Weakening Assumptions for Publicly-Verifiable Deletion

James Bartusek[1](✉), Dakshita Khurana[2], Giulio Malavolta[3], Alexander Poremba[4], and Michael Walter[5]

[1] UC Berkeley, Berkeley, USA
bartusek.james@gmail.com
[2] UIUC, Champaign, USA
dakshita@illinois.edu
[3] Bocconi University and Max Planck Institute for Security and Privacy, Bochum, Germany
giulio.malavolta@hotmail.it
[4] Caltech, Pasadena, USA
aporemba@caltech.edu
[5] Ruhr-Universität Bochum, Bochum, Germany
michael.walter@rub.de

Abstract. We develop a simple compiler that generically adds publicly-verifiable deletion to a variety of cryptosystems. Our compiler only makes use of one-way functions (or one-way state generators, if we allow the public verification key to be quantum). Previously, similar compilers either relied on indistinguishability obfuscation along with any one-way function (Bartusek et al., ePrint:2023/265), or on almost-regular one-way functions (Bartusek, Khurana and Poremba, CRYPTO 2023).

1 Introduction

Is it possible to *provably* delete information by leveraging the laws of quantum mechanics? An exciting series of recent works [1–6, 9–11, 18, 19] have built a variety of quantum cryptosystems that support certifiable deletion of plaintext data and/or certifiable revocation of ciphertexts or keys.

The notion of certified deletion was formally introduced by Broadbent and Islam [6] for the one-time pad, where once the certificate is successfully verified, the plaintext remains hidden even if the secret (one-time pad) key is later revealed. This work has inspired a large body of research, aimed at understanding what kind of cryptographic primitives can be certifiably deleted. Recently, [4] built a compiler that generically adds the certified deletion property described above to any computationally secure commitment, encryption, attribute-based encryption, fully-homomorphic encryption, witness encryption or timed-release encryption scheme, *without making any additional assumptions*. Furthermore, it provides a strong *information-theoretic* deletion guarantee: Once an adversary generates a valid (classical) certificate of deletion, they cannot recover the

G. Rothblum and H. Wee (Eds.): TCC 2023, LNCS 14372, pp. 183–197, 2023.
https://doi.org/10.1007/978-3-031-48624-1_7

plaintext that was previously computationally determined by their view even given *unbounded time.* However, the compiled schemes satisfy privately verifiable deletion – namely, the encryptor generates a ciphertext together with secret parameters which are necessary for verification and must be kept hidden from the adversary.

Publicly Verifiable Deletion. The above limitation was recently overcome in [3], which obtained *publicly-verifiable* deletion (PVD) for all of the above primitives as well as new ones, such as CCA encryption, obfuscation, maliciously-secure blind delegation and functional encryption[1]. However, the compilation process proposed in [3] required the strong notion of indistinguishability obfuscation, regardless of what primitive one starts from. This was later improved in [5], which built commitments with PVD from injective (or almost-regular) one-way functions, and X encryption with PVD for $X \in \{\text{attribute-based, fully-homomorphic, witness, timed-release}\}$, assuming X encryption and trapdoored variants of injective (or almost-regular) one-way functions.

Weakening Assumptions for PVD. Given this state of affairs, it is natural to ask whether one can further relax the assumptions underlying publicly verifiable deletion, essentially matching what is known in the private verification setting. In this work, we show that the injectivity/regularity constraints on the one-way functions from prior work [5] are not necessary to achieve publicly-verifiable deletion; *any* one-way function suffices, or even a quantum weakening called a one-way state generator (OWSG) [17] if we allow the verification key to be quantum. Kretschmer [14] showed that, relative to an oracle, pseudorandom state generators (PRSGs) [12,17] exist even if $\mathsf{BQP} = \mathsf{QMA}$ (and thus $\mathsf{NP} \subseteq \mathsf{BQP}$). Because PRSGs are known to imply OWSGs [17], this allows us to base our generic compiler for PVD on something potentially even weaker than the existence of one-way functions.w

In summary, we improve [5] to obtain X with PVD for $X \in \{\text{statistically-binding commitment, public-key encryption, attribute-based encryption, fully-homomorphic encryption, witness encryption, timed-release encryption}\}$, assuming only X and any one-way function. We also obtain X with PVD for all the X above, assuming only X and any one-way state generator [17], but with a *quantum* verification key. Our primary contribution is conceptual: Our construction is inspired by a recent work on quantum-key distribution [16], which we combine with a proof strategy that closely mimics [3,5] (which in turn build on the proof technique of [4]).

1.1 Technical Outline

Prior Approach. We begin be recalling that prior work [5] observed that, given an appropriate *two-to-one* one-way function f, a commitment (with certified deletion) to a bit b can be

[1] A concurrent updated version of [10] also obtained functional encryption with certified deletion, although in the private-verification settings.

$$\mathsf{ComCD}(b) \propto \big(y, |x_0\rangle + (-1)^b |x_1\rangle\big)$$

where $(0, x_0), (1, x_1)$ are the two pre-images of (a randomly sampled) image y. Given an image y and a quantum state $|\psi\rangle$, they showed that any pre-image of y constitutes a valid certificate of deletion of the bit b. This certificate can be obtained by measuring the state $|\psi\rangle$ in the computational basis.

Furthermore, it was shown in [5] that in fact two-to-one functions are not needed to instantiate this template, it is possible to use more general types of one-way functions to obtain a commitment of the form

$$\mathsf{ComCD}(b) \propto \left(y, \sum_{x:f(x)=y, M(x)=0} |x\rangle + (-1)^b \sum_{x:f(x)=y, M(x)=1} |x\rangle\right).$$

where M denotes some binary predicate applied to the preimages of y. The work of [5] developed techniques to show that this satisfies certified deletion, as well as binding as long as the sets

$$\sum_{x:f(x)=y, M(x)=0} |x\rangle \quad \text{and} \quad \sum_{x:f(x)=y, M(x)=1} |x\rangle$$

are somewhat "balanced", i.e. for a random image y and the sets $S_0 = \{x : f(x) = y, M(x) = 0\}$ and $S_1 = \{x : f(x) = y, M(x) = 1\}$, it holds that $\frac{|S_0|}{|S_1|}$ is a fixed constant. Such "balanced" functions can be obtained from injective (or almost-regular) one-way functions by a previous result of [8].

Using Any One-Way Function. Our first observation is that it is not necessary to require x_0, x_1 to be preimages of the same image y. Instead, we can modify the above template to use randomly sampled $x_0 \neq x_1$ and compute $y_0 = F(x_0), y_1 = F(x_1)$ to obtain

$$\mathsf{ComCD}(b) \propto \big((y_0, y_1), |x_0\rangle + (-1)^b |x_1\rangle\big)$$

Unfortunately, as described so far, the phase b may not be statistically fixed by the commitment when F is a general one-way function, since if F is not injective, the y_0, y_1 do not determine the choice of x_0, x_1 that were used to encrypt the phase. To restore binding, we can simply append a commitment to $(x_0 \oplus x_1)$ to the state above, resulting in

$$\mathsf{ComCD}(b) \propto \big((y_0, y_1), \mathsf{Com}(x_0 \oplus x_1), |x_0\rangle + (-1)^b |x_1\rangle\big)$$

Assuming that Com is statistically binding, the bit b is (statistically) determined by the commitment state above, and in fact, can even be efficiently determined given $x_0 \oplus x_1$. This is because a measurement of $|x_0\rangle + (-1)^b |x_1\rangle$ in the Hadamard basis yields a string z such that $b = (x_0 \oplus x_1) \cdot z$.

Relation to [3]. In fact, one can now view this scheme as a particular instantiation of the subspace coset state based compiler from [3]. To commit to a bit b using the compiler of [3], we would sample (i) a random subspace S of $\mathbb{F}_2^n$, (ii) a random coset of S represented by a vector v, and (iii) a random coset of $S^\perp$ represented by a vector w. Then, the commitment would be

$$\mathsf{ComCD}(b) = \mathsf{Com}(S), \left|S_{v,w}\right\rangle, b \oplus \bigoplus_i v_i,$$

where $\left|S_{v,w}\right\rangle \propto \sum_{s \in S} (-1)^{s \cdot w} \left|s+v\right\rangle$ is the subspace coset state defined by S, v, w. A valid deletion certificate would be any vector in $S^\perp + w$, obtained by measuring $\left|S_{v,w}\right\rangle$ in the Hadamard basis.

However, in order to obtain publicly-verifiable deletion, [3] publish an obfuscated membership check program for $S^\perp + w$, which is general requires post-quantum indistinguishability obfuscation. Our main observation here is that we can sample S as an $(n-1)$-dimensional subspace, which means that $S^\perp + w$ will only consist of two vectors. Then, to obfuscate a membership check program for $S^\perp + w$, it suffices to publish a one-way function evaluated at each of the two vectors in $S^\perp + w$, which in our notation are x_0 and x_1.

To complete the derivation of our commitment scheme, note that to describe S, it suffices to specify the hyperplane that defines S, which in our notation is $x_0 \oplus x_1$. Finally, we can directly encode the bit b into the subspace coset state rather than masking it with the description of a random coset (in our case, there are only two cosets of S), and if we look at the resulting state in the Hadamard basis, we obtain $\propto \left|x_0\right\rangle + (-1)^b \left|x_1\right\rangle$.

Proving Security. Naturally, certified deletion security follows by adapting the proof technique from [3], as we discuss now. Recall that we will consider an experiment where the adversary is given an encryption of b and outputs a deletion certificate. If the certificate is valid, the output of the experiment is defined to be the adversary's left-over state (which we will show to be independent of b), otherwise the output of the experiment is set to $\perp$.

We will consider a sequence of hybrid experiments to help us prove that the adversary's view is statistically independent of b when their certificate verifies. The first step is to defer the dependence of the experiment on the bit b. In more detail, we will instead imagine sampling the distribution by guessing a uniformly random $c \leftarrow \{0,1\}$, and initializing the adversary with the following: $((y_0, y_1), \mathsf{Com}(x_0 \oplus x_1), \left|x_0\right\rangle + (-1)^c \left|x_1\right\rangle)$. The challenger later obtains input b and aborts the experiment (outputs $\perp$) if $c \neq b$. Since c was a uniformly random guess, the trace distance between the $b = 0$ and $b = 1$ outputs of this modified experiment is at least half the trace distance between the outputs of the original experiment. Moreover, we can actually consider a *purification* of this experiment where a register C is initialized in a superposition $\left|0\right\rangle + \left|1\right\rangle$ of two choices for c, and is later measured to determine the bit c.

Now, we observe that the joint quantum state of the challenger and adversary can be written as

$$\frac{1}{2} \sum_{c \in \{0,1\}} \left|c\right\rangle_{\mathsf{C}} \otimes \left(\left|x_0\right\rangle + (-1)^c \left|x_1\right\rangle\right)_{\mathsf{A}} = \frac{1}{\sqrt{2}} \left(\left|+\right\rangle_{\mathsf{C}} \left|x_0\right\rangle_{\mathsf{A}} + \left|-\right\rangle_{\mathsf{C}} \left|x_1\right\rangle_{\mathsf{A}}\right),$$

where the adversary is initialized with the register A. Intuitively, if the adversary returns a successful deletion certificate x such that $F(x) = y_{c'}$ for bit c', then they must have done this by measuring in the standard basis and collapsing the joint state to $Z^{c'} |+\rangle_{\mathsf{C}} |x_{c'}\rangle_{\mathsf{A}}$. We can formalize this intuition by introducing an extra abort condition into the experiment. That is, if the adversary returns some x such that $F(x) = y_{c'}$, the challenger will then measure their register in the Hadamard basis and abort if the result $c'' \neq c'$. By the one-wayness of F, we will be able to show that no adversary can cause the challenger to abort with greater than $\mathrm{negl}(\lambda)$ probability as a result of this measurement. This essentially completes the proof of our claim, because at this point the bit c is always obtained by measuring a Hadamard basis state in the standard basis, resulting in a uniformly random bit outcome that completely masks the dependence of the experiment on b.

Applications. Finally, we note that encryption with PVD can be obtained similarly by committing to each bit of the plaintext as

$$\mathsf{EncCD}(b) \propto \big((y_0, y_1), \mathsf{Enc}(x_0 \oplus x_1), |x_0\rangle + (-1)^b |x_1\rangle\big)$$

We also note that, following prior work [4], a variety of encryption schemes (e.g., ABE, FHE, witness encryption) can be plugged into the template above, replacing Enc with the encryption algorithm of ABE/FHE/witness encryption, yielding the respective schemes with publicly-verifiable deletion.

1.2 Concurrent and Independent Work

A concurrent work of Kitagawa, Nishimaki, and Yamakawa [13] obtains similar results on publicly-verifiable deletion from one-way functions. Similarly to our work, they propose a generic compiler to obtain X with publicly-verifiable deletion only assuming X plus one-way functions, for a variety of primitives, such as commitments, quantum fully-homomorphic encryption, attribute-based encryption, or witness encryption. One subtle difference, is that they need to assume the existence of *quantum* fully-homomorphic encryption (QFHE), even for building *classical* FHE with PVD, due to the evaluation algorithm computing over a quantum state. On the other hand, we obtain FHE with PVD using only plain FHE. At a technical level, their approach is based on one-time signatures for BB84 states, whereas our approach can (in retrospect) be thought of as using one-time signatures on the $|+\rangle$ state.

Differently from our work, [13] shows that their compiler can be instantiated from *hard quantum planted problems for NP*, whose existence is *implied* by most cryptographic primitives with PVD. In this sense, their assumptions can be considered minimal. Although we do not explore this direction in our work, we believe that a similar implication holds for our compiler as well. On the other hand, we propose an additional compiler, whose security relies solely on one-way state generators (OWSG), which is an assumption conjectured to be even *weaker* than one-way function.

2 Preliminaries

Let λ denote the security parameter. We write $\mathrm{negl}(\cdot)$ to denote any *negligible* function, which is a function f such that for every constant $c \in \mathbb{N}$ there exists $N \in \mathbb{N}$ such that for all $n > N$, $f(n) < n^{-c}$.

A finite-dimensional complex Hilbert space is denoted by $\mathcal{H}$, and we use subscripts to distinguish between different systems (or registers); for example, we let $\mathcal{H}_{\mathsf{A}}$ be the Hilbert space corresponding to a system A. The tensor product of two Hilbert spaces $\mathcal{H}_{\mathsf{A}}$ and $\mathcal{H}_{\mathsf{B}}$ is another Hilbert space denoted by $\mathcal{H}_{\mathsf{AB}} = \mathcal{H}_{\mathsf{A}} \otimes \mathcal{H}_{\mathsf{B}}$. We let $\mathcal{L}(\mathcal{H})$ denote the set of linear operators over $\mathcal{H}$. A quantum system over the 2-dimensional Hilbert space $\mathcal{H} = \mathbb{C}^2$ is called a *qubit*. For $n \in \mathbb{N}$, we refer to quantum registers over the Hilbert space $\mathcal{H} = \left(\mathbb{C}^2\right)^{\otimes n}$ as n-qubit states. We use the word *quantum state* to refer to both pure states (unit vectors $|\psi\rangle \in \mathcal{H}$) and density matrices $\rho \in \mathcal{D}(\mathcal{H})$, where we use the notation $\mathcal{D}(\mathcal{H})$ to refer to the space of positive semidefinite linear operators of unit trace acting on $\mathcal{H}$. The *trace distance* of two density matrices $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ is given by

$$\mathsf{TD}(\rho, \sigma) = \frac{1}{2}\mathsf{Tr}\left[\sqrt{(\rho - \sigma)^\dagger(\rho - \sigma)}\right].$$

A quantum channel $\Phi : \mathcal{L}(\mathcal{H}_{\mathsf{A}}) \rightarrow \mathcal{L}(\mathcal{H}_{\mathsf{B}})$ is a linear map between linear operators over the Hilbert spaces $\mathcal{H}_{\mathsf{A}}$ and $\mathcal{H}_{\mathsf{B}}$. We say that a channel Φ is *completely positive* if, for a reference system R of arbitrary size, the induced map $I_{\mathsf{R}} \otimes \Phi$ is positive, and we call it *trace-preserving* if $\mathsf{Tr}[\Phi(X)] = \mathsf{Tr}[X]$, for all $X \in \mathcal{L}(\mathcal{H})$. A quantum channel that is both completely positive and trace-preserving is called a quantum CPTP channel.

A *unitary* $U : \mathcal{L}(\mathcal{H}_{\mathsf{A}}) \rightarrow \mathcal{L}(\mathcal{H}_{\mathsf{A}})$ is a special case of a quantum channel that satisfies $U^\dagger U = UU^\dagger = I_{\mathsf{A}}$. A *projector* Π is a Hermitian operator such that $\Pi^2 = \Pi$, and a *projective measurement* is a collection of projectors $\{\Pi_i\}_i$ such that $\sum_i \Pi_i = I$.

A quantum polynomial-time (QPT) machine is a polynomial-time family of quantum circuits given by $\{\mathcal{A}_\lambda\}_{\lambda \in \mathbb{N}}$, where each circuit $\mathcal{A}_\lambda$ is described by a sequence of unitary gates and measurements; moreover, for each $\lambda \in \mathbb{N}$, there exists a deterministic polynomial-time Turing machine that, on input 1^λ, outputs a circuit description of $\mathcal{A}_\lambda$.

Imported Theorem 1 (Gentle Measurement [20]). *Let* ρ^{X} *be a quantum state and let* $(\Pi, \mathbb{I} - \Pi)$ *be a projective measurement on* X *such that* $\mathsf{Tr}(\Pi\rho) \geq 1 - \delta$. *Let*

$$\rho' = \frac{\Pi\rho\Pi}{\mathsf{Tr}(\Pi\rho)}$$

be the state after applying $(\Pi, \mathbb{I} - \Pi)$ *to* ρ *and post-selecting on obtaining the first outcome. Then,* $\mathsf{TD}(\rho, \rho') \leq 2\sqrt{\delta}$.

Imported Theorem 2 (Distinguishing implies Mapping [7]). *Let* D *be a projector,* Π_0, Π_1 *be orthogonal projectors, and* $|\psi\rangle \in \mathsf{Im}\left(\Pi_0 + \Pi_1\right)$. *Then,*

$$\|\Pi_1 \mathsf{D} \Pi_0 |\psi\rangle\|^2 + \|\Pi_0 \mathsf{D} \Pi_1 |\psi\rangle\|^2 \geq \frac{1}{2}\left(\|\mathsf{D}|\psi\rangle\|^2 - \left(\|\mathsf{D}\Pi_0|\psi\rangle\|^2 + \|\mathsf{D}\Pi_1|\psi\rangle\|^2\right)\right)^2.$$

3 Main Theorem

Theorem 3. *Let $F : \{0,1\}^{n(\lambda)} \rightarrow \{0,1\}^{m(\lambda)}$ be a one-way function secure against QPT adversaries. Let $\{\mathcal{Z}_\lambda(\cdot,\cdot,\cdot,\cdot)\}_{\lambda\in\mathbb{N}}$ be a quantum operation with four arguments: an $n(\lambda)$-bit string z, two $m(\lambda)$-bit strings y_0, y_1, and an $n(\lambda)$-qubit quantum state $|\psi\rangle$. Suppose that for any QPT adversary $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, $z \in \{0,1\}^{n(\lambda)}, y_0, y_1 \in \{0,1\}^{m(\lambda)}$, and $n(\lambda)$-qubit state $|\psi\rangle$,*

$$\left| \Pr\left[\mathcal{A}_\lambda(\mathcal{Z}_\lambda(z, y_0, y_1, |\psi\rangle)) = 1\right] - \Pr\left[\mathcal{A}_\lambda(\mathcal{Z}_\lambda(0^\lambda, y_0, y_1, |\psi\rangle)) = 1\right] \right| = \text{negl}(\lambda).$$

That is, $\mathcal{Z}_\lambda$ is semantically-secure with respect to its first input.[2] *Now, for any QPT adversary $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, consider the following distribution $\left\{\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b)\right\}_{\lambda\in\mathbb{N}, b\in\{0,1\}}$ over quantum states, obtained by running $\mathcal{A}_\lambda$ as follows.*

- *Sample $x_0, x_1 \leftarrow \{0,1\}^{n(\lambda)}$ conditioned on $x_0 \neq x_1$, define $y_0 = F(x_0), y_1 = F(x_1)$ and initialize $\mathcal{A}_\lambda$ with*

$$\mathcal{Z}_\lambda\left(x_0 \oplus x_1, y_0, y_1, \frac{1}{\sqrt{2}}\left(|x_0\rangle + (-1)^b |x_1\rangle\right)\right).$$

- *$\mathcal{A}_\lambda$'s output is parsed as a string $x' \in \{0,1\}^{n(\lambda)}$ and a residual state on register A'.*
- *If $F(x') \in \{y_0, y_1\}$, then output A', and otherwise output $\perp$.*

Then,

$$\mathsf{TD}\left(\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(0), \widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(1)\right) = \text{negl}(\lambda).$$

Proof. We define a sequence of hybrids.

- $\mathsf{Hyb}_0(b)$: This is the distribution $\left\{\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b)\right\}_{\lambda\in\mathbb{N}, b\in\{0,1\}}$ described above.
- $\mathsf{Hyb}_1(b)$: This distribution is sampled as follows.
  - Sample $x_0, x_1, y_0 = F(x_0), y_1 = F(x_1)$, prepare the state

$$\frac{1}{2}\sum_{c\in\{0,1\}} |c\rangle_{\mathsf{C}} \otimes \left(|x_0\rangle + (-1)^c |x_1\rangle\right)_{\mathsf{A}},$$

  and initialize $\mathcal{A}_\lambda$ with

$$\mathcal{Z}_\lambda\left(x_0 \oplus x_1, y_0, y_1, \mathsf{A}\right).$$

[2] One can usually think of $\mathcal{Z}_\lambda$ as just encrypting its first input and leaving the remaining in the clear. However, we need to formulate the more general definition of $\mathcal{Z}_\lambda$ that operates on all inputs to handle certain applications, such as attribute-based encryption. See [4] for details.

- $\mathcal{A}_\lambda$'s output is parsed as a string $x' \in \{0,1\}^{n(\lambda)}$ and a residual state on register A'.
- If $F(x') \notin \{y_0, y_1\}$, then output $\bot$. Next, measure register C in the computational basis and output $\bot$ if the result is $1-b$. Otherwise, output A'.

– $\mathsf{Hyb}_2(b)$: This distribution is sampled as follows.

- Sample $x_0, x_1, y_0 = F(x_0), y_1 = F(x_1)$, prepare the state

$$\frac{1}{2}\sum_{c\in\{0,1\}} |c\rangle_{\mathsf{C}} \otimes (|x_0\rangle + (-1)^c |x_1\rangle)_{\mathsf{A}},$$

and initialize $\mathcal{A}_\lambda$ with

$$\mathcal{Z}_\lambda (x_0 \oplus x_1, y_0, y_1, \mathsf{A}).$$

- $\mathcal{A}_\lambda$'s output is parsed as a string $x' \in \{0,1\}^{n(\lambda)}$ and a residual state on register A'.
- If $F(x') \notin \{y_0, y_1\}$, then output $\bot$. Next, let $c' \in \{0,1\}$ be such that $F(x') = y_{c'}$, measure register C in the Hadamard basis, and output $\bot$ if the result is $1-c'$. Next, measure register C in the computational basis and output $\bot$ if the result is $1-b$. Otherwise, output A'.

We define $\mathsf{Advt}(\mathsf{Hyb}_i) := \mathsf{TD}\left(\mathsf{Hyb}_i(0), \mathsf{Hyb}_i(1)\right)$. To complete the proof, we show the following sequence of claims.

Claim. $\mathsf{Advt}(\mathsf{Hyb}_2) = 0$.

Proof. This follows by definition. Observe that Hyb_2 only depends on the bit b when it decides whether to abort after measuring register C in the computational basis. But at this point, it is guaranteed that register C is in a Hadamard basis state, so this will result in an abort with probability $1/2$ regardless of the value of b.

Claim. $\mathsf{Advt}(\mathsf{Hyb}_1) = \mathrm{negl}(\lambda)$.

Proof. Given Sect. 3, it suffices to show that for each $b \in \{0,1\}$, $\mathsf{TD}(\mathsf{Hyb}_1(b), \mathsf{Hyb}_2(b)) = \mathrm{negl}(\lambda)$. The only difference between these hybrids is the introduction of a measurement of C in the Hadamard basis. By Gentle Measurement (Theorem 1), it suffices to show that this measurement results in an abort with probability $\mathrm{negl}(\lambda)$.

So suppose otherwise. That is, the following experiment outputs 1 with probability $\mathsf{non\text{-}negl}(\lambda)$.

– Sample $x_0, x_1, y_0 = F(x_0), y_1 = F(x_1)$, prepare the state

$$\frac{1}{2}\sum_{c\in\{0,1\}} |c\rangle_{\mathsf{C}} \otimes (|x_0\rangle + (-1)^c |x_1\rangle)_{\mathsf{A}},$$

and initialize $\mathcal{A}_\lambda$ with

$$\mathcal{Z}_\lambda (x_0 \oplus x_1, y_0, y_1, \mathsf{A}).$$

- $\mathcal{A}_\lambda$'s output is parsed as a string $x' \in \{0,1\}^{n(\lambda)}$ and a residual state on register A'.
- If $F(x') \notin \{y_0, y_1\}$, then output $\bot$. Next, let $c' \in \{0,1\}$ be such that $F(x') = y_{c'}$, measure register C in the Hadamard basis, and output 1 if the result is $1 - c'$.

Next, observe that we can commute the measurement of C in the Hadamard basis to before the adversary is initialized, without affecting the outcome of the experiment:

- Sample $x_0, x_1, y_0 = F(x_0), y_1 = F(x_1)$, prepare the state

  $$\frac{1}{2}\sum_{c\in\{0,1\}} |c\rangle_{\mathsf{C}} \otimes \left(|x_0\rangle + (-1)^c |x_1\rangle\right)_{\mathsf{A}} = \frac{1}{\sqrt{2}}\left(|+\rangle_{\mathsf{C}} |x_0\rangle_{\mathsf{A}} + |-\rangle_{\mathsf{C}} |x_1\rangle_{\mathsf{A}}\right),$$

  measure C in the Hadamard basis to obtain $c'' \in \{0,1\}$ and initialize $\mathcal{A}_\lambda$ with the resulting information

  $$\mathcal{Z}_\lambda\left(x_0 \oplus x_1, y_0, y_1, |x_{c''}\rangle_{\mathsf{A}}\right).$$

- $\mathcal{A}_\lambda$'s output is parsed as a string $x' \in \{0,1\}^{n(\lambda)}$ and a residual state on register A'.
- If $F(x') \notin \{y_0, y_1\}$, then output $\bot$. Next, let $c' \in \{0,1\}$ be such that $F(x') = y_{c'}$, and output 1 if $c'' = 1 - c'$.

Finally, note that any such $\mathcal{A}_\lambda$ can be used to break the one-wayness of F. To see this, we can first appeal to the semantic security of $\mathcal{Z}_\lambda$ and replace $x_0 \oplus x_1$ with $0^{n(\lambda)}$. Then, note that the only information $\mathcal{A}_\lambda$ receives is two images and one preimage F, and $\mathcal{A}_\lambda$ is tasked with finding the *other* preimage of F. Succeeding at this task with probability $\mathsf{non\text{-}negl}(\lambda)$ clearly violates the one-wayness of F.

Claim. $\mathsf{Advt}(\mathsf{Hyb}_0) = \mathrm{negl}(\lambda)$.

Proof. This follows because $\mathsf{Hyb}_1(b)$ is identically distributed to the distribution that outputs $\bot$ with probability $1/2$ and otherwise outputs $\mathsf{Hyb}_0(b)$, so the advantage of Hyb_0 is at most double the advantage of Hyb_1.

4 Cryptography with Publicly-Verifiable Deletion

Let us now introduce some formal definitions. A public-key encryption (PKE) scheme with publicly-verifiable deletion (PVD) has the following syntax.

- $\mathsf{PVGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk})$: the key generation algorithm takes as input the security parameter λ and outputs a public key pk and secret key sk.
- $\mathsf{PVEnc}(\mathsf{pk}, b) \to (\mathsf{vk}, |\mathsf{ct}\rangle)$: the encryption algorithm takes as input the public key pk and a plaintext b, and outputs a (public) verification key vk and a ciphertext $|\mathsf{ct}\rangle$.

- $\mathsf{PVDec}(\mathsf{sk}, |\mathsf{ct}\rangle) \to b$: the decryption algorithm takes as input the secret key sk and a ciphertext $|\mathsf{ct}\rangle$ and outputs a plaintext b.
- $\mathsf{PVDel}(|\mathsf{ct}\rangle) \to \pi$: the deletion algorithm takes as input a ciphertext $|\mathsf{ct}\rangle$ and outputs a deletion certificate π.
- $\mathsf{PVVrfy}(\mathsf{vk}, \pi) \to \{\top, \bot\}$: the verify algorithm takes as input a (public) verification key vk and a proof π, and outputs $\top$ or $\bot$.

Definition 1 (Correctness of deletion). *A PKE scheme with PVD satisfies* correctness of deletion *if for any b, it holds with* $1 - \text{negl}(\lambda)$ *probability over* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{PVGen}(1^\lambda), (\mathsf{vk}, |\mathsf{ct}\rangle) \leftarrow \mathsf{PVEnc}(\mathsf{pk}, b), \pi \leftarrow \mathsf{PVDel}(|\mathsf{ct}\rangle), \mu \leftarrow \mathsf{PVVrfy}(\mathsf{vk}, \pi)$ *that* $\mu = \top$.

Definition 2 (Certified deletion security). *A PKE scheme with PVD satisfies* certified deletion security *if it satisfies standard semantic security, and moreover, for any QPT adversary $\{\mathcal{A}_\lambda\}_{\lambda \in \mathbb{N}}$, it holds that*

$$\mathsf{TD}\left(\mathsf{EvPKE}_{\mathcal{A},\lambda}(0), \mathsf{EvPKE}_{\mathcal{A},\lambda}(1)\right) = \text{negl}(\lambda),$$

where the experiment $\mathsf{EvPKE}_{\mathcal{A},\lambda}(b)$ *is defined as follows.*

- *Sample* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{PVGen}(1^\lambda)$ *and* $(\mathsf{vk}, |\mathsf{ct}\rangle) \leftarrow \mathsf{PVEnc}(\mathsf{pk}, b)$.
- *Run* $\mathcal{A}_\lambda(\mathsf{pk}, \mathsf{vk}, |\mathsf{ct}\rangle)$, *and parse their output as a deletion certificate π and a state on register* A'.
- *If* $\mathsf{PVVrfy}(\mathsf{vk}, \pi) = \top$, *output* A', *and otherwise output* $\bot$.

Construction via OWF. We now present our generic compiler that augments any (post-quantum secure) PKE scheme with the PVD property, assuming the existence of one-way functions.

Construction 4. *[PKE with PVD from OWF] Let $\lambda \in \mathbb{N}$, let*

$$F : \{0,1\}^{n(\lambda)} \to \{0,1\}^{m(\lambda)}$$

be a one-way function, and let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be a standard (post-quantum) public-key encryption scheme. Consider the PKE scheme with PVD consisting of the following efficient algorithms:*

- $\mathsf{PVGen}(1^\lambda)$*: Same as* $\mathsf{Gen}(1^\lambda)$.
- $\mathsf{PVEnc}(\mathsf{pk}, b)$*: Sample* $x_0, x_1 \leftarrow \{0,1\}^{n(\lambda)}$, *define* $y_0 = F(x_0), y_1 = F(x_1)$, *and output*

$$\mathsf{vk} := (y_0, y_1), \quad |\mathsf{ct}\rangle := \left(\mathsf{Enc}(\mathsf{pk}, x_0 \oplus x_1), \frac{1}{\sqrt{2}}\left(|x_0\rangle + (-1)^b |x_1\rangle\right)\right).$$

- $\mathsf{PVDec}(\mathsf{sk}, |\mathsf{ct}\rangle)$*: Parse* $|\mathsf{ct}\rangle$ *as a classical ciphertext* ct' *and a quantum state* $|\psi\rangle$. *Compute* $z \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ct}')$, *measure* $|\psi\rangle$ *in the Hadamard basis to obtain* $w \in \{0,1\}^{n(\lambda)}$, *and output the bit* $b = z \cdot w$.

- $\mathsf{PVDel}(|\mathsf{ct}\rangle)$: *Parse* $|\mathsf{ct}\rangle$ *as a classical ciphertext* ct' *and a quantum state* $|\psi\rangle$*. Measure* $|\psi\rangle$ *in the computational basis to obtain* $x' \in \{0,1\}^{n(\lambda)}$*, and output* $\pi := x'$*.*
- $\mathsf{PVVrfy}(\mathsf{vk}, \pi)$: *Parse* vk *as* (y_0, y_1) *and output* $\top$ *if and only if* $F(\pi) \in \{y_0, y_1\}$*.*

Theorem 5. *If one-way functions exist, then Theorem 4 instantiated with any (post-quantum) public-key encryption scheme satisfies correctness of deletion (according to Definition 1) as well as (everlasting) certified deletion security according to Definition 2.*

Proof. Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a standard (post-quantum) public-key encryption scheme. Then, correctness of deletion follows from the fact that measuring $\frac{1}{\sqrt{2}}(|x_0\rangle + |x_1\rangle)$ in the Hadamard basis produces a vector orthogonal to $x_0 \oplus x_1$, whereas measuring the state $\frac{1}{\sqrt{2}}(|x_0\rangle - |x_1\rangle)$ in the Hadamard basis produces a vector that is not orthogonal to $x_0 \oplus x_1$.

Next, we note that semantic security follows from a sequence of hybrids. First, we appeal to the semantic security of the public-key encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ to replace $\mathsf{Enc}(\mathsf{pk}, x_0 \oplus x_1)$ with $\mathsf{Enc}(\mathsf{pk}, 0^{n(\lambda)})$. Next, we introduce a measurement of $\frac{1}{\sqrt{2}}(|x_0\rangle + (-1)^b |x_1\rangle)$ in the standard basis before initializing the adversary. By a straightforward application of Theorem 2, a QPT adversary that can distinguish whether or not this measurement was applied can be used to break the one-wayness of F. Finally, note that the ciphertext now contains no information about b, completing the proof.

Finally, the remaining part of certified deletion security follows from Theorem 3, by setting $\mathcal{Z}_\lambda(x_0 \oplus x_1, y_0, y_1, |\psi\rangle) = \mathsf{Enc}(\mathsf{pk}, x_0 \oplus x_1), y_0, y_1, |\psi\rangle$ and invoking the semantic security of the public-key encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$.

Remark 1. Following [4], we can plug various primitives into the above compiler to obtain X with PVD for $X \in \{$commitment, attribute-based encryption, fully-homomormphic encryption, witness encryption, timed-release encryption$\}$.

5 Publicly-Verifiable Deletion from One-Way State Generators

In this section, we show how to relax the assumptions behind our generic compiler for PVD to something potentially even weaker than one-way functions, namely the existence of so-called one-way state generators (if we allow for quantum verification keys). Morimae and Yamakawa [17] introduced one-way state generator (OWSG) as a quantum analogue of a one-way function.

Definition 3 (One-Way State Generator). *Let* $n \in \mathbb{N}$ *be the security parameter. A one-way state generator* (OWSG) *is a tuple* $(\mathsf{KeyGen}, \mathsf{StateGen}, \mathsf{Ver})$ *consisting of QPT algorithms:*

$\mathsf{KeyGen}(1^n) \to k$: *given as input* 1^n*, it outputs a uniformly random key* $k \leftarrow \{0,1\}^n$*.*

$\mathsf{StateGen}(k) \rightarrow \phi_k$: *given as input a key* $k \in \{0,1\}^n$, *it outputs an* m*-qubit quantum state* ϕ_k.

$\mathsf{Ver}(k', \phi_k) \rightarrow \top/\bot$: *given as input a supposed key* k' *and state* ϕ_k, *it outputs* $\top$ *or* $\bot$.

We require that the following property holds:

Correctness: For any $n \in \mathbb{N}$, *the scheme* $(\mathsf{KeyGen}, \mathsf{StateGen}, \mathsf{Ver})$ *satisfies*

$$\Pr[\top \leftarrow \mathsf{Ver}(k, \phi_k) \ : \ k \leftarrow \mathsf{KeyGen}(1^n),\ \phi_k \leftarrow \mathsf{StateGen}(k)] \geq 1 - \text{negl}(n).$$

Security: For any computationally bounded quantum algorithm $\mathcal{A}$ and any $t = \text{poly}(\lambda)$:

$$\Pr[\top \leftarrow \mathsf{Ver}(k', \phi_k) \ : \ k \leftarrow \mathsf{KeyGen}(1^n),\ \phi_k \leftarrow \mathsf{StateGen}(k),\ k' \leftarrow \mathcal{A}(\phi_k^{\otimes t})] \leq \text{negl}(n).$$

Morimae and Yamakawa [17] showed that if pseudorandom quantum state generators with $m \geq c \cdot n$ for some constant $c > 1$ exist, then so do one-way state generators. Informally, a pseudorandom state generator [12] is a QPT algorithm that, on input $k \in \{0,1\}^n$, outputs an m-qubit state $|\phi_k\rangle$ such that $|\phi_k\rangle^{\otimes t}$ over uniformly random k is computationally indistinguishable from a Haar random states of the same number of copies, for any polynomial $t(n)$. Recent works [14,15] have shown oracle separations between pseudorandom state generators and one-way functions, indicating that these quantum primitives are potentially weaker than one-way functions.

Publicly Verifiable Deletion from OWSG. To prove that our generic compiler yields PVD even when instantiated with a OWSG, it suffices to extend Theorem 3 as follows.

Theorem 6. *Let* $(\mathsf{KeyGen}, \mathsf{StateGen}, \mathsf{Ver})$ *be a OSWG from* $n(\lambda)$ *bits to* $m(\lambda)$ *qubits. Let* $\{\mathcal{Z}_\lambda(\cdot,\cdot,\cdot,\cdot)\}_{\lambda \in \mathbb{N}}$ *be a quantum operation with four arguments: an* $n(\lambda)$*-bit string* z, *two* $m(\lambda)$*-qubit quantum states* ϕ_0, ϕ_1, *and an* $n(\lambda)$*-qubit quantum state* $|\psi\rangle$. *Suppose that for any QPT adversary* $\{\mathcal{A}_\lambda\}_{\lambda \in \mathbb{N}}$, $z \in \{0,1\}^{n(\lambda)}$, $m(\lambda)$*-qubit states* ϕ_0, ϕ_1, *and* $n(\lambda)$*-qubit state* $|\psi\rangle$,

$$\left| \Pr[\mathcal{A}_\lambda\left(\mathcal{Z}_\lambda\left(z, \phi_0, \phi_1, |\psi\rangle\right)\right) = 1] - \Pr[\mathcal{A}_\lambda\left(\mathcal{Z}_\lambda\left(0^{n(\lambda)}, \phi_0, \phi_1, |\psi\rangle\right)\right) = 1] \right| = \text{negl}(\lambda).$$

That is, $\mathcal{Z}_\lambda$ *is semantically-secure with respect to its first input. Now, for any QPT adversary* $\{\mathcal{A}_\lambda\}_{\lambda \in \mathbb{N}}$, *consider the following distribution* $\left\{\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b)\right\}_{\lambda \in \mathbb{N}, b \in \{0,1\}}$ *over quantum states, obtained by running* $\mathcal{A}_\lambda$ *as follows.*

- *Sample* $x_0, x_1 \leftarrow \{0,1\}^{n(\lambda)}$, *generate quantum states* ϕ_{x_0} *and* ϕ_{x_1} *by running the procedure* $\mathsf{StateGen}$ *on input* x_0 *and* x_1, *respectfully, and initialize* $\mathcal{A}_\lambda$ *with*

$$\mathcal{Z}_\lambda\left(x_0 \oplus x_1, \phi_{x_0}, \phi_{x_1}, \frac{1}{\sqrt{2}}\left(|x_0\rangle + (-1)^b |x_1\rangle\right)\right).$$

- *$\mathcal{A}_\lambda$'s output is parsed as a string $x' \in \{0,1\}^{n(\lambda)}$ and a residual state on register* A'.
- *If* $\mathsf{Ver}(x', \psi_{x_i})$ *outputs $\top$ for some $i \in \{0,1\}$, then output* A'*, and otherwise output $\bot$.*

Then,

$$\mathsf{TD}\left(\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(0), \widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(1)\right) = \mathrm{negl}(\lambda).$$

Proof. The proof is analogus to Theorem 3, except that we invoke the security of the OWSG, rather than the one-wayness of the underlying one-way function.

Construction from OWSG. We now consider the following PKE scheme with PVD. The construction is virtually identical to Theorem 4, except that we replace one-way functions with one-way state generators. This means that the verification key is now quantum.

Construction 7 (PKE with PVD from OWSG). *Let $\lambda \in \mathbb{N}$ and let* $(\mathsf{KeyGen}, \mathsf{StateGen}, \mathsf{Ver})$ *be a OSWG, and let* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be a standard (post-quantum) public-key encryption scheme. Consider the following PKE scheme with PVD:*

- $\mathsf{PVGen}(1^\lambda)$*: Same as* $\mathsf{Gen}(1^\lambda)$.
- $\mathsf{PVEnc}(\mathsf{pk}, b)$*: Sample $x_0, x_1 \leftarrow \{0,1\}^{n(\lambda)}$ and generate quantum states ϕ_{x_0} and ϕ_{x_1} by running the procedure* ***StateGen*** *on input x_0 and x_1, respectfully. Then, output*

$$\mathsf{vk} := (\phi_{x_0}, \phi_{x_1}), \qquad |\mathsf{ct}\rangle := \left(\mathsf{Enc}(\mathsf{pk}, x_0 \oplus x_1), \frac{1}{\sqrt{2}}\left(|x_0\rangle + (-1)^b |x_1\rangle\right)\right).$$

- $\mathsf{PVDec}(\mathsf{sk}, |\mathsf{ct}\rangle)$*: Parse $|\mathsf{ct}\rangle$ as a classical ciphertext ct' and a quantum state $|\psi\rangle$. Compute $z \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$, measure $|\psi\rangle$ in the Hadamard basis to obtain $w \in \{0,1\}^{n(\lambda)}$, and output the bit $b = z \cdot w$.*
- $\mathsf{PVDel}(|\mathsf{ct}\rangle)$*: Parse $|\mathsf{ct}\rangle$ as a classical ciphertext ct' and a quantum state $|\psi\rangle$. Measure $|\psi\rangle$ in the computational basis to obtain $x' \in \{0,1\}^{n(\lambda)}$, and output $\pi := x'$.*
- $\mathsf{PVVrfy}(\mathsf{vk}, \pi)$*: Parse vk as (ϕ_{x_0}, ϕ_{x_1}) and output $\top$ if and only if $\mathsf{Ver}(\pi, \phi_{x_i})$ outputs $\top$, for some $i \in \{0,1\}$. Otherwise, output $\bot$.*

Remark 2. Unlike in Theorem 4, the verification key vk in Theorem 7 is quantum. Hence, the procedure $\mathsf{PVVrfy}(\mathsf{vk}, \pi)$ in Theorem 7 may potentially consume the public verification key (ϕ_{x_0}, ϕ_{x_1}) when verifying a dishonest deletion certificate π. However, by the security of the OWSG scheme, we can simply hand out $(\phi_{x_0}^{\otimes t}, \phi_{x_1}^{\otimes t})$ for any number of $t = \mathrm{poly}(\lambda)$ many copies without compromising security. This would allow multiple users to verify whether a (potentially dishonest) deletion certificate is valid. We focus on the case $t = 1$ for simplicity.

Theorem 8. *If one-way state generators exist, then Theroem 7 instantiated with any (post-quantum) public-key encryption scheme satisfies correctness of deletion (according to Definition 1) as well as (everlasting) certified deletion security according to Definition 2.*

Proof. The proof is analogous to Theroem 5, except that we again invoke security of the OWSG, rather than the one-wayness of the underlying one-way function.

Following [4], we also immediately obtain:

Theorem 9. *If one-way state generators exist, then there exists a generic compiler that adds PVD to any (post-quantum) public-key encryption scheme. Moreover, plugging X into the compiler yields X with* PVD *for*

$$X \in \left\{ \begin{array}{c} \textit{commitment, attribute-based encryption, fully-homomormphic} \\ \textit{encryption, witness encryption, timed-release encryption} \end{array} \right\}.$$

Acknowledgements. DK was supported in part by NSF 2112890, NSF CNS-2247727, and DARPA SIEVE. This material is based upon work supported by the Defense Advanced Research Projects Agency through Award HR00112020024. GM was partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2092 CASA - 390781972.

References

1. Agrawal, S., Kitagawa, F., Nishimaki, R., Yamada, S., Yamakawa, T.: Public key encryption with secure key leasing. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, vol. 14004, pp. 581–610. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-30545-0_20
2. Ananth, P., Poremba, A., Vaikuntanathan, V.: Revocable cryptography from learning with errors. Cryptology ePrint Archive, Paper 2023/325 (2023). https://eprint.iacr.org/2023/325
3. Bartusek, J., et al.: Obfuscation and outsourced computation with certified deletion. Cryptology ePrint Archive, Paper 2023/265 (2023). https://eprint.iacr.org/2023/265
4. Bartusek, J., Khurana, D.: Cryptography with certified deletion. In: Crypto 2023 (2023, to appear)
5. Bartusek, J., Khurana, D., Poremba, A.: Publicly-verifiable deletion via target-collapsing functions. In: Crypto 2023 (2023, to appear)
6. Broadbent, A., Islam, R.: Quantum encryption with certified deletion. In: Pass, R., Pietrzak, K. (eds.) TCC 2020. LNCS, vol. 12552, pp. 92–122. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-64381-2_4
7. Dall'Agnol, M., Spooner, N.: On the necessity of collapsing. Cryptology ePrint Archive, Paper 2022/786 (2022). https://eprint.iacr.org/2022/786
8. Haitner, I., Horvitz, O., Katz, J., Koo, C.-Y., Morselli, R., Shaltiel, R.: Reducing complexity assumptions for statistically-hiding commitment. J. Cryptol. **22**(3), 283–310 (2007). https://doi.org/10.1007/s00145-007-9012-8

9. Hiroka, T., Morimae, T., Nishimaki, R., Yamakawa, T.: Quantum encryption with certified deletion, revisited: public key, attribute-based, and classical communication. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021. LNCS, vol. 13090, pp. 606–636. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-92062-3_21
10. Hiroka, T., Morimae, T., Nishimaki, R., Yamakawa, T.: Certified everlasting functional encryption. Cryptology ePrint Archive, Paper 2022/969 (2022). https://eprint.iacr.org/2022/969, https://eprint.iacr.org/2022/969
11. Hiroka, T., Morimae, T., Nishimaki, R., Yamakawa, T.: Certified everlasting zero-knowledge proof for QMA. In: Dodis, Y., Shrimpton, T. (eds.) CRYPTO 2022, Part I. LNCS, vol. 13507, pp. 239–268. Springer, Cham (2022)
12. Ji, Z., Liu, Y.-K., Song, F.: Pseudorandom quantum states. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018. LNCS, vol. 10993, pp. 126–152. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-96878-0_5
13. Kitagawa, F., Nishimaki, R., Yamakawa, T.: Publicly verifiable deletion from minimal assumptions. Cryptology ePrint Archive, Paper 2023/538 (2023). https://eprint.iacr.org/2023/538
14. Kretschmer, W.: Quantum pseudorandomness and classical complexity. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2021). https://doi.org/10.4230/LIPICS.TQC.2021.2, https://drops.dagstuhl.de/opus/volltexte/2021/13997/
15. Kretschmer, W., Qian, L., Sinha, M., Tal, A.: Quantum cryptography in algorithmica. In: Saha, B., Servedio, R.A. (eds.) Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20–23, 2023, pp. 1589–1602. ACM (2023). https://doi.org/10.1145/3564246.3585225
16. Malavolta, G., Walter, M.: Non-interactive quantum key distribution. Cryptology ePrint Archive, Paper 2023/500 (2023). https://eprint.iacr.org/2023/500
17. Morimae, T., Yamakawa, T.: Quantum commitments and signatures without one-way functions. In: Dodis, Y., Shrimpton, T. (eds.) CRYPTO 2022. LNCS, vol. 13507, pp. 269–295. Springer, Heidelberg (2022). https://doi.org/10.1007/978-3-031-15802-5_10
18. Poremba, A.: Quantum proofs of deletion for learning with errors. In: Kalai, Y.T. (ed.) 14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10–13, 2023, MIT, Cambridge, Massachusetts, USA. LIPIcs, vol. 251, pp. 90:1–90:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2023). https://doi.org/10.4230/LIPIcs.ITCS.2023.9
19. Unruh, D.: Revocable quantum timed-release encryption. J. ACM **62**(6) (2015). https://doi.org/10.1145/2817206
20. Winter, A.J.: Coding theorem and strong converse for quantum channels. IEEE Trans. Inf. Theory **45**(7), 2481–2485 (1999). https://doi.org/10.1109/18.796385