How to validate the secret of a Ring Learning with Errors (RLWE) key

Jintai Ding[1], Saraswathy RV[1], Saed Alsayigh[1], and Crystal Clough[1]

University of Cincinnati

Abstract. We use the signal function from the RLWE key exchange in [26] to derive an efficient zero knowledge authentication protocol to validate an RLWE key $p = as + e$ with secret s and error e in the Random Oracle Model (ROM). With this protocol, a verifier can validate that a key p presented to him by a prover P is of the form $p = as + e$ with s, e small and that the prover knows s. We accompany the description of the protocol with proof to show that it has negligible soundness and completeness error. The soundness of our protocol relies directly on the hardness of the RLWE problem. The protocol is applicable for both LWE and RLWE but we focus on the RLWE based protocol for efficiency and practicality. We also present a variant of the main protocol with a commitment scheme to avoid using the ROM.

Keywords: RLWE, key exchange, post-quantum, key reuse, key validation, active attacks.

1 Introduction

Lattice-based cryptographic primitives are promising for use as a post-quantum alternative for existing cryptosystems that are based on classical hard problems. The search for quantum-resistant primitives has become intense in the recent year after the NSA announced its plan to transition to quantum resistant ciphersuites and NIST is calling for proposals of post-quantum primitives [50]. Due to its appealing properties like provable security, efficiency and versatility, most of the recent lattice-based schemes designed are based on the Ring-Learning with Errors (RLWE) problem. However, one of the challenges with RLWE, as pointed out in [37], [29], [25] is that, in an interaction with a malicious user not following the instructions of a protocol, the malicious user could pretend to use a RLWE key while actually using a different form to extract our secret key. This is due to the indistinguishability of RLWE samples from uniform ones. To avoid this, we need an efficient and secure authentication technique that can validate such keys to be of the correct form and that the user presenting the key knows the corresponding secret. The need for such validation protocols was also reiterated in PQCrypto2017 in the invited talk by Lyubashevsky [45]. The protocol presented in this work aims to contribute towards such a key validation technique for RLWE using simple operations in the ring. Some of the other related work includes [10, 46, 63, 42, 7]. Our protocol has a simpler structure and better adaptability for a KE protocol with key reuse, without using any additional techniques such as rejection sampling.

Authentication plays a vital role in security and can also be used to verify the identity of a process, server or network device. In the process of authentication, it is also important for a malicious verifier to not gain information about the secret of the prover that he can use to impersonate.

Zero-knowledge: The concept of zero knowledge was introduced by Goldwasser, Micali, and Rackoff [57]. A zero knowledge protocol allows a prover to convince a verifier about a statement involving a secret, without leaking any information about the secret, even if the verifier is malicious. Usually such a protocol involves a series of challenge/responses exchanges. The main properties of a zero knowledge protocol are Completeness, Soundness and Zero

knowledge. Informally, Completeness means that with overwhelming probability, an honest prover is able to successfully make a proof to the verifier. Soundness means that the probability that a malicious prover (without the secret) can make a valid proof is negligible. Zero knowledge is shown by the ability to construct a simulator that creates transcripts of the proof that are indistinguishable from a real transcript between a prover and verifier. More details and formal mathematical definitions for these properties are included in the preliminaries section of the paper.

1.1 Previous Work

There are some lattice-based Identification (ID) schemes that has its security based on Witness Indistinguishability. Lyubashevsky in [43] introduced an ID scheme based on the hardness of SIS (Shortest Integer Solutions) problem. This scheme was shown to be witness indistinguishable and used this to prove security under the active attack model. The drawback of this scheme however, is to be able to implement the scheme for practical parameter choices retaining security of the protocol. Another work in [44] presented an ID scheme with lower communication complexity based on hardness of approximate shortest vector problem in lattices. This work also described a way to convert the ID scheme into a signature in the Random oracle model using the Fiat-Shamir transform [1].

[36] describes a variant of the ID scheme proposed by [58] in the lattice setting and proves its security under concurrent attacks. A zero knowledge proof for commitments from RLWE was described in [11] that verifies that the commitments are from the ring defined in RLWE problem but does not verify that a sample is of RLWE form or the secret of a RLWE sample.This is a generalization of the protocol from Xie et al. [63] improving its knowledge error. A previous work on group signatures [10] also proposes a zero knowledge proof for LWE samples of the form $y = as + e$ to prove the knowledge of secret s. An advantage with this protocol is a reduced soundness error of $1/2n$, however this protocol relies on rejection sampling to ensure that s and e follows the Discrete Gaussian distribution that leads to lower efficiency and increased completeness error.

Key leakage was pointed out by Kirkwood et al. in [37], which discusses about elliptic point off the curve attack [12] that results in key leakage and using public key validation to avoid such attacks by checking that the public key is a valid point on the specified curve. They also discuss about the unavailability of such validation in the case of lattice-based key exchange. For RLWE-based Key agreement, they proposed an indirect key validation technique using the Fujisaki-Okamoto transformation. The transformation derives a hybrid IND-CCA (Indistinguishability Chosen Cipherext Attack) secure scheme from weakly secure symmetric and asymmetric encryption schemes. This is adapted in [54] to derive an actively secure KEM (Key Encapsulation Mechanism) but with efficiency limitations in the implementation. The indirect validation is also more complicated and limits the ability of the party being authenticated to choose a public key since it has to be generated from the private key derived from the seed. Another disadvantage with indirect validation is that it does not allow key reuse since the private key of the party being authenticated is revealed during the validation. In [20], a cut and choose protocol for RLWE is proposed. The protocol involves generating $N \approx 32$ samples for each RLWE instantiation and revealing at random all but one of the instances to the verifier. So the provers probability of convincing the verifier is at most $1/N$ if the prover doesnt know the secret.

Other notable works in the area include [46, 42, 7]. A one shot verifiable encryption protocol (proving that the ciphertext is correctly formed and the knowledge of the plaintext) was described by Lyubashevsky et al. in Eurocrypt 2017 [46]. The protocol is secure in the random oracle model and is based on the hardness of RLWE problem. The protocol achieves negligible soundness error with only one execution of the protocol. However, the protocol

does not apply to the exact relation to be verified. Let $B \in R_q^{l \times k}$ be a matrix with the relation $Bm = u \mod q$, where $m \in R_q^k$ is secret, and let $(v, w) \in R_q^k \times R_q^k$ be a ciphertext of m obtained using an RLWE encryption scheme. Then, the protocol verifies that the ciphertext decrypts to a $\bar{m}, \bar{c}$ such that $B\bar{m} = \bar{c}u \mod q$. In order to recover $\bar{m}$, the idea is to guess values for $\bar{c}$ from a defined challenge space. This also uses rejection sampling in the protocol. A statistical zero knowledge argument of knowledge for message-signature pair, that can be applied to group signatures is proposed in [42]. The protocol verifies the knowledge of a pair $(d, z) \in \{0,1\}^l \times \mathbb{Z}^{2m}$ that satisfies $\|z\|_\infty \leq \beta$, and $A_{(d)}z = u \mod q$, where $A_{(d)} = [A|A_0 + \sum_{i=1}^{l} d_i A_i] \in \mathbb{Z}_q^{n \times 2m}$. Here, uniformly random $(A, A_0, \ldots A_l, u)$, where each $A_i \in \mathbb{Z}_q^{n \times m}$ and $u \in \mathbb{Z}_q^n$ form the verification key of Boyen's signature scheme. This protocol has a communication complexity of $(O(lm)log\ \beta + O(k_2)log\ b)log\ q$ (where n is the security parameter, l is the message length, $m \geq 2nlog\ q$, $\sigma = \Omega(\sqrt{lnlog\ q}log\ n)$, $\beta = \sigma\omega(\sqrt{log\ m}), k_2, b$ are parameters of underlying encryption scheme), with a soundness error of $2/3$ and perfect completeness. The work in [7] discusses a zero knowledge proof for the preimage of a ivOWF(one way function over integer vectors). This proof can be applied to the context of learning with errors considering the encryption function as a ivOWF. This ZK protocol is honest verifier computational ZK and uses rejection sampling and cut and choose technique for its proof.

1.2 Our Contributions

The main contributions of this paper are the design and analysis of an efficient zero knowledge authentication protocol that can be used to validate a sample p to be of the form $as + e$ with s, e small and the knowledge of s of the party presenting the sample. This is accomplished by making use of the signal function defined in the key exchange protocol using RLWE [26]. The Sig function has a significant role in the key exchange protocol, since it allows us to extract a shared key material from approximately equivalent keys and can be leveraged to derive other applications as in this work with respect to the RLWE-based key exchange. In this authentication protocol, the prover uses a challenge/response interactive proof to prove to the verifier that he has the secret s and the verification is done with the help of the Sig function. We show that the soundness error of the protocol is $1/2$ and we can achieve negligible soundness error by sequentially running the protocol enough number of times.

The relevance of this protocol comes from the observation that key reuse can be exploited in RLWE-based key exchange [29],[25] when implemented in practice. This aims to serve as a simple and direct key validation technique for RLWE keys and helps to protect against active attacks in RLWE-based key exchange with malformed keys. The advantage of using this direct validation of RLWE key over the indirect key validation using the Fujisako-Okamoto transformation is the simplicity and efficiency of the protocol and the ability to build more efficient systems and reuse keys in practice. In section 5, we provide a variant of the protocol that does not rely on the ROM for its zero knowledge, by including a commitment scheme. We also remark that when a key p is validated with this protocol, p is verified to be of the form $p = as + e$ with s, e small but does not verify to be from the Discrete Gaussian distribution. But this seems to be sufficient for the applications of our protocol since a malicious user does not have significant gain over choosing small s, e from a distribution that is different from the Discrete Gaussian. This protocol is also more simple and efficient than the cut and choose protocol in [20] that requires many RLWE samples generated and all but one of the samples revealed to the verifier, resulting in increased computation and communication complexity. Our protocol is efficient since the operations involved are only multiplication of ring elements and this idea can be used to design a new key exchange protocol that allows key reuse. The soundness of our protocol also directly relies on the hardness of the RLWE problem. Although the main motivation of our protocol is in the context of key exchange, we note that this can

be applied to RLWE verifiable encryption and can also be used to obtain efficient signatures using the Fiat Shamir transform.

2 Preliminaries

The Learning with Errors (LWE) problem was first introduced by Regev in 2005 along with a quantum reduction from solving hard lattice problems in the worst case to solving LWE in the average case. Since the introduction of LWE and the reduction, it has allowed the design of many versatile applications in Key Exchange, Digital Signatures , FHE (Fully Homomorphic Encryption) schemes [15], Identity Based Encryption (IBE) schemes [2] and more. LWE is a generalization of the parity-learning problem in computer science. In 2009, Peikert showed a classical reduction from variants of the shortest vector problem to corresponding versions of LWE [53]. The decision version of the LWE problem is to distinguish the following two distributions given $poly(n)$ samples: (**a**, **a**.**s** + e) and (**a**, **b**), where **a**, **s**, **b** $\in \mathbb{Z}_q^n$ uniformly at random and e is sampled from an error distribution on $\mathbb{Z}_q$. The search version of the problem is to solve for **s** given (**a**, **a**.**s** + e). The Ring-LWE problem was designed by Lyubashevsky et al. in [47] where the efficiency limitations of LWE are overcome by defining the cryptographic operations over a ring $R = \mathbb{Z}[x]/\phi(x)$, (analogously $R_q = \mathbb{Z}_q[x]/\phi(x)$) where $\phi(x)$ is the m-th cyclotomic polynomial. In practice, we often use the 2-power cyclotomics for efficiency and controlling the noise. By considering polynomials from the ring R_q, one sample $a, a.s+e$ by choosing a, s uniformly from R_q and e from an error distribution of R_q provides n LWE samples corresponding to each coefficient of the polynomial.

We provide the definition of the Discrete Gaussian distribution (error distribution) here:

Discrete Gaussian Distribution

Definition 1. *[64] For any positive real $\alpha \in \mathbb{R}$, and vectors $c \in \mathbb{R}^n$, the continuous Gaussian distribution over $\mathbb{R}^n$ with parameter α centered at c is defined by the probability function $\rho_{\alpha,c}(x) = (\frac{1}{\alpha})^n exp(\frac{-\pi\|x-c\|^2}{\alpha^2})$. For integer vectors $c \in \mathbb{R}^n$, let $\rho_{\alpha,c}(\mathbb{Z}^n) = \sum_{x\in\mathbb{Z}^n} \rho_{\alpha,c}(x)$. Then, we define the discrete Gaussian distribution over $\mathbb{Z}^n$ as $D_{\mathbb{Z}^n,\alpha,c}(x) = \frac{\rho_{\alpha,c}(x)}{\rho_{\alpha,c}(\mathbb{Z}^n)}$, where $x \in \mathbb{Z}^n$. The subscripts α and c are taken to be 1 and 0 (respectively) when omitted.*

For a vector $v = (v_0, \ldots, v_{n-1})$ in $\mathbb{R}^n$ or $\mathbb{C}^n$ and $p \in [1, \infty)$, we define the ℓ_p norm as $\|v\|_p = (\sum_{i=0}^{n-1} |v_i|^p)^{1/p}$ and the ℓ_∞ norm as $\|v\|_\infty = max_{i\in[n]}|v_i|$. The ℓ_2 norm corresponds to the ℓ_p norm with $p = 2$ and is denoted as $\|.\|$ in this paper. In applying the norms, we assume the coefficient embedding of elements from R to $\mathbb{R}^n$. For any element $s = \sum_{i=0}^{n-1} s_i x^i$ of R, we can embed this element into $\mathbb{R}^n$ as the vector $(s_0, \ldots s_{n-1})$.

We recall two useful lemmas here:

Lemma 1 ([64]). *Let $f(x)$ and R be defined as above. Then, for any $s, t \in R$, we have $\|s \cdot t\| \leq \sqrt{n} \cdot \|s\| \cdot \|t\|$ and $\|s \cdot t\|_\infty \leq n \cdot \|s\|_\infty \cdot \|t\|_\infty$.*

Lemma 2 ([48, 30]). *For any real number $\alpha = \omega(\sqrt{\log n})$, we have $\Pr_{\boldsymbol{x}\leftarrow\chi_\alpha}[\|\boldsymbol{x}\| > \alpha\sqrt{n}] \leq 2^{-n+1}$.*

The Hermite Normal Form (HNF)-LWE is a reduction of the LWE problem in which the secret **s** is also derived from the error distribution. It has also been shown that solving HNF-LWE problem has the same reduction to solving hard lattice problems in the worst case [4] and hence can be safely used in cryptographic applications.

Let $s \leftarrow R_q$ be a uniformly chosen element of the ring R_q, as defined above. We define A_{s,χ_α} to be the distribution of the pair $(a, as + e) \in R_q \times R_q$, where $a \leftarrow R_q$ is uniformly chosen and $e \leftarrow \chi_\alpha$ is independent of a.

Definition 2 (Ring-LWE Assumption). *Let R_q, χ_α be defined as above, and let $s \leftarrow R_q$ be uniformly chosen. The (special case) ring-LWE assumption $RLWE_{q,\alpha}$ states that it is hard for any PPT (Probabilistic Polynomial Time)algorithm to distinguish A_{s,χ_α} from the uniform distribution on $R_q \times R_q$ with only polynomially many samples.*

The search version of RLWE is to modify the above definition by requiring the PPT algorithm to find s rather than distinguish the two distributions. For certain parameter choices, the two forms are polynomially equivalent [47].

Proposition 1 ([47]). *Let n be a power of 2, let α be a real number in $(0, 1)$, and q a prime such that $q \bmod 2n = 1$ and $\alpha q > \omega(\sqrt{\log n})$. Define $R = \mathbb{Z}[x]/\langle x^n + 1\rangle$ as above. Then there exists a polynomial time quantum reduction from $\tilde{O}(\sqrt{n}/\alpha)$-SIVP (Short Independent Vectors Problem) in the worst case to average-case RLWE$_{q,\beta}$ with ℓ samples, where $\beta = \alpha q \cdot (n\ell/\log(n\ell))^{1/4}$.*

It has been proven that the RLWE assumption still holds even if the secret s is chosen according to the error distribution χ_α rather than uniformly [4, 47]. This variant is known as the *normal form*, and is preferable for controlling the size of the error term [16, 14].

2.1 Zero Knowledge definitions

We focus here on the interactive proof system with zero knowledge. Let P and V denote the honest prover and verifier respectively. P^* and V^* are usually used to denote the cheating prover and verifier respectively.

Definition 3. *[35] (P, V) is an Interactive Proof System (IPS) for a language L, and security parameter k if:*

1. Completeness *: $\forall x \in L$, $Pr[Out_V(P \leftrightarrow V)[x] = accept] \geq 1 - negl(k)$.*
2. Soundness *: $\forall x \notin L$, $\forall P^*$, $Pr[Out_V(P^* \leftrightarrow V)[x] = accept] \leq negl(k)$.*

Here negl() denotes a negligible function and $Out_V(P \leftrightarrow V)[x]$ is the final output (accept/reject) of the verifier on interaction with the prover.

The Statistical distance is defined for two random variables X, Y of a discrete distribution as

$$\Delta(X, Y) = \frac{1}{2} \sum_x |Pr[X = x] - Pr[Y = x]|$$

Let $\mathcal{X} = \{X_n\}_{n\in\mathbb{N}}$ and $\mathcal{Y} = \{Y_n\}_{n\in\mathbb{N}}$ be sequences of probability distributions, called ensembles. We say that $\mathcal{X}$ and $\mathcal{Y}$ are statistically indistinguishable, if $\Delta(X_n, Y_n) = negl(n)$.

An interactive proof system needs to satisfy an additional property to be considered zero knowledge.

Definition 4. *[6] An interactive proof system (P, V) for a language L is (statistical) zero-knowledge (SZK) if for any PPT verifier V^*, there exists an expected PPT simulator S such that for every valid public input x and private input w, the statistical distance of the following two random variables is negligible:*

- *$View_{V^*} < P_{x,w}, V^* >$*
- *$S(x)$. (Note that S can be probabilistic and so this is a random variable).*

That is, S only gets the public input and has no interaction with P, but still manages to output something indistinguishable from whatever V^* learned in the interaction. Here, the notation $View_A < B, A >$ denotes the view of A during the interaction with B: all the messages sent and received.

An NP-relation $R \subseteq \{0,1\}^* \times \{0,1\}^*$ is given by a deterministic algorithm $W(,)$ that runs in time polynomial in the length of its first input. The relation is $R = \{(x,w) : W(x,w)$ accepts$\}$. The associated NP-language $L_R = \{x : \exists w$ such that $W(x,w)$ accepts$\}$. The witness set for an $x \in \{0,1\}^*$ is $R(x) = \{w : W(x,w) = 1\}$.

2.2 Notation

Let the integer n be a power of 2, and define $f(x) = x^n + 1$ and consider the ring $R \coloneqq \mathbb{Z}[x]/\langle f(x)\rangle$. For any positive integer q, we define the ring $R_q \coloneqq \mathbb{Z}_q[x]/\langle f(x)\rangle$ analogously, where the ring of polynomials over $\mathbb{Z}$ (respectively $\mathbb{Z}_q \coloneqq \mathbb{Z}/q\mathbb{Z}$) we denote by $\mathbb{Z}[x]$ (respectively $\mathbb{Z}_q[x]$). For any polynomial $y(x)$ in R (or R_q), we identify y with its coefficient vector in $\mathbb{Z}^n$ (or $\mathbb{Z}_q^n$). Let χ_α denote the discrete Gaussian distribution on R_q with parameter α. The norm of y is computed as the norm of its coefficient vector. We say a function $f(\kappa)$ is negligible if for every $c>0$, there exists an $N \in \mathbb{N}$ such that $f(\kappa)<1/\kappa^c$ for all $\kappa>N$. We use $negl(\kappa)$ to denote a negligible function of κ, and we say a probability is overwhelming if it is $1 - negl(\kappa)$. Let $H : R_q \to \{0,1\}^\tau$ be a hash function where τ is the bit length of the output.

2.3 Pseudorandomness

Lemma 3. *If x_1 is sampled from the uniform distribution on R_q, then $\bar{x} = x_1 + x$ is uniform over R_q even if x follows an arbitrary probability distribution.*

Proof. Refer to Appendix A □

We can show that $\bar{x}$ is statistically pseudorandom assuming that x_1 is statistically pseudorandom and x follows arbitrary distribution, when x, x_1 are distributed independently. For more details, refer to Appendix. Statistical indistinguishability implies computational indistinguishability but it can also be shown directly here as follows.

Lemma 4. *$\bar{x} = x_1 + x$ is almost uniform (computationally pseudorandom) in R_q if x_1 is computationally pseudorandom in R_q and x follows an arbitrary probability distribution ϕ.*

Proof. Refer to Appendix B □

3 Random Oracle Model (ROM)

The random oracle model was introduced by Bellare and Rogaway in 1993 to bridge the gap between theoretical security and practical cryptography. Protocols using ROM to prove security, model hash functions as public random oracles [9]. This allows for more efficient protocols with security proofs, compared to the standard model (referring to protocols not using random oracles or Common Reference Strings (CRS)). The approach of ROM as described in their work is as follows:

Let $\prod$ be a protocol problem. Suppose $\mathcal{P}$ is a protocol for problem $\prod$.

1. Define problem $\prod$ under the computational model that allows all parties in the protocol to access a random public oracle $\mathcal{R}$.
2. Devise an efficient protocol $\mathcal{P}$ for $\prod$ in this random oracle model.
3. Prove that $\mathcal{P}$ satisfies the definition for $\prod$.
4. Replace oracle accesses to $\mathcal{R}$ by computation of a hash function h.

There has been a lot of serious discussion about the actual security of protocols using ROM. Some of the related work include [40, 39] in favor of ROM and [8, 19] against its use. But it has been widely used for building efficient protocols with provable security. We also present a variant of our protocol without using the random oracle model. With respect to Zero knowledge protocols, the random oracle model changes the deniability property of these protocols. Deniability is the property that guarantees that the transcript of the zero knowledge proof does not leave any evidence of the interaction. Thus, Zero knowledge protocols that does not have the deniability property may no longer be zero knowledge under sequential execution. This stems from the fact that the simulator is allowed to choose the random oracle and can reuse the output, thus violating the assumption that a verifier interacting with the prover learns nothing more than his interaction with the simulator. However, we can still achieve deniable zero knowledge proofs by adopting a weaker simulator, that is no longer allowed to choose the random oracle, but should be able to perform the simulation for all but a negligible fraction of random oracles. We adopt this same approach for our protocol in this work in restricting the simulator. For more details, please refer to [52], [62].

Definition 5. *An interactive proof system (P, V) for a language L is (statistical) deniable zero-knowledge in the random oracle model if for any PPT verifier V^*, there exists an expected PPT simulator S such that for every valid public input x and private input w, the statistical distance of the following two ensembles is negligible:*

- $\{RO, View_{V^{*RO}} < P^{RO}_{x,w}, V^{*RO} >\}$
- $\{RO, S^{RO}(x)\}$. *(Note that S can be probabilistic and so this is a random variable).*

Here, RO is a uniformly distributed random variable corresponding to the random oracle. By including RO variable in the ensemble, it is possible to verify that the simulator uses the pre-specified public random oracle for its execution.

4 The Protocol

4.1 Signal Function

Given $\mathbb{Z}_q = \{-\frac{q-1}{2}, \ldots, \frac{q-1}{2}\}$ and the middle subset $E \coloneqq \{-\lfloor \frac{q}{4} \rfloor, \ldots, \lfloor \frac{q}{4} \rceil\}$, we define $\widehat{Sig}$ as the characteristic function of the *complement* of E: $\widehat{Sig}(v) = 0$ if $v \in E$ and 1 otherwise.

In order to avoid any bias in the Sig function, we use the randomized signal function in the following way. We define two functions $\widehat{Sig_0}, \widehat{Sig_1} : \mathbb{Z}_q \to \{0, 1\}$ as follows:

$$\widehat{Sig_0}(v) = \begin{cases} 0 & v \in [-\lfloor \frac{q}{4} \rfloor, \lfloor \frac{q}{4} \rceil], \\ 1 & \text{otherwise.} \end{cases}$$

$$\widehat{Sig_1}(v) = \begin{cases} 0 & v \in [-\lfloor \frac{q}{4} \rfloor + 1, \lfloor \frac{q}{4} \rceil + 1], \\ 1 & \text{otherwise.} \end{cases}$$

Note that $\widehat{Sig_0}$ is defined to be the same function as $\widehat{Sig}$ above, while $\widehat{Sig_1}$ is a slightly shifted variant. By definition, Sig_0 causes a bias in the final shared key derived. This bias is balanced by the function Sig_1. Thus, we combine these into a randomized function $\widehat{Sig_*}$, where $\widehat{Sig_*}(v)$ is found by randomly sampling $b \leftarrow \{0, 1\}$ and returning $\widehat{Sig_b}(v)$.

Now, we can extend the function $\widehat{Sig_*}$ to R_q by applying the function coordinate-wise. For ring element $v = (v_0, v_1, \ldots, v_{n-1}) \in R_q$, we define the function $Sig_* : R_q \to \{0, 1\}^n$, $Sig_*(v) = (\widehat{Sig_*}(v_0), \widehat{Sig_*}(v_1), \ldots, \widehat{Sig_*}(v_{n-1}))$.

4.2 The Protocol Itself

As a first step in the protocol, P chooses $a \leftarrow R_q$ uniformly and $s, e \leftarrow \chi_\alpha$ and computes $p = as + e$. P then publishes (a, p) as its public key. P also publishes its parameter choice n, q, α. The secret of P is s and the protocol lets P prove to a verifier that p is of the form $as+e$ and that P knows s corresponding to p published. This set up is assuming that P is honest. If P is malicious, then p need not be of the form $p = as + e$ and the verifier accepts the prover in the protocol with only negligible probability. Let M denote the set $\{-\lfloor \frac{q}{8} \rfloor, \ldots, \lfloor \frac{q}{8} \rceil\} \cup \{-\lfloor \frac{3q}{8} \rfloor, \ldots, \lfloor \frac{3q}{8} \rceil\}$ (refer Figure 1).

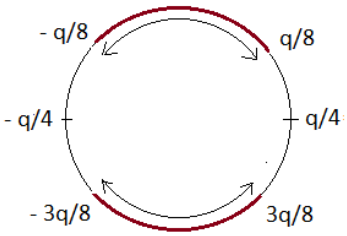

Fig. 1. The region M used for verifying the signal sent by the prover.

The interactive zero knowledge protocol is described as follows:
Let $H_1 : R_q \rightarrow \chi_\alpha$ be a hash function modeled as a random oracle $\mathcal{H}_1$.

P : P computes $p_1 = as_1 + e_1$ where $s_1, e_1 \leftarrow \chi_\alpha$ and reveals p_1 to the verifier V.

V : On receiving p_1 from P, the verifier now generates an RLWE sample x as $x = as' + e'$ where $s', e' \leftarrow \chi_\alpha$ and randomly chooses a challenge bit $b \in \{-1, 1\}$. V sends b, x to the prover P.

P : In order to complete the proof, the prover computes $s'_1 = H_1(x)$. Prover sets $\bar{x} = as'_1 + e'_1 + x$, where $e'_1 \leftarrow \chi_\alpha$. Then computes $k_p = (s_1 + bs)(\bar{x}) + g_p$ where $g_p \leftarrow \chi_{\sqrt{2}\alpha}$ and $\sigma = Sig(k_p)$. P sends σ to V. Here, the error g_p is sampled from error distribution with parameter $\sqrt{2}\alpha$ to have the same standard deviation as the secret $s' + s'_1$. Since each secret s' and s'_1 are sampled from the error distribution (Discrete Gaussian) with parameter α, the sum follows the same error distribution with parameter $\sqrt{2}\alpha$.

V : The verifier V computes $s'_1 = H_1(x)$ and $k_v = (s'_1 + s')(p_1 + bp) + g_v$ where $g_v \leftarrow \chi_{\sqrt{2}\alpha}$. The proof is accepted by V if the signal σ is verified to be correct using the computed value k_v and rejected if the signal is incorrect. The verifier only checks the indexes i of σ for which $k_v[i] \in M$. If $k_v[i] \in \{-\lfloor \frac{q}{8} \rfloor, \ldots, \lfloor \frac{q}{8} \rceil\}$, then $\sigma[i]$ is expected to be 0 and if $k_v[i] \in \{-\lfloor \frac{3q}{8} \rfloor, \ldots, \lfloor \frac{3q}{8} \rceil\}$, $\sigma[i]$ is expected to be 1. Since for any $d \in \mathbb{Z}_q$, $Pr(d \in M) = 1/2$, there are enough number of $\sigma[i]$ for the verifier to validate the correctness.

The idea behind using $\bar{x}$ for computing k_p is to avoid leaking information about s through the signal to a malicious verifier V^* since it uses information from both the prover and verifier to perform the computation. If V^* is malicious and chooses an x deviating from the protocol, by using $\bar{x}$, we ensure k_p has the form of an RLWE sample, which is indistinguishable from uniform samples and so P does not leak information by sending the signal σ. The soundness and zero knowledge property for this protocol are proven in the random oracle model with

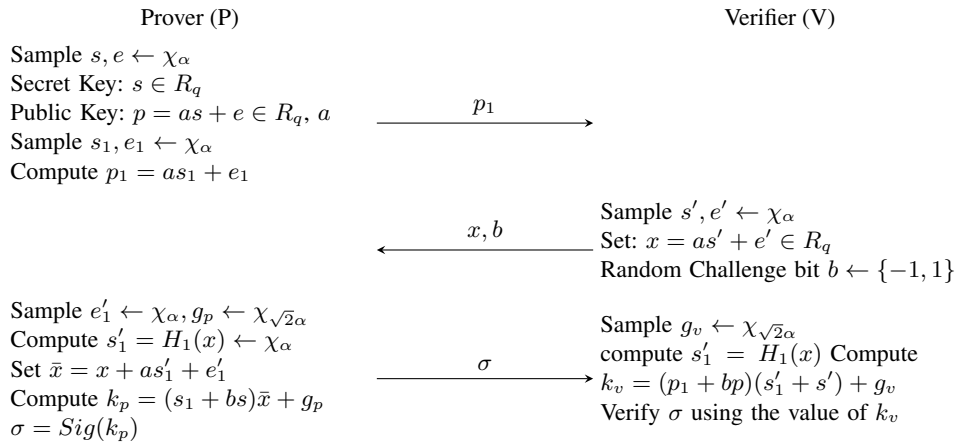

Fig. 2. Authentication Protocol

$H_1(x)$ modeled as a random oracle. Random oracle queries are answered in the usual way: new queries are answered with uniformly random values, and previously made queries are answered identically to the past response.

Lemma 5. *Let q be an odd prime such that q = 1 mod $2n$ and $q > 80\alpha^2 n^{3/2}$. Then the probability of an honest verifier accepting a signal σ from an honest prover P is at least $1 - 2^{-n+1}$.*

Proof. We see the following relation between k_v and k_p

$$\begin{aligned} k_p - k_v &= (s_1 + bs)\bar{x} + g_p - ((p_1 + bp)(s'_1 + s') + g_v) \\ &= as'_1 s_1 + bass'_1 + as' s_1 + bas' s + bse' + e' s_1 + e'_1 s_1 + bse'_1 + g_p \\ &- (as_1 s'_1 + bass'_1 + as' s_1 + bas' s + bes'_1 + bes' + e_1 s'_1 + e_1 s' + g_v) \\ &= e' s_1 + be' s + e'_1 s_1 + bse'_1 - bes'_1 - bes' - e_1 s'_1 - e_1 s' + g_p - g_v = e_k \end{aligned}$$

Here, e_k is used to denote the value $e's_1 + be's + e'_1 s_1 + bse'_1 - bes'_1 - bes' - e_1 s'_1 - e_1 s' + g_p - g_v$. Since the challenge bit b is either -1 or 1, we have $|b| = 1$. So, $\|e_k\| \leq \|e's_1\| + \|e's\| + \|e'_1 s_1\| + \|se'_1\| + \|es'_1\| + \|es'\| + \|e_1 s'_1\| + \|e_1 s'\| + \|g_p\| + \|g_v\|$. Then, using Lemma 2, each error term has norm less than $\alpha\sqrt{n}$ with overwhelming probability. Combining this with lemma 1, we get $|e_k| \leq 10\alpha^2 n^{3/2} < q/8$ with probability $1 - 2^{-n+1}$.

Since we know that $\|e_k\|_\infty \leq \|e_k\|$, we have $\|e_k\|_\infty < q/8$ with the same probability. So, we have $e_k[i] < q/8$, for each i in $0, \ldots n-1$. Then from the analysis above, we can see that if the verifier's computation $k_v[i]$ for the i^{th} coefficient is in M, then $\sigma[i]$ is accepted using the value of $k_v[i]$ with probability $1 - 2^{-n+1}$. □

We now prove that the protocol described in Figure 2 satisfies the following properties of a zero knowledge protocol.

Completeness: Let q be an odd prime such that q = 1 mod $2n$ and $q > 80\alpha^2 n^{3/2}$. For every honest P with knowledge of s for an RLWE sample $p = as + e$, $Pr[Out_V(P \leftrightarrow V) = accept] \geq 1 - 2^{-n+1}$.

Proof. This follows from Lemma 5. □

Computational Soundness: [1] Computational Soundness means that if a PPT malicious prover P^* does not know the secret s corresponding to p, then regardless of what P^* does, the verifier will accept the proof with probability at most $1/2 + negl(n)$. This is equivalent to saying that the advantage of P^* in making the proof is negligible with advantage defined as follows:

$$\mathcal{A}dv_{P^*} = Pr[Out_V(P \leftrightarrow V) = accept | b' \leftarrow \{0,1\} : b' \neq b] - 1/2$$

Proof. The soundness of the protocol is proven in the random oracle model. If P^* can guess the challenge bit of the verifier correctly, then P^* can send $p_1 + p$ (resp. $p_1 - p$) instead of an honest p_1 if the guess of the challenge bit is -1 (resp. 1). This is explained and used in the construction of the simulator in the zero knowledge part. Since P^* can only guess this challenge bit with probability $1/2$, the probability of success for P^* is at least $1/2$. Assuming P^* to be a malicious prover that can make a valid proof without the knowledge of s with non-negligible advantage, we consider P^* as an oracle that gives valid proof with non-negligible advantage, irrespective of whether P^* knows the corresponding s or not, when p is of the form $p = as + e$, with s, e small.

The signal σ sent by a prover is accepted by an honest verifier when the noise terms are small so that the difference between the verifier's computation k_v and the prover's computation k_p is small (the error bounded by $q/8$ in the ℓ_2 norm). Thus, P^* gives valid proof for RLWE samples with non-negligible advantage since RLWE samples are of the form $p = as + e$ with s, e small, usually sampled from the discrete gaussian distribution. We define games $Game_0$ through $Game_2$ that can played between a cheating prover P^* and a verifier and use them to show that if P^* can make a valid proof with success probability $1/2 + \mathcal{A}dv_{P^*}$ with $\mathcal{A}dv_{P^*}$ non-negligible, then P^* can be used to build an RLWE distinguisher with the same advantage as P^*. On interaction with the prover, the verifier's acceptance decision can be regarded as the decision bit 0 or 1 for rejection or acceptance respectively. Every game is played between the prover strategy P^* and a verifier oracle that honestly performs the verifier side of the protocol.

$Game_0$ This is the original protocol in which p has the form $p = as + e$ and $p_1 = as_1 + e_1$ according to the protocol. The transcript of the protocol is generated as p_1, x, b, σ and output is the decision bit b_d of the verifier. This decision bit b_d is 1 if the verifier accepts the proof and 0 if he rejects.

$Game_1$: This game is identical to $Game_0$ except that p is chosen uniformly over R_q.
$Game_2$: The game is identical to $Game_1$ except that x is chosen uniformly.

Lemma 6. *If P^* can make a valid proof for an RLWE sample $p = as + e$ with $s, e \leftarrow \chi_\alpha$ without the knowledge of s to an honest verifier V with non-negligible advantage, then there is a distinguisher for RLWE samples from uniform with the same advantage.*

Proof. Let $\mathcal{B}$ be the RLWE distinguisher that we will construct using the prover strategy P^*. On receiving the challenge (a, p) from the RLWE challenger, $\mathcal{B}$ sends the challenge to P^*. Here, $\mathcal{B}$ performs the role of the honest verifier. If p is an RLWE sample $p = as + e$, the interaction between P^* and $\mathcal{B}$ is exactly $Game_0$ and $\mathcal{B}$ then outputs 1 since by assumption, P^* is capable of making valid proofs for any RLWE sample p. If p is uniform, then the interaction between P^* and $\mathcal{B}$ is exactly $Game_1$ (Assuming that P^* cannot make valid proofs for uniform p except with negligible advantage, which we will prove later). Thus, an output of 1 concludes that p is RLWE and the game played is $game_0$; an output of 0 concludes that p is uniform and the game played is $Game_1$.

[1] Computationally sound interactive proofs are also referred to as Interactive argument systems [17]. We restrict to a PPT prover since the soundness of our protocol relies on the hardness of RLWE.

Suppose P^* can also make valid proofs for uniform p with non-negligible advantage, then the distinguisher as defined above would not be successful. We claim that this cannot be the case unless P^* can solve the search version of the RLWE problem with the same advantage. Suppose P^* can generate a signal σ that is accepted by the verifier for any p without the knowledge of corresponding s. Here, the key computation of the verifier is $k_v = (p_1+bp)(s'+s'_1)+g_v = (p_1+bp)s'+(p_1+bp)s'_1+g_v$, in which $(p_1+bp)s'_1$ can be computed by the malicious prover himself. Also, since the verifier accepts the signal σ, it implies that P^* is able to generate the signal for $(p_1 + bp)(s' + s'_1) + e_p$ for any p, where $e_p = g_v - e_k$ and $e_k = k_v - k_p$ from lemma 5. Now, we recall the signal function attack in [25]. This attack allows an adversary to initiate multiple key exchange sessions with a party with a fixed public key $p = as + 2e$ and gain information about the secret from the signal by creating malformed public keys of the form $p_{\mathcal{A}} = as_{\mathcal{A}} + k$, where k ranges through all values of $\mathbb{Z}_q$. The number of signal changes as k varies over values of $\mathbb{Z}_q$ is exactly $2s[i]$ for every coefficient i of the secret s.
Setup: If P^* can generate the signal of $(p_1 + bp)(s' + s'_1) + e_p$, then he can initiate multiple key exchange sessions such that he plays both the initiator and the responder roles of the KE protocol. Let P_I^* and P_R^* denote the initiator and responder role played by P^* respectively. P_R^* queries the random oracle $\mathcal{H}_1$ with input x to obtain s'_1 and fixes his public key as $x + as'_1 = a(s' + s'_1) + e'$. Note that this is possible since a, x are known to P^*. P_I^* creates his public key as $p_I = p_1 + bp$ where p_1 is malformed as $p_1 = as_1 + ke_1$, so $p_I = as_1 + k + bp$ with $e_1 = 1 \in R_q$ and initiates a key exchange session with P_R^*. Since P_R^* is capable of generating the signal of $(p_I s' + e_p)$, he responds with x, σ_R, where $\sigma_R = Sig(p_I(s' + s'_1) + e_p) = Sig((p_1 + bp)(s' + s'_1) + e_p) = Sig(k(s' + s'_1) + as_1(s' + s'_1) + bps' + e_p)$. The initiator P_I^* can choose to drop the session at this stage or continue to create an unsuccessful KE session. P_I^* repeats this process q times varying k through 0 to $q - 1$ and recovers $s' + s'_1$ as described in the attack in [25] by looking at the number of signal changes for each coefficient. The process is repeated for varying k fixing $b = 1$ or -1 and P^* can compute σ_R for both values of b. Here, $as_1(s' + s'_1) + bps'$ is fixed for each coefficient across all q queries for varying k if we fix $b = 1$ or -1. Also, since P^* can obtain s'_1 from x by querying the random oracle $\mathcal{H}_1$, he can recover s' from $s' + s'_1$. Thus, given an RLWE sample $x = as' + e'$, P^* can solve for s', thus solving the search version of the RLWE problem with the same advantage that he has in generating the signal corresponding to a uniform p that is accepted by a verifier in the authentication protocol.

□

Note that if both p and the verifier's choice of x is uniform, then the verification of the signal is no longer valid. Thus, $Game_2$ is never played between the cheating prover and the verifier in the course of the proof. □

This shows that the soundness error of the protocol is $1/2 + negl(n)$. We can use the trick in [31], explained in [59], page 69 to slightly modify the protocol to obtain a soundness error of $1/2$. The soundness error becomes negligible as the protocol is executed n times.

Zero knowledge: We first show that the proposed protocol is Honest Verifier Statistical Zero knowledge (HVSZK) and construct a probabilistic polynomial time simulator $\mathcal{S}$ that has black box oracle access to an honest verifier V and outputs transcripts that are statistically close to the transcripts between the real prover and verifier. We also assume that the simulator has access to the random oracle $\mathcal{H}_1$, proving zero knowledge in the random oracle model. Later, using [22], we transform the HVSZK into a general SZK protocol. Let $View_{V^{RO}} < P_{p,s}^{RO}, V^{RO} >$ and $S^{RO}(p)$ denote the view of the honest verifier V in an interaction with a real prover P and the simulator's output respectively, with access to the random oracle $\mathcal{H}_1$.

The idea behind the construction of a simulator is to show that a verifier can simulate the interaction it has with a real prover by itself, thereby establishing that the prover does not leak any useful information by performing a proof with the verifier. The simulator $\mathcal{S}$ for our authentication protocol described in Figure 2 is constructed as follows:

1. Input: a, p such that $a \leftarrow R_q$ and $p = as + e$ where $s, e \leftarrow \chi_\alpha$. However, the simulator does not get the value of s, e.
2. Choose $b' \leftarrow \{-1, 1\}$.
3. Choose $\hat{p_1}$ as $\hat{p_1} = as_1 + e_1$ where $s_1, e_1 \leftarrow \chi_\alpha$. Set $p_1 = \hat{p_1} - b'p$.
4. Invoke V from the state in step 2 on the message p_1 to obtain a challenge bit b and x.
5. If $b = b'$, query the random oracle $\mathcal{H}_1$ with input x to obtain s'_1 and output $\sigma = Sig((s_1)(as'_1 + e'_1 + x)$, otherwise go to step 2.

In the construction, the simulator throws away the instances when the predicted random challenge bit of the simulator does not match the random challenge bit received by invoking the verifier oracle. When $b = b' = 1$, the verifier's computation $k_v = (s'_1 + s')(p_1 + p) = (s'_1 + s')(\hat{p_1})$, and k_v still has $<q/8$ difference from k_p, preserving the acceptance probability and hence the distribution that V views. The same applies to the case when $b = b' = -1$. Here, the message p_1 is an RLWE sample irrespective of whether $b' = 1$ or $b' = -1$. So, the distribution is the same as in the case of the interaction with a real prover, regardless of whether $b' = 1$ or $b' = -1$. Also, the challenge bit b and x are independent of b' and follow the same distribution as in the protocol with $Pr[b = b'] = 1/2$. The difference here comes from the final step of the simulator as we repeat the protocol when the challenge bits do not match. If we set the simulator to stop after n tries to match b, b' and output a random uniform σ after that, to obtain a strict polynomial time simulator, then the distribution of the simulator is at a statistical distance of at most 2^{-n} from the distribution of the real interaction.

In order to prove zero knowledge with respect to any arbitrary verifier (including cheating verifiers that deviate from the protocol), we use the results from section 2.3 on $\bar{x}$ in the protocol. We observe that irrespective of the distribution of x, $\bar{x}$ is pseudorandom in R_q due to the addition of $as'_1 + e'_1$ which is an RLWE sample and is pseudorandom in R_q. So, k_p has the RLWE form $k_p = \bar{x}(s_1 + bs) + g_p$ since $s_1 + bs, g_p \leftarrow \chi_{\sqrt{2}\alpha}$ and so is pseudorandom.

Thus, the key k_p computed by P has the RLWE form and is pseudorandom, resulting in a uniform signal σ that does not leak information about s to an arbitrary cheating verifier by running the protocol. To ensure zero knowledge and to provide a more rigorous analysis, we can use the second transform in [22] to adapt the above simulator to one for SZK with an arbitrary verifier that preserves negligible completeness and soundness error of the protocol. The transform ensures that the challenge bit b from the verifier is uniform using a commitment protocol. For x sent by the verifier, we don't require this since the prover's computation of k_p uses $\bar{x}$ as already mentioned.

Even though technically the cheating verifier knows the value of s'_1 and can compute $\bar{x}$ himself, he cannot create an x to exploit this because of the commitment scheme. Thus, even if x is formed deviating from the protocol, the honest prover uses $\bar{x}$ which is indistinguishable from uniform to perform its computations and hence does not leak information about the secret s.

5 Variant of the authentication protocol

We present a variant of the protocol described above with a commitment scheme, but still preserving the zero knowledge and other desirable properties of the protocol and eliminating the need for the Random Oracle Model. The initial phase of the protocol is the commitment phase. A hash function H is used for the commitment between the prover and the verifier. We describe the protocol here (the notations and parameters are the same as the protocol above):

P : P generates an $s'_1 \leftarrow \chi_\alpha$ and sends $com_p = H(s'_1)$ to the verifier V.

V : On receiving $H(s'_1)$ from P, the verifier now generates an RLWE sample x as $x = as' + e'$ where $s', e' \leftarrow \chi_\alpha$. V sends $com_v = H(x)$ to the prover P.

P : P computes $p_1 = as_1 + e_1$ where $s_1, e_1 \leftarrow \chi_\alpha$ and reveals p_1, s'_1 to the verifier V.

V : On receiving s'_1, p_1 from P, the verifier checks if $H(s'_1) = com_p$ else, aborts the protocol. The verifier now randomly chooses a challenge bit $b \in \{-1, 1\}$ and reveals the RLWE sample x as $x = as' + e'$ generated before and the challenge bit b to the prover P.

P : The prover checks if x matches with the hash commitment sent by the verifier in the previous step, $H(x) = com_v$ else aborts the protocol. In order to complete the proof, the prover computes $\bar{x} = x + as'_1 + e'_1$, where $e'_1 \leftarrow \chi_\alpha$ and then computes $k_p = (s_1 + bs)(\bar{x}) + g_p$ where $g_p \leftarrow \chi_{\sqrt{2}\alpha}$ and $\sigma = Sig(k_p)$. P sends σ to V.

V : The verifier V computes $k_v = (s'_1 + s')(p_1 + bp) + g_v$ where $g_v \leftarrow \chi_{\sqrt{2}\alpha}$. The proof is accepted by V if the signal σ is verified to be correct using the computed value k_v and rejected if the signal is incorrect. Similar to the main protocol, the verifier only checks the indexes i of σ for which $k_v[i] \in M$. If $k_v[i] \in \{-\lfloor \frac{q}{8} \rfloor, \ldots, \lfloor \frac{q}{8} \rceil\}$, then $\sigma[i]$ is expected to be 0 and if $k_v[i] \in \{-\lfloor \frac{3q}{8} \rfloor, \ldots, \lfloor \frac{3q}{8} \rceil\}$, $\sigma[i]$ is expected to be 1.

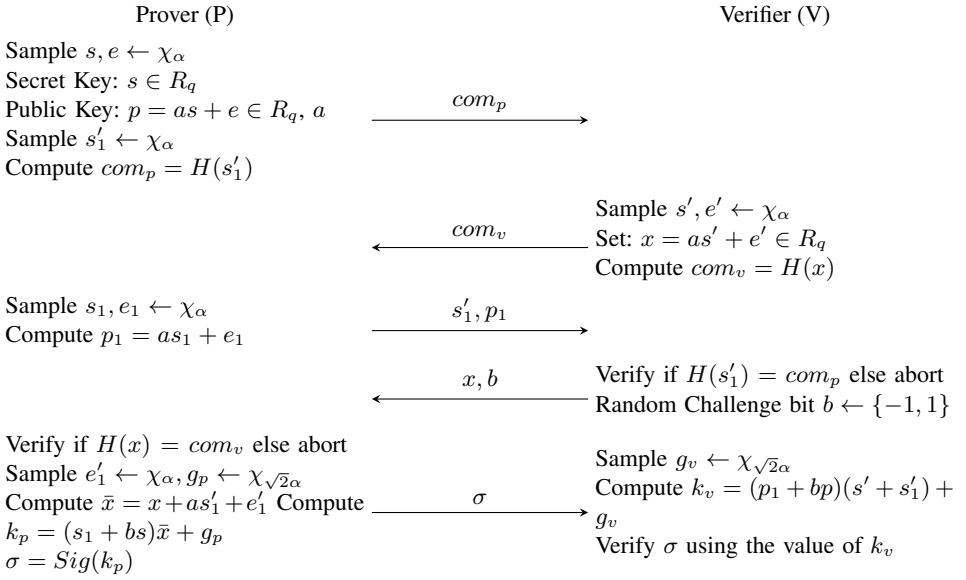

Fig. 3. Authentication Protocol - Variant

The above protocol has the same structure as the protocol in Figure 2 with the difference being that $H_1(x)$ required in the protocol to establish zero knowledge is replaced with a commitment s'_1 that performs the same functionality of randomizing x, thus preserving the zero knowledge in this variant. In order to prove zero knowledge and soundness, we require that the commitment scheme is statistically hiding and computationally binding which ensures the integrity of the commitments. We only require computational binding since we focus on Probabilistic Polynomial Time (PPT) prover and statistical hiding to ensure statistical zero knowledge in the protocol. Forcing the verifier V to commit to a message x before receiving the value of s'_1 from the prover helps with zero knowledge of the protocol, and forcing the prover to commit to an s'_1 before seeing x ensures soundness of the protocol. The com-

pleteness, soundness and zero knowledge properties of this variant can be replicated from the above proofs replacing $H_1(x)$ with s'_1.

5.1 Commitment Scheme:

In order to establish the zero knowledge and soundness properties of our variant protocol (Figure 3), the commitment scheme plays an important role. Note that the commitment of the prover and verifier in the first two message exchanged in the protocol uses standard hash functions on the committed value. The hiding property ensures zero knowledge since otherwise a malicious verifier can gain the knowledge of s'_1 from the commitment and from x accordingly. The binding property restricts the malicious prover to the use of the committed value s'_1 in the proof. The commitment scheme can be separately executed to agree on s'_1 and x values before running the validation part of the protocol and thus can be replaced with a secure commitment scheme that satisfies the required binding and hiding properties. We use Hash function H for the presentation of the commitment scheme for simplicity. Assuming one way functions and collision intractable functions exist, many commitment schemes with the required properties have been proposed [23, 34, 33].

5.2 Communication Complexity:

The communication complexity of a single execution of the first protocol, involves the messages x, p_1, b, σ exchanged between the prover and the verifier. This amounts to $2nlog\ q+n+1$ bits of data exchanged. This does not include the complexity of the messages exchanges when we apply the transform in [22] to derive a uniform challenge bit b. The commitment scheme in the variant presented would add a complexity of $2\tau + O(nlog\ \alpha)$ bits of the data exchanged.

6 Protocol for LWE

We also present a general authentication protocol for LWE form keys here for completeness of this work. Generate public parameters n, q, α. Let $D_{\mathbb{Z}^n,\alpha}$ denote the Discrete Gaussian distribution over $\mathbb{Z}^n$. Let S_* be the signal function defined analogous to Sig as $S_* : Z_q^n \rightarrow \{0,1\}^n$, $S_*(v) = (\widehat{Sig_*}(v_0), \widehat{Sig_*}(v_1), \ldots, \widehat{Sig_*}(v_{n-1}))$. We choose a uniform matrix $A \leftarrow \mathbb{Z}_q^{n\times n}$ instead of the public $a \in R_q$ in the case of the RLWE based protocol. We use the Hermite Normal Form (HNF) - LWE to ensure that the secret and error of $p = As + e$ is small for security and correctness of the protocol. In the HNF-LWE problem, both s, e are sampled according to $D_{\mathbb{Z}^n,\alpha}$. The protocol is described as follows:

P : P generates an $s'_1 \leftarrow D_{\mathbb{Z}^n,\alpha}$ and sends $com_p = H(s'_1)$ to the verifier V.

V : On receiving $H(s'_1)$ from P, the verifier now generates an RLWE sample x as $x = A^T s' + e' \mod q$ where $s', e' \leftarrow D_{\mathbb{Z}^n,\alpha}$. V sends $com_v = H(x)$ to the prover P.

P : P computes $p_1 = As_1 + e_1 \mod q$ where $s_1, e_1 \leftarrow D_{\mathbb{Z}^n,\alpha}$ and reveals p_1, s'_1 to the verifier V.

V : On receiving s'_1, p_1 from P, the verifier checks if $H(s'_1) = com_p$ else, aborts the protocol. The verifier now randomly chooses a challenge bit $b \in \{-1, 1\}$ and reveals the RLWE sample x as $x = A^T s' + e' \mod qs$ generated before and the challenge bit b to the prover P.

P : The prover checks if x matches with the hash commitment sent by the verifier in the previous step, $H(x) = com_v$ else aborts the protocol. In order to complete the proof, the prover computes $\bar{x} = x + A^T s'_1 + e'_1 \mod q$, where $e'_1 \leftarrow D_{\mathbb{Z}^n,\alpha}$ and then computes $k_p = (s_1 + bs)^T \bar{x} + g_p \mod q$ where $g_p \leftarrow D_{\mathbb{Z},\sqrt{2}\alpha}$ and $\sigma = S_*(k_p)$. P sends σ to V.

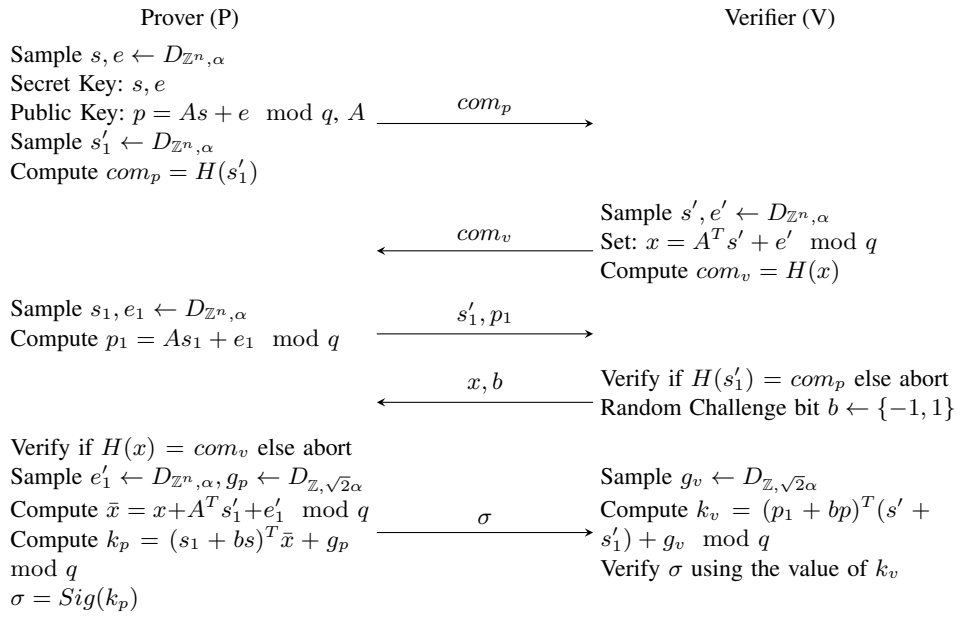

Fig. 4. Authentication Protocol for LWE

V : The verifier V computes $k_v = (p_1 + bp)^T (s_1' + s') + g_v \mod q$ where $g_v \leftarrow D_{\mathbb{Z},\sqrt{2}\alpha}$. The proof is accepted by V if the signal σ is verified to be correct using the computed value k_v and rejected if the signal is incorrect.

The zero knowledge, completeness and soundness can be similarly extended to the LWE case assuming that the HNF-LWE assumption holds and are skipped in this paper.

7 Conclusion

We have introduced a novel application of the Signal function used for reconciliation in key exchange, to derive a secure authentication protocol. The protocol is shown to be zero knowledge with negligible soundness and completeness errors. The security of the protocol is also shown to be directly related to the hardness of solving RLWE problem. We note that the Zero knowledge protocol presented here is against classical verifiers. Zero knowledge against quantum verifiers (honest and dishonest) have been studied in [60], [38], [61]. An open problem with quantum zero knowledge is with the rewinding technique that is used for constructing a simulator. In [21], Damgaård et al. proposed three techniques to transform a HVSZK protocol into a Quantum Zero Knowledge protocol, with one of them in the common reference string model. Wastrous [61] introduced a quantum rewinding technique to achieve ZK against quantum attacks. Recently, the work in [18] shows that every problem in QMA (Quantum Merlin Arthur) has a quantum computational zero knowledge proof, assuming the existence of quantum computationally secure commitment schemes. We leave extending the protocol presented in this paper to quantum zero knowledge as future work. The advantage of this key validation is that it allows us to reuse keys in RLWE-based key exchange. We derive ideas from this to also develop a non interactive proof to validate the keys and we design a new key exchange protocol based on RLWE that utilizes this key validation technique to allow key reuse and consider it as future work that is a consequence of this validation.

References

[1] M. Abdalla, J. H. An, M. Bellare, and C. Namprempre. From identification to signatures via the fiat-shamir transform: Minimizing assumptions for security and forward-security. In *Proceedings of Eurocrypt 2002, volume 2332 of LNCS*, pages 418–433. Springer-Verlag, 2002.

[2] S. Agrawal, D. Boneh, and X. Boyen. Efficient lattice (h)ibe in the standard model. In *Proceedings of the 29th Annual International Conference on Theory and Applications of Cryptographic Techniques*, EUROCRYPT'10, pages 553–572, Berlin, Heidelberg, 2010. Springer-Verlag.

[3] E. Alkim, L. Ducas, T. Pppelmann, and P. Schwabe. Post-quantum key exchange - a new hope. Cryptology ePrint Archive, Report 2015/1092, 2015. http://eprint.iacr.org/2015/1092.

[4] B. Applebaum, D. Cash, C. Peikert, and A. Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In S. Halevi, editor, *Advances in Cryptology – CRYPTO 2009*, volume 5677 of *Lecture Notes in Computer Science*, pages 595–618. Springer Berlin Heidelberg, 2009.

[5] A. Banerjee, C. Peikert, and A. Rosen. *Pseudorandom Functions and Lattices*, pages 719–737. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[6] B. Barak. Zero knowledge proofs, 2007. https://www.cs.princeton.edu/courses/archive/fall07/cos433/lec15.pdf.

[7] C. Baum, I. Damgård, K. G. Larsen, and M. Nielsen. *How to Prove Knowledge of Small Secrets*, pages 478–498. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

[8] M. Bellare, A. Boldyreva, and A. Palacio. *An Uninstantiable Random-Oracle-Model Scheme for a Hybrid-Encryption Problem*, pages 171–188. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[9] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM Conference on Computer and Communications Security*, CCS '93, pages 62–73, New York, NY, USA, 1993. ACM.

[10] F. Benhamouda, J. Camenisch, S. Krenn, V. Lyubashevsky, and G. Neven. *Better Zero-Knowledge Proofs for Lattice Encryption and Their Application to Group Signatures*, pages 551–572. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.

[11] F. Benhamouda, S. Krenn, V. Lyubashevsky, and K. Pietrzak. Efficient zero-knowledge proofs for commitments from learning with errors over rings. In *Proceedings, Part I, of the 20th European Symposium on Computer Security – ESORICS 2015 - Volume 9326*, pages 305–325, New York, NY, USA, 2015. Springer-Verlag New York, Inc.

[12] I. Biehl, B. Meyer, and V. Müller. Differential fault attacks on elliptic curve cryptosystems. In *Proceedings of the 20th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '00, pages 131–146, London, UK, UK, 2000. Springer-Verlag.

[13] J. W. Bos, C. Costello, M. Naehrig, and D. Stebila. Post-quantum key exchange for the tls protocol from the ring learning with errors problem. Cryptology ePrint Archive, Report 2014/599, 2014. http://eprint.iacr.org/2014/599.

[14] Z. Brakerski, C. Gentry, and V. Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 309–325. ACM, 2012.

[15] Z. Brakerski and V. Vaikuntanathan. Fully homomorphic encryption from ring-lwe and security for key dependent messages. In *Proceedings of the 31st Annual Conference on Advances in Cryptology*, CRYPTO'11, pages 505–524, Berlin, Heidelberg, 2011. Springer-Verlag.

[16] Z. Brakerski and V. Vaikuntanathan. Fully homomorphic encryption from ring-lwe and security for key dependent messages. In P. Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 505–524. Springer Berlin Heidelberg, 2011.

[17] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *J. Comput. Syst. Sci.*, 37(2):156–189, Oct. 1988.

[18] A. Broadbent, Z.-F. Ji, F. Song, and J. Watrous. Zero-knowledge proof systems for qma. *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 31–40, 2016.

[19] R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisited. *J. ACM*, 51(4):557–594, July 2004.

[20] E. Crockett and C. Peikert. Challenges for ring-lwe. Cryptology ePrint Archive, Report 2016/782, 2016. http://eprint.iacr.org/2016/782.

[21] I. Damgård, S. Fehr, and L. Salvail. *Zero-Knowledge Proofs and String Commitments Withstanding Quantum Attacks*, pages 254–272. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[22] I. Damgård, O. Goldreich, T. Okamoto, and A. Wigderson. *Honest Verifier vs Dishonest Verifier in Public Coin Zero-Knowledge Proofs*, pages 325–338. Springer Berlin Heidelberg, Berlin, Heidelberg, 1995.

[23] I. B. Damgard, T. P. Pedersen, and B. Pfitzmann. Statistical secrecy and multibit commitments. *IEEE Transactions on Information Theory*, 44(3):1143–1151, May 1998.

[24] I. Damgrd, O. Goldreich, and A. Wigderson. Hashing functions can simplify zero-knowledge protocol design (too). Technical report, BRICS TECHNICAL RERPORT, 1994.

[25] J. Ding, S. Alsayigh, S. RV, S. Fluhrer, and X. Lin. Leakage of signal function with reused keys in rlwe key exchange. Cryptology ePrint Archive, Report 2016/1176, 2016. http://eprint.iacr.org/2016/1176.

[26] J. Ding, X. Xie, and X. Lin. A simple provably secure key exchange scheme based on the learning with errors problem. Cryptology ePrint Archive, Report 2012/688, 2012. http://eprint.iacr.org/.

[27] U. Feige, A. Fiat, and A. Shamir. Zero-knowledge proofs of identity. *Journal of Cryptology*, 1(2):77–94, 1988.

[28] A. Fiat and A. Shamir. *How To Prove Yourself: Practical Solutions to Identification and Signature Problems*, pages 186–194. Springer Berlin Heidelberg, Berlin, Heidelberg, 1987.

[29] S. Fluhrer. Cryptanalysis of ring-lwe based key exchange with key share reuse. Cryptology ePrint Archive, Report 2016/085, 2016. http://eprint.iacr.org/2016/085.

[30] C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th annual ACM symposium on Theory of computing*, STOC '08, pages 197–206, New York, NY, USA, 2008. ACM.

[31] O. Goldreich. Some useful trivial tricks, September 1999. http://www.wisdom.weizmann.ac.il/~oded/tricks.html.

[32] O. Goldreich, A. Sahai, and S. Vadhan. Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, pages 399–408, New York, NY, USA, 1998. ACM.

[33] I. Haitner and O. Reingold. Statistically-hiding commitment from any one-way function. In *Proceedings of the Thirty-ninth Annual ACM Symposium on Theory of Computing*, STOC '07, pages 1–10, New York, NY, USA, 2007. ACM.

[34] S. Halevi and S. Micali. Practical and provably-secure commitment schemes from collision-free hashing. In *Proceedings of the 16th Annual International Cryptology*

Conference on Advances in Cryptology, CRYPTO '96, pages 201–215, London, UK, UK, 1996. Springer-Verlag.

[35] S. Hohenberger. Special topics in theoretical cryptography, 2007. http://www.cs.jhu.edu/~susan/600.641/scribes/lecture2.pdf.

[36] A. Kawachi, K. Tanaka, and K. Xagawa. Concurrently secure identification schemes based on the worst-case hardness of lattice problems, 2008.

[37] D. Kirkwood, B. C. Lackey, J. McVey, M. Motley, J. A. Solinas, and D. Tuller. Failure is not an option: Standardization issues for post-quantum key agreement, 2016. http://csrc.nist.gov/groups/ST/post-quantum-2015/presentations/session7-motley-mark.pdf.

[38] H. Kobayashi. *General Properties of Quantum Zero-Knowledge Proofs*, pages 107–124. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[39] N. Koblitz and A. J. Menezes. Another look at "provable security". *J. Cryptol.*, 20(1):3–37, Jan. 2007.

[40] N. Koblitz and A. J. Menezes. The random oracle model: a twenty-year retrospective. *Des. Codes Cryptography*, 77(2-3):587–610, 2015.

[41] Y. Lindell. How to simulate it - A tutorial on the simulation proof technique. *IACR Cryptology ePrint Archive*, 2016:46, 2016.

[42] S. Ling, K. Nguyen, and H. Wang. *Group Signatures from Lattices: Simpler, Tighter, Shorter, Ring-Based*, pages 427–449. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[43] V. Lyubashevsky. *Lattice-Based Identification Schemes Secure Under Active Attacks*, pages 162–179. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[44] V. Lyubashevsky. *Fiat-Shamir with Aborts: Applications to Lattice and Factoring-Based Signatures*, pages 598–616. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[45] V. Lyubashevsky. Standardizing lattice crypto and beyond, 2017. https://2017.pqcrypto.org/conference/slides/pqc_2017_lattice.pdf.

[46] V. Lyubashevsky and G. Neven. *One-Shot Verifiable Encryption from Lattices*, pages 293–323. Springer International Publishing, Cham, 2017.

[47] V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In H. Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23. Springer Berlin / Heidelberg, 2010.

[48] D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM J. Comput.*, 37:267–302, April 2007.

[49] D. Micciancio and S. P. Vadhan. *Statistical Zero-Knowledge Proofs with Efficient Provers: Lattice Problems and More*, pages 282–298. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[50] NIST. Post quantum cryptography:nist's plan for the future, 2016. http://csrc.nist.gov/groups/ST/post-quantum-crypto/documents/pqcrypto-2016-presentation.pdf.

[51] K. Ohta and T. Okamoto. *On concrete security treatment of signatures derived from identification*, pages 354–369. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[52] R. Pass. *On Deniability in the Common Reference String and Random Oracle Model*, pages 316–337. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[53] C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In *Proceedings of the 41st annual ACM symposium on Theory of computing*, STOC '09, pages 333–342, New York, NY, USA, 2009. ACM.

[54] C. Peikert. *Lattice Cryptography for the Internet*, pages 197–219. Springer International Publishing, Cham, 2014.

[55] C. Peikert and V. Vaikuntanathan. *Noninteractive Statistical Zero-Knowledge Proofs for Lattice Problems*, pages 536–553. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[56] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, STOC '05, pages 84–93, New York, NY, USA, 2005. ACM.

[57] C. R. Shafi Goldwasser, Silvio Micali. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1989.

[58] J. Stern. A new paradigm for public key identification. In *IEEE Transactions on Information Theory*, pages 13–21, 1996.

[59] S. P. Vadhan. *A Study of Statistical Zero-knowledge Proofs*. PhD thesis, Cambridge, MA, USA, 1999. AAI0801528.

[60] J. Watrous. Limits on the power of quantum statistical zero-knowledge. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.*, pages 459–468, 2002.

[61] J. Watrous. Zero-knowledge against quantum attacks. In *Proceedings of the Thirty-eighth Annual ACM Symposium on Theory of Computing*, STOC '06, pages 296–305, New York, NY, USA, 2006. ACM.

[62] H. Wee. *Zero Knowledge in the Random Oracle Model, Revisited*, pages 417–434. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[63] X. Xie, R. Xue, and M. Wang. *Zero Knowledge Proofs from Ring-LWE*, pages 57–73. Springer International Publishing, Cham, 2013.

[64] J. Zhang, Z. Zhang, J. Ding, M. Snook, and Ö. Dagdelen. Authenticated key exchange from ideal lattices. In *Advances in Cryptology-EUROCRYPT 2015*, pages 719–751. Springer, 2015.

A Proof of Lemma 3

Proof. Suppose x has an arbitrary distribution, we show that every coordinate of $\bar{x}$ is uniformly distributed over $\mathbb{Z}_q$. Here, the choice of x and x_1 are independent of each other. Let X_1 denote the random variable with uniform distribution on $\mathbb{Z}_q$ and X_2 denote the random variable with the distribution of x. Then, we show that the probability of $X_1 + X_2$ equals $z \in \mathbb{Z}_q$ is $1/q$, implying that $\bar{x}$ is uniform.

$$\begin{aligned} Pr(X_1 + X_2 = z) &= \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} Pr(X_1 = i) Pr(X_2 = z - i) \\ &= \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} \frac{1}{q} Pr(X_2 = z - i) \\ &= \frac{1}{q} \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} Pr(X_2 = z - i) \\ &= \frac{1}{q} \end{aligned}$$

□

We can show that $\bar{x}$ is statistically pseudorandom assuming that x is statistically pseudorandom. Recall that the statistical distance between two distributions with random variables X, Y is defined as $\Delta(X, Y) = \frac{1}{2} \sum_x |Pr[X = x] - Pr[Y = x]|$. Two distributions are statistically indistinguishable if their statistical distance is negligible. If X_1 has a distribution that is statistically pseudorandom (statistically indistinguishable from uniform) on $\mathbb{Z}_q$, for any

$q_i \in \mathbb{Z}_q$, we have $Pr(X_1 = q_i) = 1/q + \epsilon$, where ϵ is negligible. Let X_2 denote the random variable with the distribution of x. Then, X_1, X_2 are independent random variables. We show that for $z \in \mathbb{Z}_q$, the probability $Pr(X_1 + X_2 = z) \leq \frac{1}{q} + \epsilon$, where ϵ is negligible and X_2 is a random variable of an arbitrary distribution.

$$\begin{aligned} Pr(X_1 + X_2 = z) &= \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} Pr(X_1 = i) Pr(X_2 = z - i) \\ &= \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} (\frac{1}{q} + \epsilon_i) Pr(X_2 = z - i) \\ &= \frac{1}{q} \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} Pr(X_2 = z - i) + \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} \epsilon_i Pr(X_2 = z - i) \\ &< \frac{1}{q} + \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} \epsilon_i \end{aligned}$$

So, we have $Pr(X_1 + X_2 = z) < \frac{1}{q} + \epsilon$, where $\epsilon = \sum_{i=-\frac{q-1}{2}}^{\frac{q-1}{2}} \epsilon_i$ is negligible. Here, we have used a well known fact that the finite sum of negligible functions is negligible. This can be shown using the definition of negligible function from section 2.2. If we show that the sum of two negligible functions is negligible, then the result follows by induction. Suppose ϵ_1 and ϵ_2 are two negligible functions, then let $\epsilon_{12} = \epsilon_1 + \epsilon_2$. Since ϵ_1 and ϵ_2 are negligible functions, for an arbitrary $c > 0$, we have $c + 1 > 0$ and there exists $n_{\epsilon_1}, n_{\epsilon_2}$ such that for all $n \geq n_{\epsilon_1}$, $\epsilon_1(n) \leq n^{-c+1}$ and for all $n \geq n_{\epsilon_2}$, $\epsilon_2(n) \leq n^{-c+1}$. Now if we take $n_{\epsilon_{12}} = max\{n_{\epsilon_1}, n_{\epsilon_2}\}$, we have for all $n \geq n_{\epsilon_{12}}$, both $\epsilon_1(n) \leq n^{-c+1}$ and $\epsilon_2(n) \leq n^{-c+1}$. So, $\epsilon_{12}(n) = \epsilon_1(n) + \epsilon_2(n) \leq 2.n^{-c+1} \leq n.n^{-c+1} = n^{-c}$.

We recall the definition of computational pseudorandomness here: A probability distribution X on R_q is said to be computationally pseudorandom if there exists no efficient distinguisher D that distinguishes X from U, the uniform distribution on R_q with non-negligible advantage. i.e,

$$|Pr[t \leftarrow X, D(t) = 1] - Pr[t \leftarrow U, D(t) = 1]| < \epsilon(n)$$

for a negligible function $\epsilon(n)$, for all $n \in \mathbb{N}$.

B Proof of Lemma 4

Proof. Suppose D is a distinguisher for the distribution of $\bar{x}$ from uniform on R_q, we can then use oracle access to D to build a distinguisher D' for RLWE, hence proving that $\bar{x}$ is pseudorandom. D' is constructed as follows: On receiving an RLWE sample p from a RLWE challenger, D' samples x from the distribution ϕ and computes $\bar{p} = p + x$. D' then invokes the distinguisher D with input $\bar{p}$. D' is then set to output the output of the distinguisher D. Note that from lemma 3, if p is uniform, then $\bar{p}$ is uniform. Thus, if D can distinguish $\bar{p}$ from uniform with non-negligible advantage $\mathcal{A}dv_D$, then D' can distinguish RLWE sample p from uniform p with the same advantage. □