat{\boldsymbol{b}} = 2(x-x')^{-1}(\boldsymbol{f}-\boldsymbol{f}') \quad \text{and} \quad \hat{\boldsymbol{r}}_b = 2(x-x')^{-1}(\boldsymbol{z}_b - \boldsymbol{z}'_b). \tag{15}$$

Similarly, recalling the opening $(\hat{\boldsymbol{a}}, \hat{\boldsymbol{r}}_a)$ of $2A$, we have

$$\hat{\boldsymbol{a}} = 2\boldsymbol{f} - x\hat{\boldsymbol{b}} \quad \text{and} \quad \hat{\boldsymbol{r}}_a = 2\boldsymbol{z}_b - x\hat{\boldsymbol{r}}_b. \tag{16}$$

Following the same procedure using the last verification step of Protocol 1, we can get the following openings $(\hat{\boldsymbol{c}}, \hat{\boldsymbol{r}}_c)$ and $(\hat{\boldsymbol{d}}, \hat{\boldsymbol{r}}_d)$ of $2C$ and $2D$, respectively,

$$(\hat{\boldsymbol{c}}, \hat{\boldsymbol{r}}_c) = (2(x-x')^{-1}(\boldsymbol{g}-\boldsymbol{g}'), 2(x-x')^{-1}(\boldsymbol{z}_c - \boldsymbol{z}'_c)), \tag{17}$$

$$(\hat{\boldsymbol{d}}, \hat{\boldsymbol{r}}_d) = (2\boldsymbol{g} - x\hat{\boldsymbol{c}}, 2\boldsymbol{z}_c - x\hat{\boldsymbol{r}}_c). \tag{18}$$

We bound the norm of $(\hat{\boldsymbol{d}}, \hat{\boldsymbol{r}}_d)$, which also involves bounding the norm of $(\hat{\boldsymbol{c}}, \hat{\boldsymbol{r}}_c)$. Without loss of generality, assume that $\|\boldsymbol{g}\| \geq \|\boldsymbol{g}'\|$ and $\|\boldsymbol{z}_c\| \geq \|\boldsymbol{z}'_c\|$. We use

the stronger bound from Protocol 2 to bound $\|\boldsymbol{z}_c\|$ below.

$$\begin{aligned}
\left\|(\hat{\boldsymbol{d}}, \hat{\boldsymbol{r}}_d)\right\| &= \|(2\boldsymbol{g} - x\hat{\boldsymbol{c}}, 2\boldsymbol{z}_c - x\hat{\boldsymbol{r}}_c)\| \leq \|(2\boldsymbol{g}, 2\boldsymbol{z}_c)\| + \|(x\hat{\boldsymbol{c}}, x\hat{\boldsymbol{r}}_c)\| \\
&\leq 2\,\|(\boldsymbol{g}, \boldsymbol{z}_c)\| + \|(\hat{\boldsymbol{c}}, \hat{\boldsymbol{r}}_c)\| \\
&= 2\,\|(\boldsymbol{g}, \boldsymbol{z}_c)\| + \left\|(2(x-x')^{-1}(\boldsymbol{g} - \boldsymbol{g}'), 2(x-x')^{-1}(\boldsymbol{z}_c - \boldsymbol{z}'_c)\right\| \\
&\leq 2\,\|(\boldsymbol{g}, \boldsymbol{z}_c)\| + \sqrt{d}\left\|2(x-x')^{-1}\right\|\,\|((\boldsymbol{g} - \boldsymbol{g}'), (\boldsymbol{z}_c - \boldsymbol{z}'_c))\| \\
&\leq 2\,\|(\boldsymbol{g}, \boldsymbol{z}_c)\| + 2d\,\|(\boldsymbol{g}, \boldsymbol{z}_c)\| \\
&\leq (2d+2)\left(5^4\phi_1^4 d^3 k^3 \beta(\beta-1) + \left(2\sqrt{3}\phi_2 \mathcal{B} m d\right)^2\right)^{1/2} \\
&= (2d+2)\left(5^4\phi_1^4 d^3 k^3 \beta(\beta-1) + 12\phi_2^2 \mathcal{B}^2 m^2 d^2\right)^{1/2}. \qquad (19)
\end{aligned}$$

The bounds on openings of $2A$ and $2B$ are clearly weaker as they only involve $f_{j,i}$'s whereas opening of $2D$ involves the products $f_{j,i}(x - f_{j,i})$'s as part of $\boldsymbol{g}$. □

Remark 3. When considering the r-repeated protocol, the bound in (19) becomes $(2d+2)\left(5^4\phi_1^4 d^3 k^3 \beta(\beta-1) r^2 + 12\phi_2^2 \mathcal{B}^2 m^2 d^2 r\right)^{1/2}$, and this bound is used when setting the parameters for the ring signature as discussed in Section 6.