PAPER

Tighter Reduction for Lattice-Based Multisignature*

Masayuki FUKUMITSU[†a)] ***and*** **Shingo HASEGAWA**[††b)]**,** ***Members***

SUMMARY Multisignatures enable multiple users to sign a message interactively. Many instantiations are proposed for multisignatures, however, most of them are quantum-insecure, because these are based on the integer factoring assumption or the discrete logarithm assumption. Although there exist some constructions based on the lattice problems, which are believed to be quantum-secure, their security reductions are loose. In this paper, we aim to improve the security reduction of lattice-based multisignature schemes concerning tightness. Our basic strategy is combining the multisignature scheme proposed by El Bansarkhani and Sturm with the lattice-based signature scheme by Abdalla, Fouque, Lyubashevsky, and Tibouchi which has a tight security reduction from the Ring-LWE (Ring Learning with Errors) assumption. Our result shows that proof techniques for standard signature schemes can be applied to multisignature schemes, then we can improve the polynomial loss factor concerning the Ring-LWE assumption. Our second result is to address the problem of security proofs of existing lattice-based multisignature schemes pointed out by Damgård, Orlandi, Takahashi, and Tibouchi. We employ a new cryptographic assumption called the Rejected-Ring-LWE assumption, to complete the security proof.

key words: *multisignature, tight security, lattice cryptography, Ring-LWE*

1. Introduction

The multisignature scheme [2] enables multiple signers to sign the same message with the interaction among the group of signers. The multisignature scheme helps the signers to reduce the size of signatures when they sign the same message, compared with the case when each signer uses a standard signature scheme independently. This feature is also suitable for devices with small computational power, such as the IoT devices. Moreover, the multisignature recently attracts attentions because it can handle transactions efficiently in the blockchain [3].

There are many instantiations of multisignatures. Most of them are built upon the RSA assumption or the discrete logarithm assumption, e.g. [4]–[10]. However, it is known that these two assumptions will be broken when quantum computers are practical so that they can handle many qubits [11]. Thus the multisignature schemes above are also vulnerable against quantum computers.

Lattice-based cryptography is one of the most promising candidates for post-quantum cryptography. There are many lattice-based cryptographic protocols for various cryptographic primitives. For the multisignature, there are few lattice-based instantiations such as [12]–[14]. El Bansarkhani and Sturm [12] proposed the first lattice-based multisignature scheme, whose security is proven in *the plain public-key model* [4], for a constant number of signers. Their scheme is based on the lattice-based Fiat-Shamir-type (FS-type) [15] signature scheme by Güneysu, Lyubashevsky and Pöppelmann [16].

The plain public-key model (PPK) is considered to be the strongest security model of the multisignature in the sense that each signer does not have to certify the validity of his public key. Namely, in the plain public-key model, the adversary can choose a public key arbitrarily as a member of the signer group to generate a forgery.

The security of the multisignature scheme by [12] is proven in the PPK model under the Ring-SIS (Ring Short Integer Solution) assumption, however, its security proof is given by a *loose* reduction. That is, the success probability ϵ_{RSIS} of breaking the Ring-SIS assumption decreases from the success probability ϵ of the attacker to the multisignature by a polynomial loss factor.

In modern cryptography, it is desired to prove the security of the cryptographic scheme by a *tight* reduction. The tight reduction means that the success probability of breaking the cryptographic scheme is the same as that of breaking the underlying cryptographic assumption except for constant factors. Conversely, in the loose reduction, the success probability of breaking the scheme is larger than that of breaking the assumption with polynomial factors. Therefore on the scheme whose security is proven by a loose security reduction, the security parameter of the scheme must be set to be large.

In the case of the multisignature, few instantiations have tight security reduction [5], [7], [10]. Furthermore, these are based on the computational Diffie-Hellman assumption, namely, they are not quantum-resistant. On the other hand, the known lattice-based constructions [12]–[14] have only a loose security reduction, even [13] and [14] appeared after the earlier version of this paper [1] was published. Thus, to the best of our knowledge, it is an open problem to construct a quantum-resistant multisignature whose security is proven by a tight security reduction.

Manuscript received November 12, 2020.
Manuscript revised March 30, 2021.
Manuscript publicized May 25, 2021.

[†]The author is with Faculty of Information Media, Hokkaido Information University, Ebetsu-shi, 069-8585 Japan.
[††]The author is with Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980-8576 Japan.

*The earlier version of this paper was presented at APKC '19 [1].

a) E-mail: fukumitsu@do-johodai.ac.jp
b) E-mail: shingo.hasegawa.b7@tohoku.ac.jp

DOI: 10.1587/transfun.2020EAP1131

1.1 Contribution

In this paper, we aim to construct a tighter security reduction for lattice-based multisignature schemes. Our basic strategy is combining the multisignature scheme of [12] with anothor lattice-based FS-type signature scheme by [17] which has a tight security reduction from the Ring-LWE (Ring Learning with Errors) assumption.

We consider the security of the proposed scheme in the PPK model and the classical random oracle model (ROM) from the Ring-LWE assumption. In the security proof, we employ the proof technique by Katz and Wang [18] which is called the "lossy key technique". By using the lossy key technique, they proposed the tightly-secure FS-type signature from the Decisional Diffie-Hellman assumption. A security proof with this technique proceeds as follows. The security reduction merely plays the euf-cma game as the challenger with the assumed attacker for a "lossy public key" instead of the standard public key. The lossy public key is computationally indistinguishable from the standard one under the designated cryptographic assumption, and there are no corresponding secret keys for such lossy public keys. In this case, one can evaluate that the success probability of the assumed attacker is at most negligible even if such an attacker runs in unbounded polynomial time. Abdalla, Fouque, Lyubashevsky, and Tibouchi [17] expanded the lossy key technique to the lattice-based FS-type signature. In their lattice-based signature, the computational indistinguishability between the lossy public keys and the standard public keys is guaranteed by the Ring-LWE assumption.

We extend their result to the case of multisignature schemes. In this sense, our result suggests that proof techniques for standard signature schemes is applicable to multisignature schemes. More precisely, the success probability ϵ_{RLWE} of breaking the Ring-LWE assumption appears in the success probability ϵ of attacking the scheme with no loss factor, whereas such a success probability by [12] has a polynomial loss factor. Namely, we can achieve the tightness concerning the underlying Ring-LWE assumption.

Although we can improve the polynomial loss factor concerning the Ring-LWE assumption in the security proof, a complete tight reduction is not achieved. This is because we require another cryptographic assumption to address the problem pointed out by [14], and a polynomial loss factor arises on the new cryptographic assumption. In [14], they found that all lattice-based multisignature schemes [1], [12], [13] using the rejection sampling [19], including our earlier version [1], have the common problem. On the lattice-based FS-type signature scheme using the rejection sampling such as [17], [19], the signer restarts the signing protocol when a signature does not pass the verification check. In that case, a commitment to which a part of a signature generated by FS-type signature scheme is reffered, and a hash value used to produce a signature are deleted and the attacker cannot know them. On the other hand, in the multisignature case, such commitments and hash values are shared among signers because they are required to compute a signature even if the resulting signature is rejected by the verification check eventually. In this case, there is no guarantee that these shared commitments do not leak the information concerning the secret key. To address this situation, in other words, to make these comments indistinguishable from random values, we require another assumption called the Rejected-Ring-LWE (Re-Ring-LWE) assumption. Since the rejection in the signing oracle simulation can occur polynomially many times, we suffer a polynomial loss factor concerning the Re-Ring-LWE assumption. Reducing the polynomial loss concerning the new assumption and then achieving the complete tight reduction is an important open question.

Note that the Re-Ring-LWE assumption is a generalization of the Rejected-DCK (Re-DCK) assumption [20]. As mentioned in [20], it remains open to show the reasonability of the Re-DCK assumption, although one expects that this assumption holds. Thus, investigating the reasonability of the Re-Ring-LWE assumption and the relationship between the assumption and the Ring-LWE assumption is also an important problem.

To address the issue concerning the Re-Ring-LWE assumption other than the thing above, we are likely to remove this assumption by setting long parameters so that such a rejection happens with a small probability, or by applying the technique [14], [20] to the proposed scheme in a way that such commitments of signatures are masked by a commitment scheme. In both cases, the size efficiency of the resulting multisignature scheme becomes worse than the original. Therefore, we consider that the size efficiency and the restart probability, in other words, employing the Re-Ring-LWE assumption is a tradeoff.

1.2 Comparison

We give a summary and a comparison of lattice-based multisignature schemes. The summary and the comparison are given in Tables 1, 2 and 3. All schemes listed in tables are proven to be secure in the PPK and RO model, however, their security reductions are loose.

The first lattice-based multisignature scheme is proposed by [12]. The scheme is based on the Ring-SIS assumption the Decisional Compact Knapsack (DCK) assumption and its security is proven in the PPK and RO model. The construction of [12] follows the strategy by [4], then the signing protocol requires 3-round communication among signers.

Ma and Jiang [13] proposed a variant of [12] in a sense that the expected number of the repetition of the signing protocol becomes almost 1 with the tradeoff to the communication complexity and the computation complexity. Due to the modification from [12], the multisignatue scheme of [13] needs one additional round in the signing protocol.

For the two multisignature schemes above, as discussed in the previous subsection, a problem concerning the repetition of the signing protocol is pointed out by [14], as well as our earlier version [1]. To address the problem, we em-

Table 1 Summary of lattice-based multisignature.

	Assumption	Distribution	Probability of Forger	Rounds	Security	Model
[12]	Ring-SIS, DCK, (Re-DCK)	Uniform/Uniform	$\sqrt{O(Q_T)\epsilon_{\mathrm{DCK}}} + e^{2N}\epsilon_{\mathrm{ReDCK}}$	3	PPK	ROM
[13]	Ring-SIS, DCK, (Re-DCK)	Uniform/Uniform	$\sqrt{O(Q_T)\epsilon_{\mathrm{DCK}}} + e^{2N}\epsilon_{\mathrm{ReDCK}}$	4	PPK	ROM
[14]	MSIS, MLWE	Unifrom/Discrete Gaussian	$O(Q_S)\left(\epsilon_{\mathrm{MLWE}} + \sqrt{Q_T(\epsilon_{\mathrm{MSIS}} + \epsilon_{\mathrm{RSIS}})}\right)$	2	PPK	ROM
[ours]	Ring-LWE, Re-Ring-LWE	Unifrom/Unifrom	$\epsilon_{\mathrm{RLWE}} + e^{2N}\epsilon_{\mathrm{ReRLWE}}$	3	PPK	ROM

The column Assumption means the security assumptions required in each scheme. The column Distribution means the distribution of the secret key and that of the randomness used in the signature generation. The column Probability of Forger denotes the upper bound of the probability of attacking the target multisignature. The column Rounds means the number of interactions among signers in the signing protocol. All schemes are proven to be secure for the plain public-key model in the random oracle model. Q_S denotes the number of making queries to the signing oracle, and Q_T does the total number of the signing oracle queries and the random oracle queries. N is the number of group members issuing a signature. Note that N is considered as a constant in these lattice-based schemes. ϵ_{DCK}, $\epsilon_{\mathrm{ReDCK}}$, ϵ_{MLWE}, ϵ_{MSIS} ϵ_{RSIS}, ϵ_{RLWE}, $\epsilon_{\mathrm{ReRLWE}}$ are the probabilities of breaking the DCK assumption, the Re-DCK assumption, the Module-LWE assumption, the Module-SIS assumption, the Ring-SIS assumption, the Ring-LWE assumption and the Re-Ring-LWE assumption, respectively.

Table 2 Comparison of the component size of communication size.

	Component Size				Communication Size			
	pp	**pk**	**sk**	σ	1st	2nd	3rd	4th
[12]	$\lvert R\rvert$	$\lvert R\rvert$	$2\lvert R\rvert$	$3\lvert R\rvert$	$\lvert R\rvert$	$\lvert R\rvert$	$2\lvert R\rvert$	-
[13]	$\lvert R\rvert$	$\lvert R\rvert$	$2\lvert R\rvert$	$(N+2)\lvert R\rvert$	$O(N)\lvert R\rvert$	$O(N)\lvert R\rvert$	$2^{O(N)}\lvert R\rvert$	$3\lvert R\rvert$
[14]	$\lvert M_{k\times l}\rvert$ $+\lvert M_{2\times(l+2w)}\rvert$	$\lvert M_k\rvert$	$\lvert M_{k+l}\rvert$	$\lvert M_{k\times 2}\rvert + \lvert M_{k+l}\rvert$ $+\lvert M_{k\times(l+2w)}\rvert$	$\lvert M_{k\times 2}\rvert$	$\lvert M_{k+l}\rvert$ $+\lvert M_{k\times(l+2w)}\rvert$	-	-
[ours]	$\lvert R\rvert$	$\lvert R\rvert$	$2\lvert R\rvert$	$3\lvert R\rvert$	$\lvert R\rvert$	$\lvert R\rvert$	$2\lvert R\rvert$	-

pp, **pk**, **sk** and σ denote the public parameter, the public key, the secret key and the signature, respectively. R means the underlying ring of the scheme, while for any natural numbers r, c, and d, $M_{r\times c}$ and M_d mean the $(r\times c)$-matrix and d-vector of underlying module. $\lvert R\rvert$, $\lvert M_{r\times c}\rvert$, and $\lvert M_d\rvert$ means the size of one element of R, $M_{r\times c}$, and M_d, respectively. The parameters k, l are the numbers of rows and columns of a basic matrix for [14], and w is the parameter for the employed commitment scheme.

Table 3 Comparison of computation time.

	Computation Time		
	Sign		Ver
	Signing time per signer	Expected number of repetition	
[12]	2 Mul	e^{2N}	$(N+1)$ Mul
[13]	$2^{O(N)}$ Mul	1	$(N+1)$ Mul
[14]	$O(Nkl)$ Mul	$e^{o(\log(n(k+l)))/\alpha(n,k,l,s)+1/(2\alpha(n,k,l,s))^2}$	$(N+O(k(l+w)))$ Mul
[ours]	2 Mul	e^{2N}	$(N+1)$ Mul

Mul means the computation time of one multiplication in the underlying ring R. e is Napier's constant. n is the degree of the underlying module. $\alpha(n,k,l,s)$ is a rational number depending on the degree n, the numbers of rows k and columns l, and the designated standard deviation s concerned in the discrete Gaussian distribution.

ploy the Re-Ring-LWE assumption in this paper. Similarly, it seems to require the Re-DCK assumption [20], to resolve the problem on these multisignature schemes, although this point is not referred to in the original papers.

Damgård, Orlandi, Takahashi, and Tibouchi [14] proposed a 2-round lattice-based multisignature scheme. Their scheme employs a homomorphic commitment to achieve a 2-round signing protocol besides overcoming the problem discussed above. The security is proven from the Module-SIS assumption and the Module-LWE assumption. For the evaluation in Tables 1, 2 and 3, we consider the instantiation of a homomorphic commitment proposed also in [14].

We note that a lattice-based non-interactive multisignature scheme is proposed in [21]. However, the vulnerability on that scheme is found in [22]. They showed that a secret key of [21] can be recovered from sufficient but realistic numbers of signatures.

It also should be noted that the number of users is required to be constant in all of the lattice-based multisignature schemes mentioned in this subsection. This reason comes from the rejection sampling technique used in the signing algorithm of every lattice-based multisignature scheme. Nevertheless, these signature schemes are sufficient to be used in some applications such as the blockchain system. Indeed, El Bansarkhani and Sturm [12] discussed the parameter of their multisignature scheme when the number of users is 5 and 10. It also remains open whether or not lattice-based multisignature schemes with polynomially many signers can be constructed.

1.3 Related Works

There are ID-based variants of lattice-based multisignature schemes. In [23], a lattice-based ID-based multi-proxy multisignature scheme was introduced. Their scheme relies on the standard LWE assumption.

[24] proposed a lattice-based ID-based blind multisignature scheme. Its security is proven by the SIS assumption.

1.4 Differences from Proceedings Version

The earlier version of this paper appeared in [1]. For the multisignature scheme in the proceeding version, [14] pointed out that the security proof has a flaw concerning the signing protocol. We introduce the Re-Ring-LWE assumption and apply the technique of [20] to complete the security proof correctly.

2. Preliminaries

Let $\mathbb{N}$ be the set of natural numbers, and let $\mathbb{Z}$ be the set of integers, respectively. For any integers $a \leq b$, $[a,b] \subseteq \mathbb{Z}$ denotes the set of integers from a to b. For a finite set X, $x \in_{\mathrm{U}} X$ means that an element x is chosen from X uniformly at random. The cardinality of X is denoted by $|X|$. We say that a function ϵ is negligible in λ if for any polynomial μ, there exists a natural number λ_0 such that $\epsilon(\lambda) < 1/\mu(\lambda)$ for $\lambda > \lambda_0$.

2.1 Multisignature

We introduce the definitions concerning the multisignature.

Definition 1. A multisignature scheme MSig consists of the 4-tuple $(\mathsf{Setup}, \mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$. Setup and KGen are probabilistic polynomial-time algorithms. The signing process can be done by executing the designated multiparty protocol among multiple users. Each user runs an interactive polynomial-time algorithm Sig. Ver is a deterministic polynomial-time algorithm. The description of these four algorithms is as follows:

$\mathsf{Setup}(1^\lambda)$ outputs a public parameter $\mathbf{pp}$ on a security parameter λ.

$\mathsf{KGen}(\mathbf{pp})$ outputs a pair $(\mathbf{sk}, \mathbf{pk})$ of a secret key $\mathbf{sk}$ and a public key $\mathbf{pk}$ on $\mathbf{pp}$.

$\mathsf{Sig}(\mathbf{pp}, \mathbf{sk}, \mathsf{PK}, \mu)$ outputs a signature σ on a message μ and a set PK of public keys.

$\mathsf{Ver}(\mathbf{pp}, \mathsf{PK}, \mu, \sigma)$ outputs 1 if σ is a valid signature for (PK, μ), or 0 otherwise.

(1) Completeness

Let $\mathsf{MSig} = (\mathsf{Setup}, \mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$ be a multisignature. We say that MSig satisfies the completeness if for any security parameter λ, any public parameter $\mathbf{pp} \leftarrow \mathsf{Setup}(1^\lambda)$ and any message μ, $\mathsf{Ver}(\mathbf{pp}, \mathsf{PK}, \mu, \sigma) = 1$ always holds for any signature σ which is computed by all users i ($i \in [1, N]$) executing $\mathsf{Sig}(\mathbf{pp}, \mathbf{sk}_i, \mathsf{PK}, \mu)$, where $(\mathbf{sk}_i, \mathbf{pk}_i) \leftarrow \mathsf{KGen}(\mathbf{pp})$ and $\mathsf{PK} = \{\mathbf{pk}_i\}_{i=1}^N$.

(2) Security

We introduce the security notion for multisignatures in the plain public key model [4]. The security game, called the *plain public key (ppk) game*, is defined between the challenger $\mathcal{C}$ and the forger $\mathcal{F}$. The description of the game is as follows:

Setup The challenger $\mathcal{C}$ generates $\mathbf{pp} \leftarrow \mathsf{Setup}(1^\lambda)$ and the challenge key pair $(\mathbf{pk}^*, \mathbf{sk}^*) \leftarrow \mathsf{KGen}(\mathbf{pp})$. Then $\mathcal{C}$ sends $(\mathbf{pp}, \mathbf{pk}^*)$ to the forger $\mathcal{F}$.

Sign For a t-th query $(\mu^{(t)}, \mathsf{PK}^{(t)})$ of a message $\mu^{(t)}$ and a public key set $\mathsf{PK}^{(t)}$ including $\mathbf{pk}^*$, $\mathcal{C}$ computes the corresponding signature $\sigma^{(t)}$, in an interactive manner with $\mathcal{F}$ that plays the role of users having all public keys $\mathbf{pk} \in \mathsf{PK}^{(t)} \setminus \{\mathbf{pk}^*\}$. Then $\mathcal{C}$ answers $\sigma^{(t)}$ to $\mathcal{F}$. Note that $\mathcal{F}$ can make each query in concurrent manner. Namely, $\mathcal{F}$ can start to make a new query before previous queries are answered.

Challenge $\mathcal{F}$ *wins the ppk game* if the output $(\mathsf{PK}^*, \mu^*, \sigma^*)$ of $\mathcal{F}$ satisfies the following three conditions:

1. $\mathbf{pk}^* \in \mathsf{PK}^*$.
2. (μ^*, PK^*) is not queried in **Sign** phase.
3. $\mathsf{Ver}(\mathbf{pp}, \mathsf{PK}^*, \mu^*, \sigma^*) = 1$.

As in [4], [12], we suppose that $\mathcal{F}$ makes queries $(\mu^{(t)}, \mathsf{PK}^{(t)})$ such that $\mathsf{PK}^{(t)}$ contains the challenge public key $\mathbf{pk}^*$ only once in **Sign** phase, and that the first element in $\mathsf{PK}^{(t)}$ is it without loss of generality. On the other hand, $\mathcal{F}$ can finally outputs PK^* that contains $\mathbf{pk}^*$ multiple times in **Challenge** phase.

Definition 2. Let T be a polynomial and ϵ be a function. A multisignature scheme MSig with N users ($N \in \mathbb{N}$) is (T, N, ϵ, Q_s)-*ppk secure* if for any forger $\mathcal{F}$ running in at most time T which makes at most Q_s queries in **Sign** phase, the probability that $\mathcal{F}$ wins the ppk game is ϵ.

Especially, in the random oracle model, MSig is said to be $(T, N, \epsilon, Q_s, Q_1, Q_2, \ldots)$-*ppk secure* if MSig is (T, N, ϵ, Q_s)-ppk secure and the number of queries to the random oracle H_i is at most Q_i for all i.

2.2 Lattice

For positive integers q and n, R and R_q are defined by $R = \mathbb{Z}[X]/(X^n + 1)$ and $R_q = \mathbb{Z}_q[X]/(X^n + 1)$, respectively. Any element $\boldsymbol{w} = \sum_{i=0}^{n-1} w_i X^i$ in R or R_q can be expressed by its coefficient vector $(w_0, \ldots, w_{n-1})$. Especially, we assume that coefficients of elements in R_q are in the range $[-(q-1)/2, (q-1)/2]$. Let $R_q^\times$ be the set of invertible elements in R_q.

For $\boldsymbol{w} = (w_0, \ldots, w_{n-1}) \in R$, the ℓ_∞-norm of $\boldsymbol{w}$ is defined by $\|\boldsymbol{w}\|_\infty = \max_{0 \leq i \leq n-1} |w_i|$. For any real number β, $R^{\leq \beta}$ means that the set of all elements in R such that their ℓ_∞-norm is at most β. We have that $\left|R^{\leq \beta}\right|$ is $(2\beta + 1)^n$. In a similar manner, $R_q^{\leq \beta}$ is also defined with $\beta \leq q/2$.

Let β_s be a natural number. For an algorithm $\mathcal{A}$, we consider the following two probabilities:

$$P_0^{\mathrm{RLWE}} = \Pr\left[\mathcal{A}(\boldsymbol{a}, \boldsymbol{t}) = 1 : \boldsymbol{a}, \boldsymbol{t} \in_{\mathrm{U}} R_q\right],$$

$$P_1^{\mathrm{RLWE}} = \Pr\left[\mathcal{A}(\boldsymbol{a}, \boldsymbol{a}\boldsymbol{s}_1 + \boldsymbol{s}_2) = 1 : \boldsymbol{a} \in_{\mathrm{U}} R_q, \boldsymbol{s}_1, \boldsymbol{s}_2 \in_{\mathrm{U}} R_q^{\leq \beta_s}\right].$$

Then, the advantage $\mathsf{Adv}^{\mathrm{RLWE}}_{\mathcal{A},\beta_s}$ of the *Ring Learning with Error* (Ring-LWE) problem is defined as follows:

$$\mathsf{Adv}^{\mathrm{RLWE}}_{\mathcal{A},\beta_s} = \left|P^{\mathrm{RLWE}}_0 - P^{\mathrm{RLWE}}_1\right|.$$

The Ring-LWE assumption is defined as follows.

Definition 3. Let T_{RLWE} be a polynomial and ϵ_{RLWE} be a function. The $(\beta_s, T_{\mathrm{RLWE}}, \epsilon_{\mathrm{RLWE}})$-Ring-LWE assumption holds if for any algorithm $\mathcal{A}$ running in at most time T_{RLWE}, $\mathsf{Adv}^{\mathrm{RLWE}}_{\mathcal{A},\beta_s} \leq \epsilon_{\mathrm{RLWE}}$.

Following [17], we consider that the modulo q is prime and polynomial-length in n such that $q \equiv 1 \pmod{2n}$. We also consider that the element $\boldsymbol{a}$ on the both two assumptions above is chosen uniformly at random from the set $R_q^\times$ instead of R_q. This restriction does not affect the hardness of the assumption [17]. Note that we consider the uniform distribution over $R_q^{\leq\beta_s}$ as the distribution on $\boldsymbol{s}_1, \boldsymbol{s}_2$ like the Module-LWE assumption [25].

We consider another assumption concerning the Ring-LWE assumption, which is an analog of the Rejected-DCK assumption [20] to the Ring-LWE case. Let β_y, β_c and β_z be natural numbers. For an algorithm $\mathcal{A}$, we consider the following two probabilities:

$$P^{\mathrm{ReRLWE}}_0 = \Pr\left[\begin{array}{l|l} \mathcal{A}(\boldsymbol{a},\boldsymbol{u},\boldsymbol{c}) = 1 : & \boldsymbol{a} \in_{\mathrm{U}} R_q, \boldsymbol{s}_1, \boldsymbol{s}_2 \in_{\mathrm{U}} R_q^{\leq\beta_s}, \\ & \boldsymbol{y}_1, \boldsymbol{y}_2 \in_{\mathrm{U}} R_q^{\leq\beta_y}, \boldsymbol{c} \in_{\mathrm{U}} R_q^{\leq\beta_c}, \\ & \boldsymbol{u} \in_{\mathrm{U}} R_q \\ \multicolumn{2}{l}{\quad \boldsymbol{s}_1\boldsymbol{c} + \boldsymbol{y}_1 \notin R_q^{\leq\beta_z}, \text{ or } \boldsymbol{s}_2\boldsymbol{c} + \boldsymbol{y}_2 \notin R_q^{\leq\beta_z}} \end{array}\right],$$

$$P^{\mathrm{ReRLWE}}_1 = \Pr\left[\begin{array}{l|l} \mathcal{A}(\boldsymbol{a},\boldsymbol{u},\boldsymbol{c}) = 1 : & \boldsymbol{a} \in_{\mathrm{U}} R_q, \boldsymbol{s}_1, \boldsymbol{s}_2 \in_{\mathrm{U}} R_q^{\leq\beta_s}, \\ & \boldsymbol{y}_1, \boldsymbol{y}_2 \in_{\mathrm{U}} R_q^{\leq\beta_y}, \boldsymbol{c} \in_{\mathrm{U}} R_q^{\leq\beta_c}, \\ & \boldsymbol{u} = \boldsymbol{a}\boldsymbol{y}_1 + \boldsymbol{y}_2 \\ \multicolumn{2}{l}{\quad \boldsymbol{s}_1\boldsymbol{c} + \boldsymbol{y}_1 \notin R_q^{\leq\beta_z}, \text{ or } \boldsymbol{s}_2\boldsymbol{c} + \boldsymbol{y}_2 \notin R_q^{\leq\beta_z}} \end{array}\right].$$

The advantage $\mathsf{Adv}^{\mathrm{ReRLWE}}_{\mathcal{A},\beta_s,\beta_c,\beta_y,\beta_z}$ of the *Rejected-Ring Learning with Error* (Re-Ring-LWE) problem is defined as follows:

$$\mathsf{Adv}^{\mathrm{ReRLWE}}_{\mathcal{A},\beta_s,\beta_c,\beta_y,\beta_z} = \left|P^{\mathrm{ReRLWE}}_0 - P^{\mathrm{ReRLWE}}_1\right|.$$

Definition 4. Let T_{ReRLWE} be a polynomial and $\epsilon_{\mathrm{ReRLWE}}$ be a function. The $(\beta_s, \beta_c, \beta_y, \beta_z, T_{\mathrm{RLWE}}, \epsilon_{\mathrm{ReRLWE}})$-Re-Ring-LWE assumption holds if for any algorithm $\mathcal{A}$ running in at most time T_{ReRLWE}, $\mathsf{Adv}^{\mathrm{ReRLWE}}_{\mathcal{A},\beta_s,\beta_c,\beta_y,\beta_z} \leq \epsilon_{\mathrm{ReRLWE}}$.

Note that the Re-Ring-LWE assumption coincides with the Rejected-DCK assumption when $\beta_s = 1$. Namely the Re-Ring-LWE assumption can be considered as a generalization of the Rejected-DCK assumption.

3. Proposed Lattice-Based Multisignature Scheme

In this section, we introduce a new multisignature scheme

Table 4 Parameters for MSig.

	Definition	Setting
n	dimension	a power of 2
q	modulo	small prime $\equiv 1 \bmod 2n$
N	# of signer	independent of n e.g. 5, 10 [12]
γ		independent of n, $\gamma \geq 1$
β_s	bound of ℓ_∞-norm of $\boldsymbol{s}$	ratio between q and β_s is poly(n)
β_y	bound of ℓ_∞-norm of $\boldsymbol{y}$	$\gamma n^2 \beta_s \log n$
β_c	bound of ℓ_∞-norm of $\boldsymbol{c}$	$\log n$
β_z	bound of ℓ_∞-norm of $\boldsymbol{z}$	$(\gamma n - 1) n \beta_s \log n$
ℓ_H	length of hash value of H_0	poly(n)

$\boldsymbol{s}$, $\boldsymbol{y}$, $\boldsymbol{c}$ and $\boldsymbol{z}$ are parts of a secret key and a multisignature explained just latter. The settings except that of the parameter N are considered in a similar manner to [17], whereas that of N is given as in [12].

MSig. Here, we use parameters n, q, N, γ, β_s, β_y, β_c, β_z and ℓ_H which are listed in Table 4.

3.1 Construction

We now introduce MSig. Let $H_0 : \{0,1\}^* \to \{0,1\}^{\ell_H}$ and $H_1 : \{0,1\}^* \to R_q^{\leq\beta_c}$ be hash functions, respectively. The description $\mathsf{MSig} = (\mathsf{Setup}, \mathsf{KGen}, \mathsf{Sig}, \mathsf{Ver})$ is as follows:

$\mathsf{Setup}(1^\lambda)$ outputs $\mathbf{pp} \in_{\mathrm{U}} R_q^\times$.

$\mathsf{KGen}(\mathbf{pp})$ outputs a key pair $(\mathbf{sk}, \mathbf{pk})$ computed as follows:

(1) $\mathbf{sk} = (\boldsymbol{s}_1, \boldsymbol{s}_2)$, where $\boldsymbol{s}_1, \boldsymbol{s}_2 \in_{\mathrm{U}} R_q^{\leq\beta_s}$, and

(2) $\mathbf{pk} = \mathbf{pp} \cdot \boldsymbol{s}_1 + \boldsymbol{s}_2$.

$\mathsf{Sig}(\mathbf{pp}, \mathbf{sk}_i, \mathsf{PK}, \mu)$ outputs a signature σ on a message μ and a set of public keys $\mathsf{PK} = \{\mathbf{pk}_j\}_{j=1}^N$ according to the protocol below. Now, $\mathbf{sk}_i = (\boldsymbol{s}_{i,1}, \boldsymbol{s}_{i,2})$. All users interactively run Sig. The protocol halts without output if the number of the restart is beyond the expected iteration time $E_{\gamma,n}$, which will be evaluated at Sect. 3.3. Each user having $(\mathbf{sk}_i, \mathbf{pk}_i)$ executes the protocol as follows:

(i) computes $\boldsymbol{u}_i = \mathbf{pp} \cdot \boldsymbol{y}_{i,1} + \boldsymbol{y}_{i,2}$, where $\boldsymbol{y}_{i,1}, \boldsymbol{y}_{i,2} \in_{\mathrm{U}} R_q^{\leq\beta_y}$.

(ii) computes $\boldsymbol{h}_i = H_0(\boldsymbol{u}_i)$ and broadcasts $\boldsymbol{h}_i$ to the other users.

(iii) receives $\{\boldsymbol{h}_j\}_{j\in[1,N]\setminus\{i\}}$ from the other users, then broadcasts $\boldsymbol{u}_i$ to the other users.

(iv) receives $\{\boldsymbol{u}_j\}_{j\in[1,N]\setminus\{i\}}$ from the other users and checks $\boldsymbol{h}_j = H_0(\boldsymbol{u}_j)$ for all $j \in [1,N]\setminus\{i\}$. If there exists an index j such that $\boldsymbol{h}_j \neq H_0(\boldsymbol{u}_j)$, aborts the protocol.

(v) computes $\boldsymbol{u} = \sum_{j=1}^N \boldsymbol{u}_j$.

(vi) computes $\boldsymbol{c}_i = H_1(\mathbf{pk}_i, \boldsymbol{u}, \mathsf{PK}, \mu)$.

(vii) sets $(\boldsymbol{z}_{i,1}, \boldsymbol{z}_{i,2}) = (\boldsymbol{s}_{i,1}\boldsymbol{c}_i + \boldsymbol{y}_{i,1}, \boldsymbol{s}_{i,2}\boldsymbol{c}_i + \boldsymbol{y}_{i,2})$.

(viii) If $(\boldsymbol{z}_{i,1}, \boldsymbol{z}_{i,2}) \notin \left(R_q^{\leq\beta_z}\right)^2$, restarts the protocol from (i).

(ix) broadcasts $(z_{i,1}, z_{i,2})$ to the other users.

(x) receives $\{(z_{j,1}, z_{j,2})\}_{j\in[1,N]\setminus\{i\}}$ from the other users, then computes $(z_1, z_2) = \sum_{j=1}^{N}(z_{j,1}, z_{j,2})$.

(xi) outputs $\sigma = (\boldsymbol{u}, z_1, z_2)$.

$\mathsf{Ver}(\mathbf{pp}, \mathsf{PK}, \mu, \sigma)$ outputs 1 if σ satisfies the following conditions:

(i) $z_1, z_2 \in R_q^{\leq N\beta_z}$.

(ii) $\mathbf{pp}\cdot z_1 + z_2 = \boldsymbol{u} + \sum_{j=1}^{N}\mathbf{pk}_j \cdot \boldsymbol{c}_j$, where $\mathsf{PK} = \{\mathbf{pk}_j\}_{j=1}^{N}$ and $\boldsymbol{c}_j = H_1(\mathbf{pk}_j, \boldsymbol{u}, \mathsf{PK}, \mu)$ for each $j \in [1, N]$.

3.2 Completeness

We show that $\mathsf{Ver}(\mathbf{pp}, \mathsf{PK}, \mu, \sigma) = 1$ always holds for any $\mathbf{pp} \leftarrow \mathsf{Setup}(1^\lambda)$, any $i \in [1, N]$, any $(\mathbf{sk}_i, \mathbf{pk}_i) \leftarrow \mathsf{KGen}(\mathbf{pp})$ and any $\sigma \leftarrow \mathsf{Sig}(\mathbf{pp}, \mathbf{sk}_i, \mathsf{PK}, \mu)$ where $\mathsf{PK} = \{\mathbf{pk}_i\}_{i=1}^{N}$. In this case, it is guaranteed that $z_{i,1}, z_{i,2} \in R_q^{\leq\beta_z}$ in (viii) of Sig. Therefore, we have

$$\begin{aligned}\|z_1\|_\infty &= \left\|\sum_{i=1}^{N} z_{i,1}\right\|_\infty \leq \sum_{i=1}^{N}\left\|z_{i,1}\right\|_\infty \leq N\beta_z, \text{ and}\\ \|z_2\|_\infty &= \left\|\sum_{i=1}^{N} z_{i,2}\right\|_\infty \leq \sum_{i=1}^{N}\left\|z_{i,2}\right\|_\infty \leq N\beta_z,\end{aligned} \tag{1}$$

for the condition (i) of Ver. Moreover, it follows from (vii) and (x) of Sig that

$$\begin{aligned}\mathbf{pp}\cdot z_1 + z_2 &= \sum_{j=1}^{N}\left(\mathbf{pp}\cdot z_{j,1} + z_{j,2}\right)\\ &= \sum_{j=1}^{N}\left(\mathbf{pk}_j\cdot\boldsymbol{c}_j + \boldsymbol{u}_j\right)\\ &= \sum_{j=1}^{N}\mathbf{pk}_j\cdot\boldsymbol{c}_j + \sum_{j=1}^{N}\boldsymbol{u}_j\\ &= \sum_{j=1}^{N}\mathbf{pk}_j\cdot\boldsymbol{c}_j + \boldsymbol{u}.\end{aligned} \tag{2}$$

Thus, the condition (ii) of Ver holds.

3.3 Restart Probability and Expected Iteration Time

We evaluate the probability that the signature generation protocol Sig restarts. We first prove the following lemmas.

Lemma 1. Let $n, q, N, \gamma, \beta_s, \beta_y, \beta_c$ and β_z be the parameters set as in Table 4. For any $\boldsymbol{s} \in R_q^{\leq\beta_s}$ and any $\boldsymbol{c} \in R_q^{\leq\beta_c}$, we have the followings:

- $\Pr_{\boldsymbol{y}\in_U R_q^{\leq\beta_y}}\left[\boldsymbol{sc} + \boldsymbol{y} \in R_q^{\leq\beta_z}\right] = \dfrac{\left|R_q^{\leq\beta_z}\right|}{\left|R_q^{\leq\beta_y}\right|}$, and (3)
- for any $\boldsymbol{z} \in R_q^{\leq\beta_z}$,

$$\Pr_{\boldsymbol{y}\in_U R_q^{\leq\beta_y}}\left[\boldsymbol{sc} + \boldsymbol{y} = \boldsymbol{z} \mid \boldsymbol{sc} + \boldsymbol{y} \in R_q^{\leq\beta_z}\right] = \frac{1}{\left|R_q^{\leq\beta_z}\right|}. \tag{4}$$

Proof. We fix $\boldsymbol{z} \in R_q^{\leq\beta_z}$.

(3) In the similar way to the proof of Lemma 2 in [17], by letting $\boldsymbol{c} = (c_0, \ldots, c_{n-1})$ and $\boldsymbol{s} = (s_0, \ldots, s_{n-1})$, the k-th coefficient of $\boldsymbol{cs} \bmod X^n + 1$ can be expressed in the following way:

$$\begin{cases} c_0 s_0 - \sum_{i=1}^{n-1} c_i s_{n-i}, & k = 0,\\ \sum_{i=0}^{k} c_i s_{k-i} - \sum_{i=k+1}^{n-1} c_i s_{n+k-i}, & 1 \leq k \leq n-2,\\ \sum_{i=0}^{n-1} c_i s_{n-1-i}, & k = n-1.\end{cases}$$

On the other hands, we have $\left|c_i s_j\right| \leq \beta_c\beta_s$ for any $i, j \in [0, n-1]$, since $\|\boldsymbol{c}\|_\infty \leq \beta_c$ and $\|\boldsymbol{s}\|_\infty \leq \beta_s$ These imply that $\|\boldsymbol{sc}\|_\infty \leq n\beta_c\beta_s \leq n\beta_s \log n$. It follows from Table 4 that

$$\begin{aligned}\|\boldsymbol{z} - \boldsymbol{sc}\|_\infty &\leq (\gamma n - 1)n\beta_s \log n + n\beta_s \log n\\ &= \gamma n^2\beta_s \log n\\ &= \beta_y.\end{aligned}$$

Hence, it holds that $\boldsymbol{z} - \boldsymbol{sc} \in R_q^{\leq\beta_y}$. Then, we have

$$\begin{aligned}\Pr_{\boldsymbol{y}\in_U R_q^{\leq\beta_y}}[\boldsymbol{y} + \boldsymbol{cs} = \boldsymbol{z}] &= \Pr_{\boldsymbol{y}\in_U R_q^{\leq\beta_y}}[\boldsymbol{y} = \boldsymbol{z} - \boldsymbol{cs}]\\ &= \frac{1}{\left|R_q^{\leq\beta_y}\right|}.\end{aligned}$$

We can evaluate the probability as

$$\Pr_{\boldsymbol{y}\in_U R_q^{\leq\beta_y}}\left[\boldsymbol{y} + \boldsymbol{cs} \in R_q^{\leq\beta_z}\right] = \frac{\left|R_q^{\leq\beta_z}\right|}{\left|R_q^{\leq\beta_y}\right|}.$$

(4) It follows from the above discussion that

$$\begin{aligned}&\Pr_{\boldsymbol{y}\in_U R_q^{\leq\beta_y}}\left[\boldsymbol{y} + \boldsymbol{cs} = \boldsymbol{z} \mid \boldsymbol{y} + \boldsymbol{cs} \in R_q^{\leq\beta_z}\right]\\ &= \frac{\Pr_{\boldsymbol{y}\in_U R_q^{\leq\beta_y}}\left[\boldsymbol{y} + \boldsymbol{cs} = \boldsymbol{z} \wedge \boldsymbol{y} + \boldsymbol{cs} \in R_q^{\leq\beta_z}\right]}{\Pr_{\boldsymbol{y}\in_U R_q^{\leq\beta_y}}\left[\boldsymbol{y} + \boldsymbol{cs} \in R_q^{\leq\beta_z}\right]}\\ &= \frac{1}{\left|R_q^{\leq\beta_z}\right|}.\end{aligned}$$

□

Lemma 2. Let $\gamma \geq 1$. For $z_{i,1} = \boldsymbol{s}_{i,1}\boldsymbol{c}_i + \boldsymbol{y}_{i,1}$ and $z_{i,2} = \boldsymbol{s}_{i,2}\boldsymbol{c}_i + \boldsymbol{y}_{i,2}$, the non-abort probability in (viii) of Sig is evaluated by

$$\Pr\left[z_{i,1}, z_{i,2} \in R_q^{\leq \beta_z}\right] = \left(\frac{\left|R_q^{\leq \beta_z}\right|}{\left|R_q^{\leq \beta_y}\right|}\right)^2. \tag{5}$$

Moreover, it can be evaluated as

$$\Pr\left[z_{i,1}, z_{i,2} \in R_q^{\leq \beta_z}\right] \geq \left(1 - \frac{1}{\gamma n}\right)^{2n} \approx \frac{1}{e^2}, \tag{6}$$

for sufficiently large n.

Proof. Eq. (3) implies Eq. (5). Moreover, for $z_{i,k}$ where $k = 1, 2$, we have

$$\begin{aligned}
\Pr\left[z_{i,k} \in R_q^{\leq \beta_z}\right] &= \frac{\left|R_q^{\leq \beta_z}\right|}{\left|R_q^{\leq \beta_y}\right|} \\
&= \left(\frac{2\beta_z + 1}{2\beta_y + 1}\right)^n \\
&= \left(\frac{2((\gamma n - 1)n\beta_s \log n) + 1}{2\left(\gamma n^2 \beta_s \log n\right) + 1}\right)^n \\
&= \left(\frac{2\gamma n^2 \beta_s \log n + 1 - 2n\beta_s \log n}{2\gamma n^2 \beta_s \log n + 1}\right)^n \\
&= \left(1 - \frac{2n\beta_s \log n}{2\gamma n^2 \beta_s \log n + 1}\right)^n \\
&\geq \left(1 - \frac{2n\beta_s \log n}{2\gamma n^2 \beta_s \log n}\right)^n \\
&= \left(1 - \frac{1}{\gamma n}\right)^n \\
&\geq \left(1 - \frac{1}{n}\right)^n \\
&\approx \frac{1}{e}.
\end{aligned}$$

Therefore, Eq. (6) follows. □

To the protocol succeeds to the end, all N users are required to pass the check at (viii) simultaneously. It follows from Eq. (6) that the probability Pr[pass] that all users pass the check in (viii) is evaluated by the following.

$$\begin{aligned}
\Pr[\mathsf{pass}] &= \left(\Pr\left[z_{i,1}, z_{i,2} \in R_q^{\leq \beta_z}\right]\right)^N \\
&\geq \left(1 - \frac{1}{\gamma n}\right)^{2Nn} \\
&\approx \frac{1}{e^{2N}}.
\end{aligned} \tag{7}$$

Let $E_{\gamma,n}$ be the expected iteration time of Sig. Then, $E_{\gamma,n}$ can be evaluated by $1/\Pr[\mathsf{pass}] \approx e^{2N}$.

3.4 Discussion on Multisignature Size

We discuss the size of a multisignature $\sigma = (\boldsymbol{u}, z_1, z_2) \in R_q \times R_q^{\leq N\beta_z} \times R_q^{\leq N\beta_z}$ issued by Sig. Recall that any element in R_q (in $R_q^{\leq N\beta_z}$, resp.) is represented by the tuple of n integers in $[-(q-1)/2, (q-1)/2]$ (in $[-N\beta_z, N\beta_z]$, resp.). It follows from Eq. (1) and Table 4 that the size is $n\left(\log_2 q + 2\log_2\left(2N(\gamma n - 1)n\beta_s \log n\right)\right)$. Observe that the number N of users few affect the size of multisignatures, although the size depends on N. On the other hand, this sufficiently affects the size of signatures in the case where each of N users issues a signature by using AFLT signature individually. In fact, each size of signatures issued by AFLT signature is evaluated by $n\left(\log_2 q + 2\log\left(2(n-1)\sqrt{n}\beta_s \log^3 n\right)\right)$ [17], and hence the total size is $N \cdot n\left(\log_2 q + 2\log\left(2(n-1)\sqrt{n}\beta_s \log^3 n\right)\right)$.

4. Security Proof

We prove the security of MSig from the Ring-LWE assumption and the Rejected-Ring-LWE assumption.

Theorem 1. Let $q \gg (\gamma\beta_s N)^{2/\alpha} \cdot n^{4/\alpha+\eta}$ for some α and η. Assume that the $(\beta_s, T_{\mathrm{RLWE}}, \epsilon_{\mathrm{RLWE}})$-Ring-LWE assumption and the $(\beta_s, \beta_c, \beta_y, \beta_z, T_{\mathrm{RLWE}}, \epsilon_{\mathrm{ReRLWE}})$-Re-Ring-LWE assumption hold. Then, MSig is $(T, N, \epsilon, Q_s, Q_1, Q_2)$-ppk secure in the random oracle model, where

$$\begin{aligned}
T &= T_{\mathrm{RLWE}} - O(n) = T_{\mathrm{ReRLWE}} - O(n), \\
\epsilon &\leq \epsilon_{\mathrm{RLWE}} + \frac{\left(Q_0 + Q_s N E_{\gamma,n}\right)^2}{2^{\ell_H}} + Q_s E_{\gamma,n} \epsilon_{\mathrm{ReRLWE}} \\
&\quad + \frac{Q_s E_{\gamma,n}\left(Q_s E_{\gamma,n} + Q_1 + 1\right)}{(2(\gamma n - 1)n\beta_s \log n + 1)^n} + \frac{1}{2^{\ell_H}} + \epsilon_{\mathsf{Game}_4},
\end{aligned}$$

for some negligible function $\epsilon_{\mathsf{Game}_4}$.

Proof. We prove the statement by the hybrid argument. Let $\mathcal{F}$ denote a forger against MSig. We consider the sequence of games $\mathsf{Game}_0, \ldots, \mathsf{Game}_4$ defined below. Let Win_k denote the event that $\mathcal{F}$ wins in Game_k for $k \in [0, 4]$.

(1) Game_0

Game_0 is identical to the ppk game on MSig. The precise description is as follows:

Setup C sends $(\mathbf{pp}, \mathbf{pk}^*)$ to $\mathcal{F}$, where $\mathbf{pp} \in_{\mathrm{U}} R_q^{\times}$, and $\mathbf{pk}^* = \mathbf{pp} \cdot \boldsymbol{s}_1^* + \boldsymbol{s}_2^*$ for $\boldsymbol{s}_1^*, \boldsymbol{s}_2^* \in_{\mathrm{U}} R_q^{\leq \beta_s}$.

H_0 On a query $\boldsymbol{u}^{(k)}$ from $\mathcal{F}$, C returns $H_0(\boldsymbol{u}^{(k)})$ if it is already defined, or chooses and returns $\boldsymbol{h}^{(k)} \in_{\mathrm{U}} \{0,1\}^{\ell_H}$ otherwise.

H_1 On a query $\left(\overline{\mathbf{pk}}^{(k)}, \boldsymbol{u}^{(k)}, \mathsf{PK}^{(k)}, \mu^{(k)}\right)$ from $\mathcal{F}$, C returns $H_1(\overline{\mathbf{pk}}^{(k)}, \boldsymbol{u}^{(k)}, \mathsf{PK}^{(k)}, \mu^{(k)})$ if it is already defined, or chooses and returns $\boldsymbol{c}^{(k)} \in_{\mathrm{U}} R_q^{\leq \beta_c}$ otherwise.

Sign C and $\mathcal{F}$ compute a signature $\sigma^{(t)}$ on $\left(\mathsf{PK}^{(t)}, \mu^{(t)}\right)$ according to the following protocol.
For any $t \in [1, Q_s]$, we can assume that the first public

key $\mathbf{pk}_1^{(t)}$ in $\mathsf{PK}^{(t)} = \left\{\mathbf{pk}_j^{(t)}\right\}_{j=1}^{N}$ is just $\mathbf{pk}^*$ generated in **Setup** phase without loss of generality. $\mathcal{F}$ plays a role of users which have other public keys $\mathbf{pk}_j^{(t)}$ ($j \in [2,N]$).

When $\mathcal{F}$ queries $\left(\mathsf{PK}^{(t)}, \mu^{(t)}\right)$, C behaves as follows:

(S1-1) C chooses $\boldsymbol{y}_{1,1}^{(t)}, \boldsymbol{y}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_y}$, and sets $\boldsymbol{u}_1^{(t)} = \mathbf{pp} \cdot \boldsymbol{y}_{1,1}^{(t)} + \boldsymbol{y}_{1,2}^{(t)}$.

(S1-2) C sets $\boldsymbol{h}_1^{(t)} = H_0(\boldsymbol{u}_1^{(t)})$, then returns $\boldsymbol{h}_1^{(t)}$ to $\mathcal{F}$.

When $\mathcal{F}$ sends $\boldsymbol{h}_j^{(t)}$ ($j \in [2,N]$), C returns $\boldsymbol{u}_1^{(t)}$ to $\mathcal{F}$.

When $\mathcal{F}$ sends $\boldsymbol{u}_j^{(t)}$ ($j \in [2,N]$), C behaves as follows:

(S3-1) C checks $\boldsymbol{h}_j^{(t)} = H_0(\boldsymbol{u}_j^{(t)})$ for all $j \in [2,N]$. If there exists j such that $\boldsymbol{h}_j^{(t)} \neq H_0(\boldsymbol{u}_j^{(t)})$, aborts the protocol.

(S3-2) C sets $\boldsymbol{u}^{(t)} = \sum_{j=1}^{N} \boldsymbol{u}_j^{(t)}$.

(S3-3) C sets $\boldsymbol{c}_1^{(t)} = H_1(\mathbf{pk}^*, \boldsymbol{u}^{(t)}, \mathsf{PK}^{(t)}, \mu^{(t)})$.

(S3-4) C sets $\left(\boldsymbol{z}_{1,1}^{(t)}, \boldsymbol{z}_{1,2}^{(t)}\right) = \left(\boldsymbol{s}_1^* \boldsymbol{c}_1^{(t)} + \boldsymbol{y}_{1,1}^{(t)}, \boldsymbol{s}_2^* \boldsymbol{c}_1^{(t)} + \boldsymbol{y}_{1,2}^{(t)}\right)$.

(S3-5) If $\left(\boldsymbol{z}_{1,1}^{(t)}, \boldsymbol{z}_{1,2}^{(t)}\right) \notin \left(R_q^{\leq \beta_z}\right)^2$, C restarts the protocol.

(S3-6) C returns $\left(\boldsymbol{z}_{1,1}^{(t)}, \boldsymbol{z}_{1,2}^{(t)}\right)$ to $\mathcal{F}$.

When $\mathcal{F}$ sends $\left(\boldsymbol{z}_{j,1}^{(t)}, \boldsymbol{z}_{j,2}^{(t)}\right)$ ($j \in [2,N]$), C behaves as follows:

(S4-1) C computes $\left(\boldsymbol{z}_1^{(t)}, \boldsymbol{z}_2^{(t)}\right) = \sum_{j=1}^{N} \left(\boldsymbol{z}_{j,1}^{(t)}, \boldsymbol{z}_{j,2}^{(t)}\right)$.

(S4-2) C outputs a signature $\sigma = \left(\boldsymbol{u}^{(t)}, \boldsymbol{z}_1^{(t)}, \boldsymbol{z}_2^{(t)}\right)$.

Challenge If the final output $(\mathsf{PK}^*, \mu^*, \sigma^*)$ of $\mathcal{F}$ satisfies the following conditions, $\mathcal{F}$ wins the game:

1. $\mathbf{pk}^* \in \mathsf{PK}^*$.
2. (μ^*, PK^*) is not queried in **Sign** phase.
3. $\sigma^* = \left(\boldsymbol{u}^*, \boldsymbol{z}_1^*, \boldsymbol{z}_2^*\right)$ satisfies
   a. $\boldsymbol{z}_1^*, \boldsymbol{z}_2^* \in R_q^{\leq N\beta_z}$,
   b. $\mathbf{pp} \cdot \boldsymbol{z}_1^* + \boldsymbol{z}_2^* = \boldsymbol{u}^* + \sum_{j=1}^{N} \mathbf{pk}_j^* \cdot \boldsymbol{c}_j^*$, where $\mathsf{PK}^* = \left\{\mathbf{pk}_j^*\right\}_{j=1}^{N}$ and $\boldsymbol{c}_j^* = H_1(\mathbf{pk}_j^*, \boldsymbol{u}^*, \mathsf{PK}^*, \mu^*)$.

Note that H_0 and H_1 denote phases for the random oracle queries to H_0 and H_1, respectively.

Since MSig is $(T, N, \epsilon, Q_s, Q_1, Q_2)$-ppk secure, we have the following lemma.

Lemma 3.

$$\Pr[\mathsf{Win}_0] = \epsilon. \tag{8}$$

Proof. Game_0 is completely identical to the ppk game. Thus the winning probability of the forger in Game_0 and that of the ppk game equal. □

(2) Game_1

Game_1 is identical to Game_0 except for the behaver of C in H_0 phase. On a k-th query $\boldsymbol{u}^{(k)}$ from $\mathcal{F}$, C behaves as follows:

1. C returns $H_0(\boldsymbol{u}^{(k)})$ if it is already defined.
2. Otherwise, C chooses $\boldsymbol{h}^{(k)} \in_{\mathrm{U}} \{0,1\}^{\ell_H}$.
3. C checks whether or not there exists $j \in [1, k-1]$ such that $\boldsymbol{h}^{(k)} = H_0(\boldsymbol{u}^{(j)})$ for the previous query $\boldsymbol{u}^{(j)}$. If such an index j exists, then C aborts the game.
4. Otherwise, C sets $H_0(\boldsymbol{u}^{(k)}) := \boldsymbol{h}^{(k)}$ and returns $\boldsymbol{h}^{(k)}$ to $\mathcal{F}$.

Lemma 4.

$$\Pr[\mathsf{Win}_1] \geq \Pr[\mathsf{Win}_0] - \frac{\left(Q_0 + Q_s N E_{\gamma,n}\right)^2}{2^{\ell_H}}. \tag{9}$$

Proof. The difference between Game_1 and Game_0 is that C aborts the game when C tries to set the same hash value for different two inputs. We evaluate the probability that C aborts the game in this situation. $\mathcal{F}$ makes queries to H_0 oracle is at most Q_0, whereas for each t-th query in **Sign** phase, C does N times as seen in (S1-2) and (S3-1). Since **Sign** phase is repeated at most $E_{\gamma,n}$ times as evaluated in Sect. 3.3 for each t-th query by $\mathcal{F}$, the total number of calling H_0 oracle is at most $Q_0 + Q_s N E_{\gamma,n}$. Thus the probability that C aborts the game is

$$\begin{aligned}
&\sum_{t=1}^{Q_0 + Q_s N E_{\gamma,n}} \Pr_{\boldsymbol{h}_t \in_{\mathrm{U}} R_q}\left[\boldsymbol{h}_t \in \left\{\boldsymbol{h}_j\right\}_{j=1}^{t-1}\right] \\
&\leq \sum_{t=1}^{Q_0 + Q_s N E_{\gamma,n}} \frac{t-1}{2^{\ell_H}} \\
&\leq \frac{\left(Q_0 + Q_s N E_{\gamma,n}\right)^2}{2^{\ell_H}}.
\end{aligned}$$

Then, the winning probability of $\mathcal{F}$ is

$$\Pr[\mathsf{Win}_1] \geq \Pr[\mathsf{Win}_0] - \frac{\left(Q_0 + Q_s N E_{\gamma,n}\right)^2}{2^{\ell_H}}.$$

□

(3) Game_2

Game_2 is identical to Game_1 except the processes just after "$\mathcal{F}$ **sends** $\boldsymbol{h}_j^{(t)}$ ($j \in [2,N]$)" and the process (S3-1) at **Sign** phase. In the same way as Game_0, we assume that the first public key $\mathbf{pk}_1^{(t)}$ in $\mathsf{PK}^{(t)} = \left\{\mathbf{pk}_j^{(t)}\right\}_{j=1}^{N}$ is $\mathbf{pk}^*$ which is generated in **Setup** phase.

When $\mathcal{F}$ sends $\boldsymbol{h}_j^{(t)}$ ($j \in [2,N]$), C behaves as follows:

(Sim2-1) For each $j \in [2,N]$, C finds $\tilde{\boldsymbol{u}}_j^{(t)}$ such that $\boldsymbol{h}_j^{(t)} = H_0(\tilde{\boldsymbol{u}}_j^{(t)})$ from the list of query-answer pairs in H_0 phase.

(Sim2-2) If there does not exist $\tilde{\boldsymbol{u}}_{j'}^{(t)}$ such that $\boldsymbol{h}_{j'}^{(t)} = H_0(\tilde{\boldsymbol{u}}_{j'}^{(t)})$ for some j', C sets $\tilde{\boldsymbol{u}}_{j'}^{(t)} \in_{\mathrm{U}} R_q$ and $\mathsf{bad} = 1$.

(Sim2-3) C sets $\boldsymbol{u}^{(t)} = \boldsymbol{u}_1^{(t)} + \sum_{j=2}^{N} \tilde{\boldsymbol{u}}_j^{(t)}$.

(Sim2-4) If $H_1(\mathbf{pk}^*, \boldsymbol{u}^{(t)}, \mathsf{PK}^{(t)}, \mu^{(t)})$ is already defined to some value which is not $\boldsymbol{c}_1^{(t)}$, C aborts the protocol.

(Sim2-5) C sets $\boldsymbol{c}_1^{(t)} = H_1(\mathbf{pk}^*, \boldsymbol{u}^{(t)}, \mathsf{PK}^{(t)}, \mu^{(t)})$.

(Sim2-6) C returns $\boldsymbol{u}_1^{(t)}$ to $\mathcal{F}$.

Moreover, (S3-1) is replaced with "C checks $\boldsymbol{u}_j^{(t)} = \tilde{\boldsymbol{u}}_j^{(t)}$ for all $j \in [2,N]$. If there exists j such that $\boldsymbol{u}_j^{(t)} \neq \tilde{\boldsymbol{u}}_j^{(t)}$ or $\mathsf{bad} = 1$, then aborts the protocol."

The following lemma guarantees that this change does not affect the success probability by $\mathcal{F}$ unless one of the bad events happens, and the probability that it happens is negligible. Here, the *bad events* are the events that C sets $\mathsf{bad} = 1$ in (Sim2-2) and that aborts in (Sim2-4).

Lemma 5.

$$\Pr[\mathsf{Win}_2] \geq \Pr[\mathsf{Win}_1] - \frac{Q_s E_{\gamma,n}\left(Q_s E_{\gamma,n} + Q_1 + 1\right)}{(2(\gamma n - 1)n\beta_s \log n + 1)^n} - \frac{1}{2^{\ell_H}}. \tag{10}$$

Proof. We first show that the change at Game_2 does not affect the success probability by $\mathcal{F}$ unless one of the bad events happens. Since C sends $\boldsymbol{u}_1^{(t)}$, which is set before "$\mathcal{F}$ **sends** $\boldsymbol{h}_j^{(t)}$ ($j \in [2,N]$)" in both games, the black-box forger $\mathcal{F}$ cannot recognize the change in this game at the point where $\mathcal{F}$ receives $\boldsymbol{u}_1^{(t)}$ unless one of the bad events happens.

We now confirm that the replaced equation $\boldsymbol{u}_j^{(t)} = \tilde{\boldsymbol{u}}_j^{(t)}$ for all $j \in [2,N]$ is equivalent to the original one $\boldsymbol{h}_j^{(t)} = H_0(\boldsymbol{u}_j^{(t)})$ for all $j \in [2,N]$. We fix j. Then, (Sim2-1) implies that $\boldsymbol{h}_j^{(t)} = H_0(\tilde{\boldsymbol{u}}_j^{(t)})$. This means that if $\boldsymbol{u}_j^{(t)} = \tilde{\boldsymbol{u}}_j^{(t)}$, then we have $\boldsymbol{h}_j^{(t)} = H_0(\tilde{\boldsymbol{u}}_j^{(t)}) = H_0(\boldsymbol{u}_j^{(t)})$. On the contrary, if $\boldsymbol{h}_j^{(t)} = H_0(\boldsymbol{u}_j^{(t)})$, then $H_0(\boldsymbol{u}_j^{(t)}) = \boldsymbol{h}_j^{(t)} = H_0(\tilde{\boldsymbol{u}}_j^{(t)})$. It follows from the setting of Game_1 that there is only one value $\boldsymbol{u}$ such that $\boldsymbol{h}_j^{(t)} = H_0(\boldsymbol{u})$. This implies that $\boldsymbol{u}_j^{(t)} = \tilde{\boldsymbol{u}}_j^{(t)}$. Thus, the above two equations are equivalent. Hence, the behaviors of C in both games are equivalent.

We evaluate the probability that one of the bad events happens. C may set $\mathsf{bad} = 1$ in (Sim2-2) if there exists some j' such that there does not exist $\tilde{\boldsymbol{u}}_{j'}^{(t)}$ which satisfies $\boldsymbol{h}_{j'}^{(t)} = H_0(\tilde{\boldsymbol{u}}_{j'}^{(t)})$. This means that $\mathcal{F}$ sends $\boldsymbol{h}_{j'}^{(t)}$ which is not queried to H_0 phase, although $\mathcal{F}$ has to find $\boldsymbol{u}_{j'}^{(t)}$ satisfying $H_0(\boldsymbol{u}_{j'}^{(t)}) = \boldsymbol{h}_{j'}^{(t)}$. Since C chooses a hash value from R_q uniformly at random for a queried $\boldsymbol{u}$, the probability that $\mathcal{F}$ finds an appropriate $\boldsymbol{h}_{j'}^{(t)}$ is $1/2^{\ell_H}$. Namely the probability of $\mathsf{bad} = 1$ in (Sim2-2) is bounded by $1/2^{\ell_H}$.

C may also abort the protocol in (Sim2-4), if $H_1(\mathbf{pk}^*, \boldsymbol{u}^{(t)}, \mathsf{PK}^{(t)}, \mu^{(t)})$ is already defined. In order to evaluate this event, we estimate the distribution of $\boldsymbol{u}^{(t)} = \sum_{j=1}^{N} \boldsymbol{u}_j^{(t)}$ in input, especially that of $\boldsymbol{u}_1^{(t)}$. Since $\boldsymbol{u}_1^{(t)} = \mathbf{pp} \cdot \boldsymbol{z}_{1,1}^{(t)} + \boldsymbol{z}_{1,2}^{(t)} - \mathbf{pk}^* \cdot \boldsymbol{c}_1^{(t)}$ for $\boldsymbol{z}_{1,1}^{(t)}, \boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}$, for any $\boldsymbol{u} \in R_q$, we have

$$\begin{aligned}
&\Pr_{\boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}}\left[\boldsymbol{u} = \mathbf{pp} \cdot \boldsymbol{z}_{1,1}^{(t)} + \boldsymbol{z}_{1,2}^{(t)} - \mathbf{pk}^* \cdot \boldsymbol{c}_1^{(t)}\right] \\
&= \Pr_{\boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}}\left[\boldsymbol{z}_{1,2}^{(t)} = \boldsymbol{u} - \mathbf{pp} \cdot \boldsymbol{z}_{1,1}^{(t)} + \mathbf{pk}^* \cdot \boldsymbol{c}_1^{(t)}\right] \\
&= \Pr_{\boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}}\left[\boldsymbol{z}_{1,2}^{(t)} = Z\right] \\
&= \Pr_{\boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}}\left[\boldsymbol{z}_{1,2}^{(t)} = Z \wedge Z \in R_q^{\leq \beta_z}\right] \\
&\quad + \Pr_{\boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}}\left[\boldsymbol{z}_{1,2}^{(t)} = Z \wedge Z \notin R_q^{\leq \beta_z}\right] \\
&= \Pr_{\boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}}\left[\boldsymbol{z}_{1,2}^{(t)} = Z \wedge Z \in R_q^{\leq \beta_z}\right] \\
&= \Pr\left[Z \in R_q^{\leq \beta_z}\right] \Pr_{\boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}}\left[\boldsymbol{z}_{1,2}^{(t)} = Z \mid Z \in R_q^{\leq \beta_z}\right] \\
&\leq \Pr_{\boldsymbol{z}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}}\left[\boldsymbol{z}_{1,2}^{(t)} = Z \mid Z \in R_q^{\leq \beta_z}\right] \\
&= \frac{1}{\left|R_q^{\leq \beta_z}\right|}.
\end{aligned}$$

where Z stands for $\boldsymbol{u} - \mathbf{pp} \cdot \boldsymbol{z}_{1,1}^{(t)} + \mathbf{pk}^* \cdot \boldsymbol{c}_1^{(t)}$. The number of calling **Sign** phase is at most Q_s, and **Sign** is repeated at most $E_{\gamma,n}$ times as mentioned in Sect. 3.3. Since $\boldsymbol{u}_j^{(t)}$ is determined before $\boldsymbol{u}_1^{(t)}$ has been broadcasted for $j \in [2,N]$, the probability of aborting in (Sim2-4) is bounded by

$$\sum_{t=1}^{Q_s E_{\gamma,n}} \frac{t + Q_1}{\left|R_q^{\leq \beta_z}\right|} \leq \frac{Q_s E_{\gamma,n}\left(Q_s E_{\gamma,n} + 1\right) + Q_s E_{\gamma,n} Q_1}{\left|R_q^{\leq \beta_z}\right|}.$$

From the discussion above, we can evaluate $\Pr[\mathsf{Win}_2]$ as

$$\begin{aligned}
\Pr[\mathsf{Win}_2] &\geq \Pr[\mathsf{Win}_1] - \frac{Q_s E_{\gamma,n}\left(Q_s E_{\gamma,n} + Q_1 + 1\right)}{\left|R_q^{\leq \beta_z}\right|} - \frac{1}{2^{\ell_H}} \\
&= \Pr[\mathsf{Win}_1] - \frac{Q_s E_{\gamma,n}\left(Q_s E_{\gamma,n} + Q_1 + 1\right)}{(2(\gamma n - 1)n\beta_s \log n + 1)^n} - \frac{1}{2^{\ell_H}}.
\end{aligned}$$

□

(4) Game_3

Game_3 is identical to Game_2 except the processes just after "$\mathcal{F}$ **queries** $\left(\mathsf{PK}^{(t)}, \mu^{(t)}\right)$", those just after "$\mathcal{F}$ **sends** $\boldsymbol{u}_j^{(t)}$ $(j \in [2,N])$" and the process (Sim2-5) at **Sign** phase.

When $\mathcal{F}$ queries $\left(\mathsf{PK}^{(t)}, \mu^{(t)}\right)$, C behaves as follows:

(Sim1-1) C chooses $\boldsymbol{c}_1^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_c}$ and $z_{1,1}^{(t)}, z_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_z}$.

(Sim1-2) C sets $b = 1$ with the probability $1 - \left(\left|R_q^{\leq \beta_z}\right| / \left|R_q^{\leq \beta_y}\right|\right)^2$ which is indicated in Eq. (5), or sets $b = 0$ otherwise.

(Sim1-3) C computes $\boldsymbol{u}_1^{(t)} = \mathbf{pp} \cdot z_{1,1}^{(t)} + z_{1,2}^{(t)} - \mathbf{pk}^* \cdot \boldsymbol{c}_1^{(t)}$ if $b = 0$, or $\boldsymbol{u}_1^{(t)} \in_{\mathrm{U}} R_q$ otherwise.

(Sim1-4) C sets $\boldsymbol{h}_1^{(t)} = H_0(\boldsymbol{u}_1^{(t)})$, and then returns $\boldsymbol{h}_1^{(t)}$ to $\mathcal{F}$.

When $\mathcal{F}$ sends $\boldsymbol{u}_j^{(t)}$ $(j \in [2,N])$, C behaves as follows:

(Sim3-1) C checks $\boldsymbol{u}_j^{(t)} = \tilde{\boldsymbol{u}}_j^{(t)}$ for all $j \in [2,N]$. If there exists j such that $\boldsymbol{u}_j^{(t)} \neq \tilde{\boldsymbol{u}}_j^{(t)}$ or $\mathsf{bad} = 1$, then aborts the protocol.

(Sim3-2) C restarts the protocol if $b = 1$.

(Sim3-3) C returns $\left(z_{1,1}^{(t)}, z_{1,2}^{(t)}\right)$ to $\mathcal{F}$.

We also replace "C sets $\boldsymbol{c}_1^{(t)} = H_1(\mathbf{pk}^*, \boldsymbol{u}^{(t)}, \mathsf{PK}^{(t)}, \mu^{(t)})$" at (Sim2-5) with "$C$ sets $H_1(\mathbf{pk}^*, \boldsymbol{u}^{(t)}, \mathsf{PK}^{(t)}, \mu^{(t)}) = \boldsymbol{c}_1^{(t)}$", where $\boldsymbol{c}_1^{(t)}$ is chosen at (Sim1-1).

In (Sim1-2), C picks the bit b randomly. To reflect the event of the restart in (S3-5) of Game_2, it is decided whether or not C restarts the signing protocol. The case $b = 1$ represents that C attempts to simulate the signing oracle with the restart event. Indeed, the probability that $b = 1$ is $1 - \left(\left|R_q^{\leq \beta_z}\right| / \left|R_q^{\leq \beta_y}\right|\right)^2$ which expresses the restart probability in (S3-5) of Game_2. We show by the following lemma that C on Game_3 with $b = 1$ behaves in the computationally same as the processes of Game_2 with restart, whereas it on Game_3 with $b = 0$ does in the computationally same as those of Game_2 with non-restart.

Lemma 6.

$$\Pr[\mathsf{Win}_3] \geq \Pr[\mathsf{Win}_2] - Q_s E_{\gamma,n} \epsilon_{\mathrm{ReRLWE}}. \tag{11}$$

Proof. We first show that $\sigma^{(t)} = \left(\boldsymbol{u}^{(t)}, z_1^{(t)}, z_2^{(t)}\right)$ satisfies $\mathsf{Ver}(\mathbf{pp}, \mathsf{PK}^{(t)}, \mu^{(t)}, \sigma^{(t)}) = 1$ when C does not abort the protocol in **Sign** phase and $\mathcal{F}$ honestly plays the role of users having all $\mathbf{pk}_j$ $(j \in [2,N])$.

We have $z_{1,1}^{(t)}, z_{1,2}^{(t)} \in R_q^{\leq \beta_z}$ by (Sim1-1). For each $j \in [2,N]$, it always holds $z_{j,1}^{(t)}, z_{j,2}^{(t)} \in R_q^{\leq \beta_z}$ if $\mathcal{F}$ honestly plays the role of the users having $\mathbf{pk}_j^{(t)}$. Therefore, the condition (i) of Ver holds as Eq. (1).

For $(z_{1,1}^{(t)}, z_{1,2}^{(t)})$, $\mathbf{pp} \cdot z_{1,1}^{(t)} + z_{1,2}^{(t)} = \mathbf{pk}_1^{(t)} \cdot \boldsymbol{c}_1^{(t)} + \boldsymbol{u}_1^{(t)}$ follows by (Sim1-3). For each $j \in [2,N]$, $\mathbf{pp} \cdot z_{j,1}^{(t)} + z_{j,2}^{(t)} = \mathbf{pk}_j^{(t)} \cdot \boldsymbol{c}_j^{(t)} + \boldsymbol{u}_j^{(t)}$ also holds if $\mathcal{F}$ honestly plays the role of the users having $\mathbf{pk}_j^{(t)}$. From the setting of H_0 phase on Game_1, we have $H_0(\boldsymbol{u}) \neq H_0(\tilde{\boldsymbol{u}})$ for all $\boldsymbol{u} \neq \tilde{\boldsymbol{u}}$. Then, it holds that $\boldsymbol{u}^{(t)} = \sum_{j=1}^N \boldsymbol{u}_j^{(t)} = \boldsymbol{u}_1^{(t)} + \sum_{j=2}^N \tilde{\boldsymbol{u}}_j^{(t)}$. Thus, the condition (ii) of Ver holds as Eq. (2).

We next show that the distribution of

$$\left(\boldsymbol{h}_1^{(t)}, \boldsymbol{u}_1^{(t)}, z_{1,1}^{(t)}, z_{1,2}^{(t)}, \boldsymbol{u}^{(t)}, z_1^{(t)}, z_2^{(t)}\right),$$

in Game_2 when $\left(z_{1,1}^{(t)}, z_{1,2}^{(t)}\right) \in \left(R_q^{\leq \beta_z}\right)^2$, which is output by C in **Sign** phase, is identical to that in Game_3 when $b = 0$. $\boldsymbol{h}_1^{(t)}$ is chosen uniformly at random from R_q in H_0 phase in both games. In Game_2, $\boldsymbol{u}_1^{(t)}$ can be expressed as

$$\begin{aligned}
\boldsymbol{u}_1^{(t)} &= \mathbf{pp} \cdot \boldsymbol{y}_{1,1}^{(t)} + \boldsymbol{y}_{1,2}^{(t)} \\
&= \mathbf{pp}\left(z_{1,1}^{(t)} - \boldsymbol{s}_1^* \boldsymbol{c}_1^{(t)}\right) + z_{1,2}^{(t)} - \boldsymbol{s}_2^* \boldsymbol{c}_1^{(t)} \\
&= \mathbf{pp} \cdot z_{1,1}^{(t)} + z_{1,2}^{(t)} - \left(\mathbf{pp} \cdot \boldsymbol{s}_1^* + \boldsymbol{s}_2^*\right) \boldsymbol{c}_1^{(t)} \\
&= \mathbf{pp} \cdot z_{1,1}^{(t)} + z_{1,2}^{(t)} - \mathbf{pk}^* \cdot \boldsymbol{c}_1^{(t)}.
\end{aligned}$$

It follows from Eq. (4) of Lemma 1 that the distributions of $z_{1,1}^{(t)}$ and $z_{1,2}^{(t)}$ are uniformly at random over $R_q^{\leq \beta_z}$ for the uniformly chosen elements $\boldsymbol{y}_{1,1}^{(t)}, \boldsymbol{y}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_y}$ under the condition $\left(z_{1,1}^{(t)}, z_{1,2}^{(t)}\right) \in \left(R_q^{\leq \beta_z}\right)^2$. This implies that those of $\boldsymbol{u}_1^{(t)}$ as well as $\left(z_{1,1}^{(t)}, z_{1,2}^{(t)}\right)$ in the both games are identical, as long as it holds that $\left(z_{1,1}^{(t)}, z_{1,2}^{(t)}\right) \in \left(R_q^{\leq \beta_z}\right)^2$ in Game_2. For $\boldsymbol{u}^{(t)}, z_1^{(t)}, z_2^{(t)}$, it holds

$$\left(\boldsymbol{u}^{(t)}, z_1^{(t)}, z_2^{(t)}\right) = \sum_{i=1}^N \left(\boldsymbol{u}_i^{(t)}, z_{i,1}^{(t)}, z_{i,2}^{(t)}\right),$$

in both games. Since the distributions of $\left(\boldsymbol{u}_1^{(t)}, z_{1,1}^{(t)}, z_{1,2}^{(t)}\right)$ in the both games are identical as mentioned above and those of $\left(\boldsymbol{u}_j^{(t)}, z_{j,1}^{(t)}, z_{j,2}^{(t)}\right)$ for any $j \in [2,N]$ are chosen by $\mathcal{F}$, the distributions of

$$\left(\boldsymbol{h}_1^{(t)}, \boldsymbol{u}_1^{(t)}, z_{1,1}^{(t)}, z_{1,2}^{(t)}, \boldsymbol{u}^{(t)}, z_1^{(t)}, z_2^{(t)}\right),$$

between Game_2 and Game_3 equal.

We consider the case where $\left(z_{1,1}^{(t)}, z_{1,2}^{(t)}\right) \notin \left(R_q^{\leq \beta_z}\right)$ in Game_2 and $b = 1$ in Game_3, namely C restarts the protocol. In this case, the conversation between C and $\mathcal{F}$ consists of $\left(\boldsymbol{h}_1^{(t)}, \boldsymbol{u}_1^{(t)}\right)$ from the beginning of **Sign** phase to the time C restarts the protocol. Consider the distributions of $\left(\boldsymbol{h}_1^{(t)}, \boldsymbol{u}_1^{(t)}\right)$ in both games. $\boldsymbol{h}_1^{(t)}$ is chosen uniformly at random from R_q in H_0 phase in both games, thus the distributions of $\boldsymbol{h}_1^{(t)}$ in both games are identical. In Game_2, $\boldsymbol{u}_1^{(t)}$ is computed as $\boldsymbol{u}_1^{(t)} = \mathbf{pp} \cdot \boldsymbol{y}_{1,1}^{(t)} + \boldsymbol{y}_{1,2}^{(t)}$ with $\boldsymbol{y}_{1,1}^{(t)}, \boldsymbol{y}_{1,2}^{(t)} \in_{\mathrm{U}} R_q^{\leq \beta_y}$,

whereas $\boldsymbol{u}_1^{(t)} \in_{\mathrm{U}} R_q$ in Game_3. Then the difference between the distributions of $\boldsymbol{u}_1^{(t)}$ is bounded by $\epsilon_{\mathrm{ReRLWE}}$ which is induced from the $(\beta_s, \beta_c, \beta_y, \beta_z, T_{\mathrm{ReRLWE}}, \epsilon_{\mathrm{ReRLWE}})$-Re-Ring-LWE assumption. Since $\mathcal{F}$ makes at most Q_s queries in **Sign** phase and C restarts the protocol at most $E_{\gamma,n}$ times for each query, the total difference is bounded by $Q_s E_{\gamma,n} \epsilon_{\mathrm{ReRLWE}}$.

Thus, we can evaluate $\Pr[\mathsf{Win}_3]$ as

$$\Pr[\mathsf{Win}_3] \geq \Pr[\mathsf{Win}_2] - Q_s E_{\gamma,n} \epsilon_{\mathrm{ReRLWE}}.$$

□

(5) Game_4

Game_4 is identical to Game_3 except that **Setup** phase is changed as follows: C sends $(\mathbf{pp}, \mathbf{pk}^*)$ to $\mathcal{F}$ as the target public key, where $\mathbf{pp} \in_{\mathrm{U}} R_q^{\times}$ and $\mathbf{pk}^* \in_{\mathrm{U}} R_q$.

Lemma 7.

$$|\Pr[\mathsf{Win}_4] - \Pr[\mathsf{Win}_3]| \leq \epsilon_{\mathrm{RLWE}}. \tag{12}$$

Proof. The difference between Game_4 and Game_3 is the way of generating the target public key $\mathbf{pk}^*$. Thus the statement follows by definition of $\mathsf{Adv}^{\mathrm{RLWE}}_{\mathcal{A},\beta_s}$. □

(6) $\mathcal{F}$'s winning probability in Game_4

We evaluate the probability that $\mathcal{F}$ wins in Game_4. We denote by $(\mathsf{PK}^*, \mu^*, \sigma^*) = \left(\mathsf{PK}^*, \mu^*, \left(\boldsymbol{u}^*, z_1^*, z_2^*\right)\right)$ the output of $\mathcal{F}$ in **Challenge** phase, where $\mathsf{PK}^* = \left\{\mathbf{pk}_j^*\right\}_{j=1}^N$, $\boldsymbol{c}_j^* = H_1(\mathbf{pk}_j^*, \boldsymbol{u}^*, \mathsf{PK}^*, \mu^*)$ $(j \in [1, N])$. For PK^*, let J^* be the set of all the indexies j such that $\mathbf{pk}_j^* = \mathbf{pk}^*$.

This probability can be shown by using the notion of an α-partial modular ratio [17]. Let α be a rational number such that $0 < \alpha < 1$, the denominator of α is a small power of 2 and αn is an integer. For $d = \alpha n$, $\boldsymbol{t} \in R_q$ is an *α-partial modular ratio with respect to* $\mathbf{pp}$ if there exist elements $\boldsymbol{c}, \boldsymbol{c}' \in R_q^{\leq \beta_c}$, $z_1, \tilde{z_1}, z_2, \tilde{z_2} \in R_q^{N\beta_z}$ and a degree d-divisor $P \in \mathbb{Z}_q[X]$ of $X^n + 1$ such that

- $\boldsymbol{c} - \boldsymbol{c}'$ is invertible modulo P; and
- $\boldsymbol{t} \equiv \dfrac{\mathbf{pp}(z_1 - \tilde{z_1}) + (z_2 - \tilde{z_2})}{\boldsymbol{c} - \boldsymbol{c}'} \pmod{P}$.

We introduce the following lemma.

Lemma 8 (Lemma 2 [17])**.** Let $\boldsymbol{t} \in_{\mathrm{U}} R_q$. We have

$$\begin{aligned}
&\Pr[\boldsymbol{t} \text{ is an } \alpha\text{-partial modular ratio}] \\
&\leq \binom{n}{d}\left(\frac{(4N\beta_z + 1)^{2n}(2\beta_c + 1)^n}{q^s}\right) \\
&\leq \binom{n}{d}\left(\frac{33n^4(\gamma\beta_s N)^2 \log^3 n}{q^{\alpha}}\right)^n.
\end{aligned}$$

Namely, this probability is negligible if $q \gg (\gamma\beta_s N)^{2/\alpha} \cdot n^{4/\alpha+\eta}$ for some $\eta > 0$.

We now estimate that for $\boldsymbol{u}^*$, there are few fractions of $\boldsymbol{c} \in R_q^{\leq \beta_c}$ that satisfies the verification formula i.e. the condition 3-b of **Challenge** phase. Assume that there are two hash values $\boldsymbol{c}^*, \tilde{\boldsymbol{c}}^* \in R_q^{\leq \beta_c}$ satisfying the verification formula. By letting m be the cardinality of J^*, we have

$$\begin{aligned}
\mathbf{pp} \cdot z_1^* + z_2^* &= \boldsymbol{u}^* + \sum_{j=1}^N \mathbf{pk}_j^* \boldsymbol{c}_j^* \\
&= \boldsymbol{u}^* + m \cdot \mathbf{pk}^* \boldsymbol{c}^* + \sum_{j \in [1,N] \setminus J^*} \mathbf{pk}_j^* \boldsymbol{c}_j^*, \\
\mathbf{pp} \cdot \tilde{z}_1^* + \tilde{z}_2^* &= \boldsymbol{u}^* + \sum_{j=1}^N \mathbf{pk}_j^* \boldsymbol{c}_j^* \\
&= \boldsymbol{u}^* + m \cdot \mathbf{pk}^* \tilde{\boldsymbol{c}}^* + \sum_{j \in [1,N] \setminus J^*} \mathbf{pk}_j^* \boldsymbol{c}_j^*,
\end{aligned}$$

for some $\left(z_1^*, z_2^*\right), \left(\tilde{z}_1^*, \tilde{z}_2^*\right) \in \left(R_q^{\leq N\beta_z}\right)^2$.

Let $\boldsymbol{t}^* = m \cdot \mathbf{pk}^*$. By these equations, we have

$$\mathbf{pp} \cdot \left(z_1^* - \tilde{z}_1^*\right) + \left(z_2^* - \tilde{z}_2^*\right) = \boldsymbol{t}^*(\boldsymbol{c}^* - \tilde{\boldsymbol{c}}^*).$$

The winning condition $\mathbf{pk}^* \in \mathsf{PK}$ implies that $1 \leq m \leq N$. Then, m is a unit of $\mathbb{Z}_q$, since q is prime and N is smaller than q. We have that $\boldsymbol{t}^*$ is uniformly distributed over R_q. This is because the map $\boldsymbol{t} \in R_q$ to $m \cdot \boldsymbol{t} \in R_q$ is bijective, and $\mathbf{pk}^*$ is chosen uniformly at random over R_q in this game. It follows from Lemma 8 that $\boldsymbol{t}^*$ is α-partial ratio with the negligible probability. Therefore, we now assume that $\boldsymbol{t}^*$ is not α-partial ratio. As discussed in the proof of *Lossiness* of Theorem 4 in [17], $\boldsymbol{c}^* \equiv \tilde{\boldsymbol{c}}^* \pmod{\tilde{P}}$ for some degree $(n-d)$ divisor $\tilde{P}$ of $X^n + 1$, and then the total fraction of elements $\boldsymbol{c}' \in R_q^{\leq \beta_c}$ satisfying the verification formula is bounded by

$$\binom{n}{n-d}\left(\frac{1}{2\log n}\right)^{n-d} \ll \left(\frac{1}{2\alpha^{\alpha}(1-\alpha)^{1-\alpha}\log^{1-\alpha} n}\right)^n.$$

Now, the element $\boldsymbol{c}^*$ is chosen by C in H_1 phase and is uniformly distributed over $R_q^{\leq c}$. This means that for $\mathcal{F}$, the probability of the appearance of $\boldsymbol{c}^*$ which satisfies the verification formula in this case is negligible. Thus, we can evaluate $\Pr[\mathsf{Win}_4]$ as

$$\Pr[\mathsf{Win}_4] \leq \epsilon_{\mathsf{Game}_4} \tag{13}$$

for some negligible function $\epsilon_{\mathsf{Game}_4}$.

Putting together, we have

$$\begin{aligned}
\epsilon \leq\ & \epsilon_{\mathrm{RLWE}} + \frac{\left(Q_0 + Q_s N E_{\gamma,n}\right)^2}{2^{\ell_H}} + Q_s E_{\gamma,n} \epsilon_{\mathrm{ReRLWE}} \\
&+ \frac{Q_s E_{\gamma,n}\left(Q_s E_{\gamma,n} + Q_1 + 1\right)}{(2(\gamma n - 1)n\beta_s \log n + 1)^n} + \frac{1}{2^{\ell_H}} + \epsilon_{\mathsf{Game}_4}.
\end{aligned}$$

□

5. Concluding Remarks

We have constructed a tighter security reduction for a lattice-based multisignature scheme in PPK model and ROM. Our

strategy is combining the multisignature scheme by [12] with the lattice-based signature scheme by [17]. The resulting multisigature scheme has achieved the tightness concerning the Ring-LWE assumption. Our result shows that proof techniques for standard signature schemes are applicable to multisignature schemes.

However, our security proof requires an additional assumption, the Re-Ring-LWE assumption, to address the problem concerning the repetition of the signing protocol. The Re-Ring-LWE assumption is a generalization of the Re-DCK assumption defined in [20], and thus one of the open questions is to elucidate the reasonability of this assumption with the Re-DCK assumption. Concerning this problem, the implication of the Ring-LWE assumption to the Re-Ring-LWE assumption is also an interesting problem.

The complete tight reduction for lattice-based multisignature scheme is another important open question. We do not give a complete tight reduction in this paper because the security loss concerning the Re-Ring-LWE assumption arises, whereas such a loss concerning the Ring-LWE assumption no longer appears. To resolve this question, we have two approaches. The first one is to reduce the security loss concerning the Re-Ring-LWE assumption as in the case of the Ring-LWE assumption. The second one is to enhance the parameters or the construction so that the repetition in the signing protocol no longer happens. Since these two approaches are independent, we will take both of them in the future.

Acknowledgments

We would like to thank anonymous reviewers for their valuable comments and suggestions. We are also grateful to Akira Takahashi for his fruitful comments on the security proof. This work was supported in part by JSPS KAKENHI Grant Numbers JP18K11288 and JP19K20272.

References

[1] M. Fukumitsu and S. Hasegawa, "A tightly-secure lattice-based multisignature," Proc. APKC 2019, APKC'19, Auckland, New Zealand, pp.3–11, ACM, July 2019.
[2] K. Itakura and K. Nakamura, "A public-key cryptosystem suitable for digital multisignature," NEC Research and Development, vol.71, pp.1–8, 1983.
[3] S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, Bitcoin Inc., 2008.
[4] M. Bellare and G. Neven, "Multi-signatures in the plain public-key model and a general forking lemma," Proc. ACM CCS 2006, Alexandria, USA, pp.390–399, ACM, Oct. 2006.
[5] M. Bellare, C. Namprempre, and G. Neven, "Unrestricted aggregate signatures," Proc. ICALP 2007, L. Arge, C. Cachin, T. Jurdziński, and A. Tarlecki, eds., LNCS, vol.4596, Wrocław, Poland, pp.411–422, Springer Berlin Heidelberg, July 2007.
[6] Y. Komano, K. Ohta, A. Shimbo, and S. Kawamura, "Formal security model of multisignatures," Proc. ISC 2006, S.K. Katsikas, J. López, M. Backes, S. Gritzalis, and B. Preneel, eds., LNCS, vol.4176, Samos Island, Greece, pp.146–160, Springer Berlin Heidelberg, Aug. 2006.
[7] D.P. Le, A. Bonnecaze and A. Gabillon, "Multisignatures as secure as the Diffie-Hellman problem in the plain public-key model," Proc. Pairing 2009, H. Shacham and B. Waters, ed., LNCS, vol.5671, Palo Alto, USA, pp.35–51, Springer Berlin Heidelberg, Aug. 2009.
[8] S. Lu, R. Ostrovsky, A. Sahai, H. Shacham, and B. Waters, "Sequential aggregate signatures and multisignatures without random oracles," Proc. EUROCRYPT 2006, S. Vaudenay, ed., LNCS, vol.4004, St. Petersburg, Russia, pp.465–485, Springer Berlin Heidelberg, May 2006.
[9] K. Ohta and T. Okamoto, "Multi-signature schemes secure against active insider attacks," IEICE Trans. Fundamentals, vol.E82-A, no.1, pp.21–31, Jan. 1999.
[10] N. Yanai, "Meeting tight security for multisignatures in the plain public key model," IEICE Trans. Fundamentals, vol.E101-A, no.9, pp.1484–1493, Sept. 2018.
[11] P.W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," SIAM Rev., vol.41, no.2, pp.303–332, 1999.
[12] R. El Bansarkhani and J. Sturm, "An efficient lattice-based multisignature scheme with applications to bitcoins," Proc. CANS 2016, S. Foresti and G. Persiano, eds., LNCS, vol.10052, Milan, Italy, pp.140–155, Springer International Publishing, Nov. 2016.
[13] C. Ma and M. Jiang, "Practical lattice-based multisignature schemes for blockchains," IEEE Access, vol.7, pp.179765–179778, 2019.
[14] I. Damgård, C. Orlandi, A. Takahashi, and M. Tibouchi, "Two-round n-out-of-n and multi-signatures and trapdoor commitment from lattices," Proc. PKC 2021, Part I, J.A. Garay, ed., LNCS, vol.12710, Virtual Event, pp.99–130, Springer International Publishing, 2021.
[15] A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems," Proc. CRYPTO'86, A.M. Odlyzko, ed., LNCS, vol.263, Santa Barbara, USA, pp.186–194, Springer Berlin Heidelberg, 1987.
[16] T. Güneysu, V. Lyubashevsky, and T. Pöppelmann, "Practical lattice-based cryptography: A signature scheme for embedded systems," Proc. CHES 2012, E. Prouff and P. Schaumont, eds., LNCS, vol.7428, Berlin, Heidelberg, pp.530–547, Springer Berlin Heidelberg, 2012.
[17] M. Abdalla, P.A. Fouque, V. Lyubashevsky, and M. Tibouchi, "Tightly secure signatures from lossy identification schemes," J. Cryptol., vol.29, no.3, pp.597–631, 2016.
[18] J. Katz and N. Wang, "Efficiency improvements for signature schemes with tight security reductions," Proc. ACM CCS 2003, Washington D.C., USA, pp.155–164, ACM, Oct. 2003.
[19] V. Lyubashevsky, "Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures," Proc. ASIACRYPT 2009, M. Matsui, ed., LNCS, vol.5912, Tokyo, Japan, pp.598–616, Springer Berlin Heidelberg, Dec. 2009.
[20] G. Barthe, S. Belaïd, T. Espitau, P.A. Fouque, B. Grégoire, M. Rossi, and M. Tibouchi, "Masking the GLP lattice-based signature scheme at any order," Proc. EUROCRYPT 2018, Part II, J.B. Nielsen and V. Rijmen, eds., LNCS, vol.10821, Tel Aviv, Israel, pp.354–384, Springer International Publishing, April 2018.
[21] M. Kansal and R. Dutta, "Round optimal secure multisignature schemes from lattice with public key aggregation and signature compression," Proc. AFRICACRYPT 2020, A. Nitaj and A. Youssef, eds., LNCS, vol.12174, Cairo, Egypt, pp.281–300, Springer International Publishing, July 2020.
[22] Z.Y. Liu, Y.F. Tseng, and R. Tso, "Cryptanalysis of a round optimal lattice-based multisignature scheme," Cryptology ePrint Archive, Report 2020/1172, 2020. https://eprint.iacr.org/2020/1172
[23] R. Toluee and T. Eghlidos, "An efficient and secure ID-based multi-proxy multi-signature scheme based on lattice," Cryptology ePrint Archive, Report 2019/1031, 2019. https://eprint.iacr.org/2019/1031
[24] R. Tso, Z. Liu, and Y. Tseng, "Identity-based blind multisignature from lattices," IEEE Access, vol.7, pp.182916–182923, 2019.
[25] A. Langlois and D. Stehlé, "Worst-case to average-case reductions for module lattices," Des. Codes Cryptogr., vol.75, no.3, pp.565–599, 2015.

Masayuki Fukumitsu received his B.Eng. degree in software and information science from Iwate Prefectural University, Japan, in 2009, and M.S. and Ph.D degrees in information sciences, Tohoku University, Japan, in 2011 and 2014, respectively. He has been with Hokkaido Information University since 2014, where he is an Associate Professor. His research interests include information security theory, computational complexity, discrete mathematics, and computer security.

Shingo Hasegawa received his B.Eng. degree from Tohoku University, Japan, in 2003, and M.S. and Ph.D degrees in information sciences, Tohoku University, Japan, in 2005 and 2009, respectively. He has been with Tohoku University since 2009, where he is an Assistant Professor. His research interests include information security theory, computational complexity and discrete mathematics.