Cryptography with Certified Deletion

James Bartusek* and Dakshita Khurana**

Abstract. We propose a unifying framework that yields an array of cryptographic primitives with certified deletion. These primitives enable a party in possession of a quantum ciphertext to generate a classical certificate that the encrypted plaintext has been information-theoretically deleted, and cannot be recovered even given unbounded computational resources.

- For $X \in \{\mathsf{public\text{-}key}, \mathsf{attribute\text{-}based}, \mathsf{fully\text{-}homomorphic}, \mathsf{witness}, \mathsf{timed\text{-}release}\}$, our compiler converts any (post-quantum) X encryption to X encryption with certified deletion. In addition, we compile statistically-binding commitments to statistically-binding commitments with certified everlasting hiding. As a corollary, we also obtain statistically-sound zero-knowledge proofs for QMA with certified everlasting zero-knowledge assuming statistically-binding commitments.
- We also obtain a strong form of everlasting security for two-party and multi-party computation in the dishonest majority setting. While simultaneously achieving everlasting security against *all* parties in this setting is known to be impossible, we introduce *everlasting security transfer (EST)*. This enables *any one* party (or a subset of parties) to dynamically and certifiably information-theoretically delete other participants' data after protocol execution. We construct general-purpose secure computation with EST assuming statistically-binding commitments, which can be based on one-way functions or pseudorandom quantum states.

We obtain our results by developing a novel proof technique to argue that a bit b has been *information-theoretically deleted* from an adversary's view once they output a valid deletion certificate, despite having been previously *information-theoretically determined* by the ciphertext they held in their view. This technique may be of independent interest.

1 Introduction

Deletion in a classical world. On classical devices, data is stored and exchanged as a string of bits. There is nothing that can prevent an untrusted device with access to such a string from making arbitrarily many copies of it. Thus, it seems hopeless to try to *force* an untrusted device to delete classical data. Even if the string is merely a ciphertext encoding an underlying plaintext, there is no way to prevent a server from keeping that ciphertext around in memory forever. If at some point in the future, the security of the underlying encryption scheme is

* UC Berkeley. Email: bartusek.james@gmail.com
** UIUC. Email: dakshita@illinois.edu.

broken either via brute-force or major scientific advances, or if the key is compromised and makes its way to the server, the server will be able to recover the underlying plaintext. This may be unacceptable in situations where extremely sensitive data is being transmitted or computed upon.

In fact, there has recently been widespread interest in holding data collectors accountable in responding to "data deletion requests" from their clients, as evidenced by data deletion clauses in legal regulations adopted by the European Union [17] and California [1]. Unfortunately, the above discussion shows that these laws cannot be cryptographically enforced against malicious data collectors, though there has been recent work on cryptographically *formalizing* what it means for *honest* data collectors to follow such guidelines [19].

Deletion in a quantum world. The *uncertainty principle* [24], which lies at the foundation of quantum mechanics, completely disrupts the above classical intuition. It asserts the existence of pairs of measurable quantities such that precisely determining one quantity (e.g. the position of an electron) implies the *inability* to determine the other (e.g. the momentum of the electron). While such effects only become noticeable at an extreme microscopic scale, the pioneering work of Wiesner [43] suggested that the peculiar implications of the uncertainty principle could be leveraged to perform seemingly impossible "human-scale" information processing tasks.

Given the inherent "destructive" properties of information guaranteed by the uncertainty principle, provable data deletion appears to be a natural information processing task that, while impossible classically, may become viable quantumly. Surprisingly, the explicit study of data deletion in a quantum world has only begun recently. However, over the last few years, this question has been explored in many different contexts. Initial work studied deletion in the context of non-local games [18] and information-theoretic proofs of deletion with partial security [12], while the related notion of *revocation* was introduced in [41].

The work of [11] first considered certified deletion in the context of encryption schemes, leveraging the uncertainty principle to obtain one-time pad encryption with certified deletion. This caused a great deal of excitement, leading to many recent followup works on deletion in a cryptographic context: device-independent security of one-time pad encryption with certified deletion [32], public-key and attribute-based encryption with certified deletion [25], commitments and zero-knowledge with certified everlasting hiding [27], and most recently fully-homomorphic encryption with certified deletion [39].

This work. Our work makes new definitional, conceptual and technical contributions. Our key contribution is a new proof technique to show that many natural encryption schemes satisfy security with certified deletion. This improves prior work in many ways, as we summarize below.

1. **A unified framework.** We present a simple compiler that relies on conjugate coding/BB84 states [43,6] to bootstrap semantically-secure cryptosystems to semantically-secure cryptosystems with certified deletion. For any

$X \in$ {public-key encryption, attribute-based encryption, witness encryption, timed-release encryption, statistically-binding commitment}, we immediately obtain "X with certified deletion" by plugging X into our compiler. This compiler builds on [11], who used BB84 states in the context of certified deletion for one-time pad encryption.

2. **Stronger definitions.** We consider a strong definition of security with certified deletion for public-key primitives, which stipulates that if an adversary in possession of a quantum ciphertext encrypting bit b issues a certificate of deletion which passes verification, then the bit b must now be *information-theoretically* hidden from the adversary.
   Previous definitions of public-key and fully-homomorphic encryption with certified deletion [25,39] considered a weaker experiment, inspired by [11], where after deletion, the adversary is explicitly given the secret key, but is still required to be computationally bounded. For the public-key setting, we consider this prior definition to capture a (strong) *security against key leakage* property, as opposed to a *certified deletion* property[1]. In the full version [5], we show that the everlasting flavor of our definition implies prior definitions. Intuitively, this is because for public-key schemes, an adversary can sample a secret key on its own given sufficient computational resources. Moreover, in the case of fully-homomorphic encryption (FHE), prior work [39] considered definitions (significantly) weaker than semantic security.[2] We obtain the first semantically-secure FHE with certified deletion from standard LWE.
3. **Simpler constructions and weaker assumptions.** Our compiler removes the need to rely on complex cryptographic primitives such as non-committing encryption and indistinguishability obfuscation as in [25], or idealized models such as random oracles as in [41,27], or complex quantum states (such as Gaussian coset states) as in [39], instead yielding simple schemes satisfying certified deletion for a range of primitives from BB84 states and minimal assumptions.
   In fact, reliance on non-committing encryption was a key reason that prior techniques did not yield homomorphic encryption schemes with certified deletion, since compact homomorphic encryption schemes cannot simultaneously be non-committing [29]. Our work builds simple homomorphic encryption schemes that support certified deletion by eliminating the need to rely on non-committing properties, and instead only relying on semantic security of an underlying encryption scheme.
4. **Overcoming barriers to provable security.** How can one prove that a bit b has been *information-theoretically deleted* from an adversary's view once

[1] In contrast, in the one-time pad encryption setting as considered by [11], the original encrypted message is already information-theoretically hidden from the adversary, so to obtain any interesting notion of certified deletion, one must explicitly consider leaking the secret key.

[2] Subsequent to the original posting of our paper on arXiv, an update to [39] was posted with somewhat different results. We provide a comparison between our work and the updated version of [39] in Section 1.3.

they produce a valid deletion certificate, while it was previously information-theoretically *determined* by the ciphertext they hold in their view?
Prior work [41,25,27,39] resorted to either idealized models or weaker definitions, and constructions with layers of indirection, in order to get around this barrier. We develop a novel proof technique that resolves this issue by (1) carefully deferring the dependence of the experiment on the plaintext bit, and (2) identifying an efficiently checkable predicate on the adversary's state after producing a valid deletion certificate. We rely on semantic security of encryption to show that this predicate must hold, and we argue that if the predicate holds, the adversary's left-over state is statistically independent of the plaintext bit. This allows us to prove certified deletion security for simple and natural schemes.

5. **New implications to secure computation: Everlasting Security Transfer (EST).** We introduce the concept of *everlasting security transfer*. Everlasting security guarantees (malicious) security against a participant in a secure two-(or multi-)party computation protocol even if the participant becomes computationally unbounded after protocol execution. We introduce and build secure computation protocols where participants are able to *transfer* everlasting security properties from one party to another, even after the protocol ends.

We elaborate on our results in more detail below, then we provide an overview of our techniques.

1.1 Our results

Warmup: secret sharing with certified deletion. We begin by considering certified deletion in the context of one of the simplest cryptographic primitives: information-theoretic, two-out-of-two secret sharing. Here, a dealer Alice would like to share a classical secret bit b between two parties Bob and Charlie, such that

1. **(Secret sharing.)** The individual views of Bob and Charlie perfectly hide b, while the joint view of Bob and Charlie can be used to reconstruct b, and
2. **(Certified deletion.)** Bob may generate a deletion certificate for Alice, guaranteeing that b has been *information theoretically removed* from the *joint* view of Bob and Charlie.

That is, as long as Bob and Charlie do not collude at the time of generating the certificate of deletion, their joint view upon successful verification of this certificate is guaranteed to become independent of b. As long as the certificate verifies, b will be perfectly hidden from Bob and Charlie *even if* they decide to later collude.

To build such a secret sharing scheme, we start by revisiting the usage of conjugate coding/BB84 states to obtain encryption with certified deletion, which was first explored in [11]. While the construction in [11] relies on a seeded randomness extractor in combination with BB84 states, we suggest a simpler alternative that replaces the seeded extractor with the XOR function. Looking

ahead, this simplification combined with other proof techniques will help generically lift our secret sharing scheme to obtain several encryption schemes with certified deletion.

Consider a random string $x \leftarrow \{0,1\}^\lambda$, and a random set of bases $\theta \leftarrow \{0,1\}^\lambda$ (where 0 corresponds to the standard basis and 1 corresponds to the Hadamard basis). To obtain a scheme with certifiable deletion, we will build on the intuition that it is impossible to recover x given only BB84 states $|x\rangle_\theta$ without knowledge of the basis θ. Furthermore, measuring $|x\rangle_\theta$ in an incorrect basis θ' will destroy (partial) information about x.

Thus to secret-share a bit b in a way that supports deletion, the dealer will sample $x \leftarrow \{0,1\}^\lambda$ and bases $\theta \leftarrow \{0,1\}^\lambda$. Bob's share is then

$$|x\rangle_\theta$$

and Charlie's share is

$$\theta, b' = b \oplus \bigoplus_{i:\theta_i=0} x_i$$

That is, in Charlie's share, b is masked by the bits of x that are encoded in the standard basis.

We note that Bob's share contains only BB84 states while Charlie's share is entirely classical. Bob can now produce a certificate of deletion by returning the results of measuring all his BB84 states in the Hadamard basis, and Alice will accept as a valid certificate any string x' such that $x_i = x'_i$ for all i where $\theta_i = 1$. We show that this scheme is indeed a two-out-of-two secret sharing scheme that satisfies certified deletion as defined above.

A conceptually simple and generic compiler. As our key technical contribution, we upgrade the secret sharing with certified deletion scheme to the public-key setting by encrypting Charlie's share. In more detail, to encrypt a bit b with respect to any encryption scheme, we first produce two secret shares of b as described above, and then release a ciphertext that contains (1) Bob's share in the clear and (2) an encryption of Charlie's share. To certifiably delete a ciphertext, one needs to simply measure the quantum part of the ciphertext (i.e., Bob's share) in the Hadamard basis. Intuitively, since information about the bases (Charlie's share) is hidden at the time of producing the certificate of deletion, generating a certificate that verifies must mean information theoretically losing the description of computational basis states.

This method of converting a two-party primitive (i.e. secret sharing with certified deletion) into one-party primitives (i.e. encryption schemes with certified deletion) is reminiscent of other similar compilers in the literature, for instance those converting probabilistically checkable proofs to succinct arguments [7,28]. In our case, just like those settings, while the intuition is relatively simple, the proof turns out to be fairly non-trivial.

Our main theorem. In (almost) full generality, our main theorem says the following.[3] Consider an arbitrary family of distributions $\{\mathcal{Z}_\lambda(\theta)\}_{\lambda\in\mathbb{N},\theta\in\{0,1\}^\lambda}$ and an arbitrary class $\mathscr{A}$ of computationally bounded adversaries $\mathcal{A} = \{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, such that $\mathcal{Z}_\lambda(\theta)$ semantically hides θ against $\mathcal{A}_\lambda$. Then, consider the following distribution $\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b)$ over quantum states, parameterized by a bit $b \in \{0,1\}$.

- Sample $x, \theta \leftarrow \{0,1\}^\lambda$ and initialize $\mathcal{A}_\lambda$ with

$$\mathcal{Z}_\lambda(\theta), b \oplus \bigoplus_{i:\theta_i=0} x_i, |x\rangle_\theta\,.$$

- $\mathcal{A}_\lambda$'s output is parsed as a bitstring $x' \in \{0,1\}^\lambda$ and a residual state on register A'.
- If $x_i = x'_i$ for all i such that $\theta_i = 1$ then output A', and otherwise output a special symbol $\bot$.

Then,

Theorem 1. *For every $\mathcal{A} \in \mathscr{A}$, the trace distance between $\widetilde{Z}_\lambda^{\mathcal{A}_\lambda}(0)$ and $\widetilde{Z}_\lambda^{\mathcal{A}_\lambda}(1)$ is* $\mathrm{negl}(\lambda)$.

Intuitively, this means that as long as the adversary $\mathcal{A}_\lambda$ is computationally bounded *at the time of producing any deletion certificate* x' that properly verifies (meaning that x'_i is the correct bit encoded at index i for any indices encoded in the Hadamard basis), their left-over state *statistically* contains only negligible information about the original encrypted bit b. That is, once the certificate verifies, information about b cannot be recovered information-theoretically even given unbounded time from the adversary's residual state.

This theorem is both quite simple and extremely general. The quantum part that enables certified deletion only involves simple BB84 states, and we require no additional properties of the underlying distribution $\mathcal{Z}_\lambda$ except for the fact that $\mathcal{Z}_\lambda(\theta)$ and $\mathcal{Z}_\lambda(0^\lambda)$ are indistinguishable to some class of adversaries. [4] We now discuss our (immediate) applications in more detail.

[3] In order to fully capture all of our applications, we actually allow $\mathcal{Z}_\lambda$ to operate on all inputs, including the BB84 states. See Section 3 for the precise details.

[4] It may seem counter-intuitive that the certified deletion guarantees provided by our theorem hold even when instantiating $\mathcal{Z}_\lambda$ with general semantically secure schemes, such as a fully-homomorphic encryption scheme. In particular, what if an adversary evaluated the FHE to recover a classical encryption of b, and then reversed their computation and finally produced a valid deletion certificate? This may seem to contradict everlasting security, since a classical ciphertext could be used to recover b given unbounded time. However, this attack is actually not feasible. After performing FHE evaluation coherently, the adversary would obtain a register holding a superposition over classical ciphertexts encrypting b, but with different random coins. Measuring this superposition to obtain a single classical ciphertext would collapse the state, and prevent the adversary from reversing their computation to eventually produce a valid deletion certificate. Indeed, our Theorem rules out this (and all other) efficient attacks.

Public-key, attribute-based and witness encryption. Instantiating the distribution $\mathcal{Z}_\lambda$ with the encryption procedure for any public-key encryption scheme, we obtain a public-key encryption scheme with certified deletion.

We also observe that we can instantiate the distribution $\mathcal{Z}_\lambda$ with the encryption procedure for any *attribute-based* encryption scheme, and immediately obtain an attribute-based encryption scheme with certified deletion. Previously, this notion was only known under the assumption of indistinguishability obfuscation, and also only satisfied the weaker key leakage style definition discussed above [25]. Finally, instantiating $\mathcal{Z}_\lambda$ with any *witness encryption* scheme implies a witness encryption scheme with certified deletion.

Fully-homomorphic encryption. Next, we consider the question of computing on encrypted data. We observe that, if $\mathcal{Z}_\lambda$ is instantiated with the encryption procedure Enc for a *fully-homomorphic* encryption scheme [20,9,21], then given $|x\rangle_\theta, \mathsf{Enc}(\theta, b \oplus \bigoplus_{i:\theta_i=0} x_i)$, one could run a homomorphic evaluation procedure in superposition to recover (a superposition over) $\mathsf{Enc}(b)$. Additionally, given multiple ciphertexts, one can even compute arbitrary functionalities over the encrypted plaintexts. Moreover, if such evaluation is done *coherently* (without performing measurements), then it can be reversed and the deletion procedure can subsequently be run on the original ciphertexts.

This immediately implies what we call a "blind delegation with certified deletion" protocol, which allows a computationally weak client to utilize the resources of a computationally powerful server, while (i) keeping its data hidden from the server during the protocol, and (ii) ensuring that its data is *information-theoretically* deleted from the server afterwards, by requesting a certificate of deletion. We show that, as long as the server behaves honestly during the "function evaluation" phase of the protocol, then even if it is arbitrarily malicious after the function evaluation phase, it cannot both pass deletion verification and maintain any information about the client's original plaintexts.

Recently, Poremba [39] also constructed a fully-homomorphic encryption scheme satisfying a weaker notion of certified deletion.[5] In particular, the guarantee in [39] is that from the perspective of any server that passes deletion with *sufficiently high probability*, there is significant entropy in the client's original *ciphertext*. This does not necessarily imply anything about the underlying plaintext, since a ciphertext encrypting a fixed bit b may be (and usually will be) highly entropic. Moreover, their construction makes use of relatively complicated and highly entangled *Gaussian coset states* in order to obtain these deletion properties. In summary, our framework simultaneously strengthens the security (to standard semantic security of the plaintext) and simplifies the construction of fully-homomorphic encryption with certified deletion. We also remark that neither our work nor [39] considers security against servers that may be malicious during the function evaluation phase of the blind delegation with certified deletion protocol. We leave obtaining security against fully malicious servers as an interesting direction for future research.

[5] We discuss comparisons with a recently updated version of [39] in Section 1.3.

Commitments and zero-knowledge. Next, we consider *commitment schemes*. A fundamental result in quantum cryptography states that one cannot use quantum communication to build a commitment that is simultaneously statistically hiding and statistically binding [35,34]. Intriguingly, [27] demonstrated the feasibility of statistically-binding commitments with a *certified everlasting hiding* property, where hiding is computational during the protocol, but becomes information-theoretic after the receiver issues a valid deletion certificate. However, their construction relies on the idealized quantum random oracle model. Using our framework, we show that *any* (post-quantum) statistically-binding computationally-hiding commitment implies a statistically-binding commitment with certified everlasting hiding. Thus, we obtain statistically-binding commitments with certified everlasting hiding in the plain model from post-quantum one-way functions, and even from plausibly weaker assumptions like *pseudorandom quantum states* [3,37].

Following implications in [27] from commitments with certified deletion to zero-knowledge, we also obtain interactive proofs for NP (and more generally, QMA) with *certified everlasting zero-knowledge*. These are proofs that are statistically sound, and additionally the verifier may issue a classical certificate *after the protocol ends* showing that the verifier has information-theoretically deleted all secrets about the statement being proved. Once a computationally bounded verifier issues a valid certificate, the proof becomes *statistically* zero-knowledge (ZK). Similarly to the case of commitments, while proofs for QMA or NP are unlikely to simultaneously satisfy *statistical soundness* and *statistical ZK*, [27] previously introduced and built statistically sound, certified everlasting ZK proofs in the random oracle model. On the other hand, we obtain a construction in the plain model from any statistically-binding commitment.

Timed-release encryption. As another immediate application, we consider the notion of *revocable* timed-release encryption. Timed-release encryption schemes (also known as time-lock puzzles) have the property that, while ciphertexts can eventually be decrypted in some polynomial time, it takes *at least* some (parallel) $T(\lambda)$ time to do so. [41] considered adding a *revocable* property to such schemes, meaning that the recipient of a ciphertext can either eventually decrypt the ciphertext in $\geq T(\lambda)$ time, or issue a certificate of deletion proving that they will *never* be able to obtain the plaintext. [41] constructs semantically-secure revocable timed-release encryption assuming post-quantum timed-release encryption, but with the following drawbacks: the certificate of deletion is a *quantum state*, and the underlying scheme must either be *exponentially* hard or security must be proven in the idealized quantum random oracle model.

We can plug any post-quantum timed-release encryption scheme into our framework, and obtain revocable timed-released encryption from (polynomially-hard) post-quantum timed-released encryption, with a classical deletion certificate. Note that, when applying our main theorem, we simply instantiate the class of adversaries to be those that are $T(\lambda)$-parallel time bounded.

Secure computation with Everlasting Security Transfer (EST). Secure computation allows mutually distrusting participants to compute on joint private inputs while revealing no information beyond the output of the computation. The first templates for secure computation that make use of quantum information were proposed in a combination of works by Crépeau and Kilian [13], and Kilian [31]. For a while [36,46] it was believed that *unconditionally secure computation* could be realized based on a specific cryptographic building block: an *unconditionally secure quantum bit commitment*. Unfortunately, beliefs that unconditionally secure quantum bit commitments exist [10] were subsequently proven false [35,34], and the possibility of unconditional secure computation was also ruled out [33].

As such, secure computation protocols must either assume an honest majority or necessarily rely on computational hardness to achieve security against adversaries that are computationally bounded. But this may be troublesome when participants wish to compute on extremely sensitive data, such as medical or government records. In particular, consider a server that computes on highly sensitive data and keeps information from the computation around in memory forever. Such a server may be able to eventually recover data if the underlying hardness assumption breaks down in the future. In this setting, it is natural to ask: Can we use computational assumptions to design "everlasting" secure protocols against an adversary that is computationally bounded during protocol execution but becomes *computationally unbounded* after protocol execution?

Unfortunately, everlasting secure computation against *every participant in a protocol* is *also* impossible [40] for most natural two-party functionalities (or multi-party functionalities against dishonest majority corruptions). For the specific case of two parties, this means that it is impossible to achieve everlasting security against *both* players, without relying on special tools like trusted/ideal hardware. Nevertheless, it is still possible to obtain everlasting (or even the stronger notion of statistical) security against one unbounded participant (see eg., [30] and references therein). But in *all existing protocols*, which party may be unbounded and which one must be assumed to be computationally bounded must necessarily be fixed *before protocol execution*. We ask if this is necessary. That is,

Can participants transfer everlasting security from one party to another even after a protocol has already been executed?

We show that the answer is yes, under the weak cryptographic assumption that (post-quantum) statistically-binding computationally-hiding bit commitments exist. These commitments can in turn be based on one-way functions [38] or even pseudo-random quantum states [37,3].

We illustrate our novel security property by considering it in the context of Yao's classic millionaire problem [45]. Stated simply, this toy problem requires two millionaires to securely compute who is richer without revealing to each other or anyone else information about their wealth. That is, the goal is to only reveal the bit indicating whether $x_1 > x_2$ where x_1 is Alice's private input and x_2 is Bob's private input. In our extension, the millionaires would also like

(certified) everlasting security against the wealthier party, while maintaining standard simulation-based security against the other party. Namely, if $x_1 > x_2$ then the protocol should satisfy certified everlasting security against Alice and standard simulation-based security against computationally bounded Bob; and if it turns out that $x_2 \geq x_1$, then the protocol should satisfy certified everlasting security against Bob and simulation-based security against bounded Alice.

More generally, our goal is to enable any one party (or a subset of parties) to dynamically and certifiably information-theoretically delete other participants' inputs, during or even after a secure computation protocol completes. At the same time, the process of deletion should not destroy standard simulation-based security.

We build a two-party protocol that is (a) designed to be secure against computationally *unbounded Alice* and computationally *bounded Bob*. In addition, even after the protocol ends, (b) Bob has the capability to generate a proof whose validity certifies that the protocol has now become secure against *unbounded Bob* while remaining secure against *bounded Alice*. In other words, verification of the proof implies that everlasting security roles have switched: this is why we call this property *everlasting security transfer*. This implies zero-knowledge proofs for NP/QMA with certified everlasting ZK as a special case. We also extend this result to obtain *multi-party computation* where even after completion of the protocol, any arbitrary subset of parties can certifiably, information-theoretically remove information about the other party inputs from their view.

At a high level, we build these protocols by carefully combining Theorem 1 with additional techniques to ensure that having one party generate a certificate of deletion does not ruin standard (simulation-based, computational) security against the other party.

In what follows, we provide a detailed overview of our techniques.

1.2 Techniques

We first provide an overview of our proof of Theorem 1.

Our construction and analysis include a couple of crucial differences from previous work on certified deletion. First, our analysis diverges from recent work [11,39] that relies on "generalized uncertainty relations" which provide lower bounds on the sum of entropies resulting from two incompatible measurements, and instead builds on the simple but powerful "quantum cut-and-choose" formalism of Bouman and Fehr [8]. Next, we make crucial use of an *unseeded* randomness extractor (the XOR function), as opposed to a seeded extractor, as used by [11].

Delaying the dependence on b. A key tension that must be resolved when proving a claim like Theorem 1 is the following: how to *information-theoretically* remove the bit b from the adversary's view, when it is initially information-theoretically *determined* by the adversary's input. Our first step towards a proof is a simple change in perspective. We will instead imagine sampling the distribution by *guessing* a uniformly random $b' \leftarrow \{0,1\}$, and initializing the adversary with

$|x\rangle_\theta, b', \mathcal{Z}_\lambda(\theta)$. Then, we abort the experiment (output $\perp$) if it happens that $b' \neq b \oplus \bigoplus_{i:\theta_i=0} x_i$. Since b' was a uniformly random guess, we always abort with probability exactly $1/2$, and thus the trace distance between the $b = 0$ and $b = 1$ outputs of this experiment is at least half the trace distance between the outputs of the original experiment.[6]

Now, the bit b is only used by the experiment to determine whether or not to output $\perp$. This is not immediately helpful, since the result of this "abort decision" is of course included in the output of the experiment. However, we can make progress by delaying this abort decision (and thus, the dependence on b) until *after* the adversary outputs x' and their residual state on register A'. To do so, we will make use of a common strategy in quantum cryptographic proofs: replace the BB84 states $|x\rangle_\theta$ with halves of EPR pairs $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Let C be the register holding the "challenger's" halves of EPR pairs, and A be the register holding the other halves, which is part of the adversary's input. This switch is perfectly indistinguishable from the adversary's perspective, and it allows us to *delay* the measurement of C in the θ-basis (and thus, delay the determination of the string x and subsequent abort decision), until after the adversary outputs (x', A').

We still have not shown that when the deletion certificate is accepted, information about b doesn't exist in the output of the experiment. However, note that at this point it suffices to argue that $\bigoplus_{i:\theta_i=0} x_i$ is distributed like a uniformly random bit, even conditioned on the adversary's "side information" on register A' (which may be entangled with C). This is because, if $\bigoplus_{i:\theta_i=0} x_i$ is uniformly random, then the outcome of the abort decision, whether $b' = b \oplus \bigoplus_{i:\theta_i=0} x_i$, is also a uniformly random bit, regardless of b.

Identifying an efficiently-checkable predicate. To prove that $\bigoplus_{i:\theta_i=0} x_i$ is uniformly random, we will need to establish that the measured bits $\{x_i\}_{i:\theta_i=0}$ contain sufficient entropy. To do this, we will need to make some claim about the structure of the state on registers $\mathsf{C}_{i:\theta_i=0}$. These registers are measured in the computational basis to produce $\{x_i\}_{i:\theta_i=0}$, so if we could claim that these registers are in a Hadamard basis state, we would be done. We won't quite be able to claim something this strong, but we don't need to. Instead, we will rely on the following claim: consider any (potentially entangled) state on systems X and Y, such that the part of the state on system Y is in a superposition of Hadamard basis states $|u\rangle_1$ where each u is a vector of somewhat low Hamming weight.[7] Then, measuring Y in the computational basis and computing the XOR of the resulting bits produces a bit that is *uniformly random and independent*

[6] One might be concerned that extending this argument to multi-bit messages may eventually reduce the advantage by too much, since the entire message must be guessed. However, it actually suffices to prove Theorem 1 for single bit messages and then use a bit-by-bit hybrid argument to obtain security for any polynomial-length message.

[7] It suffices to require that the relative Hamming weight of each u is $< 1/2$.

of system X.[8] This claim can be viewed as saying that XOR is a good (seedless) randomness extractor for the quantum source of entropy that results from measuring certain structured states in the conjugate basis. Indeed, such a claim was developed to remove the need for *seeded* randomness extraction in applications like quantum oblivious transfer [2], and it serves a similar purpose here.[9]

Thus, it suffices to show that the state on registers $\mathsf{C}_{i:\theta_i=0}$ is only supported on low Hamming weight vectors in the Hadamard basis. A priori, it is not clear why this would even be true, since C, A are initialized with EPR pairs, and the adversary, who has access to A, can simply measure its halves of these EPR pairs in the computational basis. However, recall that the experiment we are interested in only outputs the adversary's final state when its certificate of deletion is valid, and moreover, a valid deletion certificate is a string x' that matches x in all the *Hadamard* basis positions. Moreover, which positions will be checked is semantically hidden from the adversary. Thus, in order to be sure that it passes the verification, an adversary should intuitively be measuring most of its registers A in the Hadamard basis.

Reducing to semantic security. One remaining difficulty in formalizing this intuition is that if the adversary knew θ, it could decide which positions to measure in the Hadamard basis to pass the verification check, and then measure $\mathsf{A}_{i:\theta_i=0}$ in the computational basis in order to thwart the above argument from going through. And in fact, the adversary *does* have information about θ, encoded in the distribution $\mathcal{Z}_\lambda(\theta)$.

This is where the assumption that $\mathcal{A}_\lambda$ cannot distinguish between $\mathcal{Z}_\lambda(\theta)$ and $\mathcal{Z}_\lambda(0^\lambda)$ comes into play. We interpret the condition that registers $\mathsf{C}_{i:\theta_i=0}$ must be in a superposition of low Hamming weight vectors in the Hadamard basis (or verification doesn't pass) as an efficient predicate (technically a binary projective measurement) that can be checked by a reduction to the indistinguishability of distributions $\mathcal{Z}_\lambda(\theta)$ and $\mathcal{Z}_\lambda(0^\lambda)$. Thus, this predicate must have roughly the same probability of being true when the adversary receives $\mathcal{Z}_\lambda(0^\lambda)$. But now, since θ is independent of the adversary's view, we can show *information-theoretically* that this predicate must be true with overwhelming probability.

We note that the broad strategy of identifying an efficiently-checkable predicate which implies the *uncheckable property that some information is random and independent of the adversary's view* has been used in similar (quantum cryptographic) contexts by Gottesman [22] in their work on the related concept of

[8] This proof strategy is inspired by the techniques of [8], who show a similar claim using a *seeded* extractor.

[9] If we had tried to rely on generic properties of a seeded randomness extractor, as done in [11], we would still have had to deal with the fact the adversary's view includes an encryption of the seed, which is required to be *uniform and independent* of the source of entropy. Even if the challenger's state can be shown to produce a sufficient amount of min-entropy when measured in the standard basis, we cannot immediately claim that this source of entropy is perfectly independent of the seed of the extractor. Similar issues with using seeded randomness extraction in a related context are discussed by [41] in their work on revocable timed-release encryption.

uncloneable (or perhaps more accurately, *tamper-detectable*) encryption[10] and by Unruh [41] in their work on revocable timed-release encryption.

Application: A variety of encryption schemes with certified deletion. For any $X \in$ {public-key encryption, attribute-based encryption, witness encryption, statistically-binding commitment, timed-release encryption}, we immediately obtain "X with certified deletion" by instantiating the distribution $\mathcal{Z}_\lambda$ with the encryption/encoding procedure for X, and additionally encrypting/encoding the bit $b \oplus \bigoplus_{i:\theta_i=0} x_i$ to ensure that semantic security holds regardless of whether the adversary deletes the ciphertext or not.

Similarly, if $\mathcal{Z}_\lambda$ is instantiated with the encryption procedure for a *fully-homomorphic* encryption scheme [20,9,21], then the scheme also allows for arbitrary homomorphic operations over the ciphertext. We also note that such a scheme can be used for blind delegation with certified deletion, allowing a weak client to outsource computations to a powerful server and subsequently verify deletion of the plaintext. In particular, a server may perform homomorphic evaluation coherently (i.e. by not performing any measurements), and return the register containing the output to the client. The client can coherently decrypt this register to obtain a classical outcome, then reverse the decryption operation and return the output register to the server. Finally, the server can use this register to reverse the evaluation operation and recover the original ciphertext. Then, the server can prove deletion of the original plaintext as above, i.e. measure the quantum state associated with this ciphertext in the Hadamard basis, and report the outcomes as their certificate.

Application: Secure computation with Everlasting Security Transfer (EST). Recall that in building two-party computation with EST, the goal is to build protocols (a) secure against *unbounded Alice* and computationally *bounded Bob* such that, during or even after the protocol ends, (b) Bob can generate a proof whose validity certifies that the protocol has now become secure against *unbounded Bob* while remaining secure against *bounded Alice*.

Our goal is to realize two-party secure computation with EST from minimal cryptographic assumptions. We closely inspect a class of protocols for secure computation that do not a-priori have any EST guarantees, and develop techniques to equip them with EST.

In particular, we observe that a key primitive called quantum oblivious transfer (QOT) is known to unconditionally imply secure computation of *all classical (and quantum) circuits* [31,14,16]. Namely, given OT with information-theoretic security, it is possible to build secure computation with everlasting

[10] In this notion, the adversary is an eavesdropper who sits between a ciphertext generator Alice and a ciphertext receiver Bob (using a symmetric-key encryption scheme), who attempts to learn some information about the ciphertext. The guarantee is that, *either* the eavesdropper gains information-theoretically no information about the underlying plaintext, *or* Bob can detect that the ciphertext was tampered with. While this is peripherally related to our setting, [22] does not consider public-key encryption, and moreover Bob's detection procedure is quantum.

(and even unconditional) security against unbounded participants. We recall that information-theoretically secure OT cannot exist in the plain model, even given quantum resources [33]. However, for the case of EST, we establish a general sequential composition theorem (in the full version [5]) which shows that oblivious transfer with EST can be plugged into the above unconditional protocols to yield secure computation protocols with EST.

Furthermore, a recent line of work [13,15,8,4,23] establishes *ideal* commitments[11] as the basis for QOT. Intuitively, these are commitments that satisfy the (standard) notion of simulation-based security against computationally bounded quantum committers and receivers. Namely, for every adversarial committer (resp., receiver) that interacts with an honest receiver (resp., committer) in the real protocol, there is a simulator that interacts with the ideal commitment functionality and generates a simulated state that is indistinguishable from the committer's (resp., receiver's) state in the real protocol. Our composition theorem combined with [4] also immediately shows that ideal commitments *with EST* imply QOT with EST. Thus, the problem reduces to building ideal commitments with EST.

Constructing Ideal Commitments with EST. An ideal commitment with EST satisfies statistical simulation-based security against unbounded committers, and computational simulation-based security against bounded receivers. Furthermore, after an optional delete/transfer phase succeeds, everlasting security is *transfered*: that is, then the commitment satisfies statistical (simulation-based) security against unbounded receivers, and remains computationally (simulation-based) secure against bounded committers.

To build ideal commitments with EST, we start with any commitment that satisfies standard computational hiding, and a strong form of binding: namely, simulation-based security against an unbounded malicious committer. At a high level, this means that there is an efficient extractor that can extract the input committed by an unbounded committer, thereby statistically simulating the view of the adversarial committer in its interaction with the ideal commitment functionality. We call this a *computationally-hiding statistically-efficiently-extractable* (CHSEE) commitment, and observe that prior work ([4]) builds such commitments from black-box use of any statistically-binding, computationally-hiding commitment. Our construction of ideal commitments with EST starts with CHSEE commitments, and proceeds in two steps, where the first involves new technical insights and the second follows from ideas in prior work [4].

Step 1: One-Sided Ideal Commitments with EST. While CHSEE commitments satisfy simulation-based security against a malicious committer, they do not admit security transfer. Therefore, our first step is to add the EST property to CHSEE commitments, which informally additionally allows receivers to

[11] The term "ideal committment" can sometimes refer to the commitment *ideal funtionality*, but in this work we use the term ideal commitment to refer to a *real-world protocol* that can be shown to securely implement the commitment ideal functionality.

certifiably, information-theoretically, delete the committed input. We call the resulting primitive *one-sided ideal commitments with EST*. The word "one-sided" denotes that these commitments satisfy simulation-based security against any malicious committer, but are not necessarily simulation-secure against malicious receivers. Instead, these commitments semantically hide the committed bit from a malicious receiver and furthermore, support certified everlasting hiding against malicious receivers.

We observe that invoking Theorem 1 while instantiating $\mathcal{Z}_\lambda$ with a CHSEE commitments already helps us add the certified everlasting hiding property to any CHSEE commitment. While this ensures the desired certified everlasting security against malicious receivers, the scheme appears to become insecure against malicious committers after certified deletion!

To see why, recall that the resulting commitment is now $|x\rangle_\theta, \mathsf{Com}(\theta, b')$, where Com is a CHSEE commitment and $b' = b \oplus \bigoplus_{i:\theta_i=0} x_i$. In particular, to simulate (i.e., to extract the bit committed by) a malicious committer $\mathcal{C}^*$, a simulator must extract the bases θ and masked bit b' from the CHSEE commitment, measure the accompanying state $|\psi\rangle$ in basis θ to recover x, and then XOR the parity $\bigoplus_{i:\theta_i=0} x_i$ with b' to obtain the committed bit b. Thus, the simulator will have to first measure qubits of $|\psi\rangle$ that correspond to $\theta_i = 0$ in the computational basis to recover x_i values at these positions. If the committer makes a delete request after this point, the simulator must measure *all positions* in the Hadamard basis to generate the certificate of deletion. But consider a cheating committer that (maliciously) generates the qubit at a certain position (say $i = 1$) as a half of an EPR pair, keeping the other half to itself. Next, this committer commits to $\theta_i = 0$ (i.e., computational basis) corresponding to the index $i = 1$. The simulation strategy outlined above will first measure the first qubit of $|\psi\rangle$ in the computational basis, and then later in the Hadamard basis to generate a deletion certificate. On the other hand, an honest receiver will only ever measure this qubit in the Hadamard basis to generate a deletion certificate. This makes it easy for such a committer to distinguish simulation from an honest receiver strategy, simply by measuring its half of the EPR pair in the Hadamard basis, thereby breaking simulation security post-deletion.

To prevent this attack, we modify the scheme so that the committer $\mathcal{C}^*$ *only ever obtains* the receiver's outcomes of Hadamard basis measurements on indices where the committed $\theta_i = 1$. In particular, we make the delete phase interactive: the receiver will first commit to all measurement outcomes in Hadamard bases, $\mathcal{C}^*$ will then decommit to θ, and then finally the receiver will *only* open the committed measurement outcomes on indices i where $\theta_i = 1$. Against malicious receivers, we prove that this scheme is computationally hiding before deletion, and is certified everlasting hiding after deletion. Against a malicious committer, we prove statistical simulation-based security before deletion, and show that computational simulation-based security holds *even after deletion*.

Step 2: Ideal Commitments with EST. Next, we upgrade the one-sided ideal commitments with EST obtained above to build (full-fledged) ideal com-

mitments with EST. Recall that the one-sided ideal commitments with EST do not satisfy simulation-based security against malicious receivers. Intuitively, simulation-based security against malicious receivers requires the existence of a simulator that interacts with a malicious receiver to produce a state in the commit phase, that can later be opened (or *equivocated*) to a bit that is only revealed to the simulator at the end of the commit phase. We show that this property can be generically obtained (with EST) by relying on a previous compiler, namely an *equivocality compiler* from [4]. We defer additional details of this step to the full version [5] since this essentially follows from ideas in prior work [4]. This also completes an overview of our techniques.

Roadmap. We refer the reader to Section 3 for the proof of our main theorem, Section 4 for secret sharing and public-key encryption with certified deletion, and the full version [5] for additional cryptosystems with certified deletion including details on building secure computation with everlasting security transfer.

1.3 Concurrent and independent work

Subsequent to the original posting of our paper on arXiv, an updated version of [39] was posted with some independent new results on fully-homomorphic encryption with certified deletion. The updated FHE scheme with certified deletion is shown to satisfy standard semantic security, but under a newly introduced conjecture that a particular hash function is "strong Gaussian-collapsing". Proving this conjecture based on a standard assumption such as LWE is left as an open problem in [39]. Thus, the FHE scheme presented in our paper is the first to satisfy certified deletion based on a standard assumption (and in addition satisfies *everlasting hiding*). On the other hand, the updated scheme of [39] also satisfies the property of publicly-verifiable deletion, which we do not consider in this work.

Also, a concurrent and independent work of Hiroka et al. [26] was posted shortly after the original posting of our paper. In [26], the authors construct public-key encryption schemes satisfying the definition of security that we use in this paper: certified everlasting security. However, their constructions are either in the *quantum random oracle model*, or require a *quantum* certificate of deletion. Thus, our construction of PKE with certified everlasting security, which is simple, in the plain model, and has a classical certificate of deletion, subsumes these results. On the other hand, [26] introduce and construct the primitive of (bounded-collusion) *functional encryption* with certified deletion, which we do not consider in this work.

2 Preliminaries

Let λ denote the security parameter. We write $\mathsf{negl}(\cdot)$ 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 $n > N$, $f(n) < n^{-c}$.

Given an alphabet A and string $x \in A^n$, let $h(x)$ denote the Hamming weight (number of non-zero indices) of x, and $\omega(x) \coloneqq h(x)/n$ denote the *relative* Hamming weight of x. Given two strings $x, y \in \{0,1\}^n$, let $\Delta(x,y) \coloneqq \omega(x \oplus y)$ denote the *relative Hamming distance* between x and y.

2.1 Quantum preliminaries

A register X is a named Hilbert space $\mathbb{C}^{2^n}$. A pure quantum state on register X is a unit vector $|\psi\rangle^{\mathsf{X}} \in \mathbb{C}^{2^n}$, and we say that $|\psi\rangle^{\mathsf{X}}$ consists of n qubits. A mixed state on register X is described by a density matrix $\rho^{\mathsf{X}} \in \mathbb{C}^{2^n \times 2^n}$, which is a positive semi-definite Hermitian operator with trace 1.

A *quantum operation* F is a completely-positive trace-preserving (CPTP) map from a register X to a register Y, which in general may have different dimensions. That is, on input a density matrix ρ^{X}, the operation F produces $F(\rho^{\mathsf{X}}) = \tau^{\mathsf{Y}}$ a mixed state on register Y. We will sometimes write a quantum operation F applied to a state on register X and resulting in a state on register Y as $\mathsf{Y} \leftarrow F(\mathsf{X})$. Note that we have left the actual mixed states on these registers implicit in this notation, and just work with the names of the registers themselves.

A *unitary* $U : \mathsf{X} \to \mathsf{X}$ is a special case of a quantum operation that satisfies $U^\dagger U = U U^\dagger = \mathbb{I}^{\mathsf{X}}$, where $\mathbb{I}^{\mathsf{X}}$ is the identity matrix on register X. 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 $\sum_i \Pi_i = \mathbb{I}$.

Let Tr denote the trace operator. For registers X, Y, the *partial trace* Tr^{Y} is the unique operation from X, Y to X such that for all $(\rho, \tau)^{\mathsf{X},\mathsf{Y}}$, $\mathsf{Tr}^{\mathsf{Y}}(\rho, \tau) = \mathsf{Tr}(\tau)\rho$. The *trace distance* between states ρ, τ, denoted $\mathsf{TD}(\rho, \tau)$ is defined as

$$\mathsf{TD}(\rho,\tau) \coloneqq \frac{1}{2}\|\rho - \tau\|_1 \coloneqq \frac{1}{2}\mathsf{Tr}\left(\sqrt{(\rho-\tau)^\dagger(\rho-\tau)}\right).$$

We will often use the fact that the trace distance between two states ρ and τ is an upper bound on the probability that any (unbounded) algorithm can distinguish ρ and τ. When clear from context, we will write $\mathsf{TD}(\mathsf{X}, \mathsf{Y})$ to refer to the trace distance between a state on register X and a state on register Y.

Lemma 1 (Gentle measurement [44]). *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, $\mathsf{TD}(\rho, \rho') \leq 2\sqrt{\delta}$.

We will make use of the convention that 0 denotes the computational basis $\{|0\rangle, |1\rangle\}$ and 1 denotes the Hadamard basis $\left\{\frac{|0\rangle+|1\rangle}{\sqrt{2}}, \frac{|0\rangle-|1\rangle}{\sqrt{2}}\right\}$. For a bit $r \in \{0,1\}$, we write $|r\rangle_0$ to denote r encoded in the computational basis, and $|r\rangle_1$

to denote r encoded in the Hadamard basis. For strings $x, \theta \in \{0,1\}^\lambda$, we write $|x\rangle_\theta$ to mean $|x_1\rangle_{\theta_1}, \ldots, |x_\lambda\rangle_{\theta_\lambda}$.

A non-uniform quantum polynomial-time (QPT) machine $\{\mathcal{A}_\lambda, |\psi\rangle_\lambda\}_{\lambda\in\mathbb{N}}$ is a family of polynomial-size quantum machines $\mathcal{A}_\lambda$, where each is initialized with a polynomial-size advice state $|\psi_\lambda\rangle$. Each $\mathcal{A}_\lambda$ is in general described by a CPTP map. Similar to above, when we write $\mathsf{Y} \leftarrow \mathcal{A}(\mathsf{X})$, we mean that the machine $\mathcal{A}$ takes as input a state on register X and produces as output a state on register Y, and we leave the actual descripions of these states implicit. Finally, a quantum *interactive* machine is simply a sequence of quantum operations, with designated input, output, and work registers.

2.2 The XOR extractor

We make use of a result from [2] which shows that the XOR function is a good randomness extractor from certain *quantum* sources of entropy, even given quantum side information. We include a proof here for completeness.

Imported Theorem 2 ([2]) *Let X be an n-qubit register, and consider any quantum state $|\gamma\rangle^{\mathsf{A},\mathsf{X}}$ that can be written as*

$$|\gamma\rangle^{\mathsf{A},\mathsf{X}} = \sum_{u:h(u)<n/2} |\psi_u\rangle^{\mathsf{A}} \otimes |u\rangle^{\mathsf{X}},$$

where $h(\cdot)$ denotes the Hamming weight. Let $\rho^{\mathsf{A},\mathsf{P}}$ be the mixed state that results from measuring X in the Hadamard basis to produce a string $x \in \{0,1\}^n$, and writing $\bigoplus_{i\in[n]} x_i$ into a single qubit register P. Then it holds that

$$\rho^{\mathsf{A},\mathsf{P}} = \mathsf{Tr}^{\mathsf{X}}(|\gamma\rangle\langle\gamma|) \otimes \left(\frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|\right).$$

Proof. First, write the state on registers $\mathsf{A}, \mathsf{X}, \mathsf{P}$ that results from applying Hadamard to X and writing the parity, denoted by $p(x) := \bigoplus_{i\in[n]} x_i$, to P:

$$\frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} \left(\sum_{u:h(u)<n/2} (-1)^{u\cdot x} |\psi_u\rangle^{\mathsf{A}} \right) |x\rangle^{\mathsf{X}} |p(x)\rangle^{\mathsf{P}} := \frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} |\phi_x\rangle^{\mathsf{A}} |x\rangle^{\mathsf{X}} |p(x)\rangle^{\mathsf{P}}.$$

Then, tracing out the register X, we have that

$$\begin{aligned}
\rho^{\mathsf{A},\mathsf{P}} &= \frac{1}{2^n} \sum_{x \in \{0,1\}^n} |\phi_x\rangle |p(x)\rangle \langle p(x)| \langle \phi_x| \\
&= \frac{1}{2^n} \sum_{x:p(x)=0} |\phi_x\rangle \langle \phi_x| \otimes |0\rangle \langle 0| + \frac{1}{2^n} \sum_{x:p(x)=1} |\phi_x\rangle \langle \phi_x| \otimes |1\rangle \langle 1| \\
&= \frac{1}{2^n} \sum_{x:p(x)=0} \left(\sum_{u_1,u_2:h(u_1),h(u_2)<n/2} (-1)^{(u_1 \oplus u_2)\cdot x} |\psi_{u_1}\rangle \langle \psi_{u_2}| \right) \otimes |0\rangle \langle 0| \\
&\quad + \frac{1}{2^n} \sum_{x:p(x)=1} \left(\sum_{u_1,u_2:h(u_1),h(u_2)<n/2} (-1)^{(u_1 \oplus u_2)\cdot x} |\psi_{u_1}\rangle \langle \psi_{u_2}| \right) \otimes |1\rangle \langle 1| \\
&= \sum_{u_1,u_2:h(u_1),h(u_2)<n/2} |\psi_{u_1}\rangle \langle \psi_{u_2}| \otimes \left(\frac{1}{2^n} \sum_{x:p(x)=0} (-1)^{(u_1 \oplus u_2)\cdot x} |0\rangle \langle 0| + \frac{1}{2^n} \sum_{x:p(x)=1} (-1)^{(u_1 \oplus u_2)\cdot x} |1\rangle \langle 1| \right) \\
&= \sum_{u:h(u)<n/2} |\psi_u\rangle \langle \psi_u| \otimes \left(\frac{1}{2} |0\rangle \langle 0| + \frac{1}{2} |1\rangle \langle 1| \right) \\
&= \mathsf{Tr}^{\mathsf{X}}(|\gamma\rangle \langle \gamma|) \otimes \left(\frac{1}{2} |0\rangle \langle 0| + \frac{1}{2} |1\rangle \langle 1| \right),
\end{aligned}$$

where the 5th equality is due to the following claim, plus the observation that $u_1 \oplus u_2 \neq 1^n$ for any u_1, u_2 such that $h(u_1) < n/2$ and $h(u_2) < n/2$.

Claim. For any $u \in \{0,1\}^n$ such that $u \notin \{0^n, 1^n\}$, it holds that

$$\sum_{x:p(x)=0} (-1)^{u\cdot x} = \sum_{x:p(x)=1} (-1)^{u\cdot x} = 0.$$

Proof. For any such $u \notin \{0^n, 1^n\}$, define $S_0 = \{i : u_i = 0\}$ and $S_1 = \{i : u_i = 1\}$. Then, for any $y_0 \in \{0,1\}^{|S_0|}$ and $y_1 \in \{0,1\}^{|S_1|}$, define $x_{y_0,y_1} \in \{0,1\}^n$ to be the n-bit string that is equal to y_0 when restricted to indices in S_0 and equal to y_1 when restricted to indices in S_1. Then,

$$\begin{aligned}
\sum_{x:p(x)=0} (-1)^{u\cdot x} &= \sum_{y_1 \in \{0,1\}^{|S_1|}} \sum_{y_0 \in \{0,1\}^{|S_0|} : p(x_{y_0,y_1})=0} (-1)^{u \cdot x_{y_0,y_1}} \\
&= \sum_{y_1 \in \{0,1\}^{|S_1|}} 2^{|S_0|-1} (-1)^{1^{|S_1|}\cdot y_1} = 2^{|S_0|-1} \sum_{y_1 \in \{0,1\}^{|S_1|}} (-1)^{p(y_1)} = 0,
\end{aligned}$$

where the second equality can be seen to hold by noting that for any fixed $y_1 \in \{0,1\}^{|S_1|}$, there are exactly $2^{|S_0|-1}$ strings $y_0 \in \{0,1\}^{|S_0|}$ such that the parity of x_{y_0,y_1} is 0. Finally, the same sequence of equalities can be seen to hold for $x : p(x) = 1$.

2.3 Quantum rewinding

We will make use of the following lemma from [42].

Lemma 2. *Let $\mathcal{Q}$ be a quantum circuit that takes n qubits as input and outputs a classical bit b and m qubits. For an n-qubit state $|\psi\rangle$, let $p(|\psi\rangle)$ denote the probability that $b = 0$ when executing $\mathcal{Q}$ on input $|\psi\rangle$. Let $p_0, q \in (0,1)$ and $\epsilon \in (0, 1/2)$ be such that:*

- *For every n-qubit state $|\psi\rangle$, $p_0 \leq p(|\psi\rangle)$,*
- *For every n-qubit state $|\psi\rangle$, $|p(|\psi\rangle) - q| < \epsilon$,*
- *$p_0(1-p_0) \leq q(1-q)$,*

Then, there is a quantum circuit $\widehat{\mathcal{Q}}$ of size $O\left(\frac{\log(1/\epsilon)}{4 \cdot p_0(1-p_0)}|\mathcal{Q}|\right)$, taking as input n qubits, and returning as output m qubits, with the following guarantee. For an n qubit state $|\psi\rangle$, let $\mathcal{Q}_0(|\psi\rangle)$ denote the output of $\mathcal{Q}$ on input $|\psi\rangle$ conditioned on $b = 0$, and let $\widehat{\mathcal{Q}}(|\psi\rangle)$ denote the output of $\widehat{\mathcal{Q}}$ on input $|\psi\rangle$. Then, for any n-qubit state $|\psi\rangle$,

$$\mathsf{TD}\left(\mathcal{Q}_0(|\psi\rangle), \widehat{\mathcal{Q}}(|\psi\rangle)\right) \leq 4\sqrt{\epsilon}\frac{\log(1/\epsilon)}{p_0(1-p_0)}.$$

3 Main theorem

Theorem 3. *Let $\{\mathcal{Z}_\lambda(\cdot,\cdot,\cdot)\}_{\lambda\in\mathbb{N}}$ be a quantum operation with three arguments: a λ-bit string θ, a bit b', and a λ-bit quantum register A. Let $\mathscr{A}$ be a class of adversaries[12] such that for all $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}} \in \mathscr{A}$, and for any string $\theta \in \{0,1\}^\lambda$, bit $b' \in \{0,1\}$, and state $|\psi\rangle^{\mathsf{A},\mathsf{C}}$ on λ-bit register A and arbitrary size register C,*

$$\left|\Pr[\mathcal{A}_\lambda(\mathcal{Z}_\lambda(\theta, b', \mathsf{A}), \mathsf{C}) = 1] - \Pr[\mathcal{A}_\lambda(\mathcal{Z}_\lambda(0^\lambda, b', \mathsf{A}), \mathsf{C}) = 1]\right| = \mathrm{negl}(\lambda).$$

That is, $\mathcal{Z}_\lambda$ is semantically-secure against $\mathcal{A}_\lambda$ with respect to its first input. For any $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}} \in \mathscr{A}$, 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 $x, \theta \leftarrow \{0,1\}^\lambda$ and initialize $\mathcal{A}_\lambda$ with*

$$\mathcal{Z}_\lambda\left(\theta, b \oplus \bigoplus_{i:\theta_i=0} x_i, |x\rangle_\theta\right).$$

- *$\mathcal{A}_\lambda$'s output is parsed as a string $x' \in \{0,1\}^\lambda$ and a residual state on register A'.*

[12] Technically, we require that for any $\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}} \in \mathscr{A}$, every adversary $\mathcal{B}$ with time and space complexity that is linear in λ more than that of $\mathcal{A}_\lambda$, is also in $\mathscr{A}$.

– *If $x_i = x'_i$ for all i such that $\theta_i = 1$ then output A', and otherwise output a special symbol $\perp$.*

Then,

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

Remark 1. We note that, in fact, the above theorem is true as long as x, θ are $\omega(\mathsf{log}\lambda)$ bits long.

Proof. We define a sequence of hybrid distributions.

– $\mathsf{Hyb}_0(b)$: This is the distribution $\left\{\widetilde{\mathcal{Z}}_\lambda^{\mathcal{A}_\lambda}(b)\right\}_{\lambda\in\mathbb{N}}$ described above.
– $\mathsf{Hyb}_1(b)$: This distribution is sampled as follows.
  • Prepare λ EPR pairs $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ on registers $(\mathsf{C}_1, \mathsf{A}_1), \ldots, (\mathsf{C}_\lambda, \mathsf{A}_\lambda)$. Define $\mathsf{C} \coloneqq \mathsf{C}_1, \ldots, \mathsf{C}_\lambda$ and $\mathsf{A} \coloneqq \mathsf{A}_1, \ldots, \mathsf{A}_\lambda$.
  • Sample $\theta \leftarrow \{0,1\}^\lambda, b' \leftarrow \{0,1\}$, measure register C in basis θ to obtain $x \in \{0,1\}^\lambda$, and initialize $\mathcal{A}_\lambda$ with $\mathcal{Z}_\lambda(\theta, b', \mathsf{A})$.
  • If $b' = b \oplus \bigoplus_{i:\theta_i=0} x_i$ then proceed as in Hyb_0 and otherwise output $\perp$.
– $\mathsf{Hyb}_2(b)$: This is the same as $\mathsf{Hyb}_1(b)$ except that measurement of register C to obtain x is performed after $\mathcal{A}_\lambda$ outputs x' and ρ.

We define $\mathsf{Advt}(\mathsf{Hyb}_i) \coloneqq \mathsf{TD}\left(\mathsf{Hyb}_i(0), \mathsf{Hyb}_i(1)\right)$. Then, we have that

$$\mathsf{Advt}(\mathsf{Hyb}_1) \geq \mathsf{Advt}(\mathsf{Hyb}_0)/2,$$

which follows because $\mathsf{Hyb}_1(b)$ is identically distributed to the distribution that outputs $\perp$ with probability $1/2$ and otherwise outputs $\mathsf{Hyb}_0(b)$. Next, we have that

$$\mathsf{Advt}(\mathsf{Hyb}_2) = \mathsf{Advt}(\mathsf{Hyb}_1),$$

which follows because the register C is disjoint from the registers that $\mathcal{A}_\lambda$ operates on. Thus, it remains to show that

$$\mathsf{Advt}(\mathsf{Hyb}_2) = \mathrm{negl}(\lambda).$$

To show this, we first define the following hybrid.

– $\mathsf{Hyb}'_2(b)$: This is the same as Hyb_2 except that $\mathcal{A}_\lambda$ is initialized with $\mathcal{Z}_\lambda(0^\lambda, b', \mathsf{A})$.

Now, for any $b \in \{0,1\}$, consider the state on register C immediately after $\mathcal{A}_\lambda$ outputs (x', A') in $\mathsf{Hyb}'_2(b)$. For any $\theta \in \{0,1\}^\lambda$, define sets $\theta_0 \coloneqq \{i : \theta_i = 0\}$ and $\theta_1 \coloneqq \{i : \theta_i = 1\}$, and define the projector

$$\Pi_{x',\theta} \coloneqq \left(H^{\otimes|\theta_1|}\,|x'_{\theta_1}\rangle\,\langle x'_{\theta_1}|\,H^{\otimes|\theta_1|}\right)^{\mathsf{C}_{\theta_1}} \otimes \sum_{\substack{y\in\{0,1\}^{|\theta_0|} \text{ s.t.} \\ \Delta\left(y, x'_{\theta_0}\right)\geq 1/2}} \left(H^{\otimes|\theta_0|}\,|y\rangle\,\langle y|\,H^{\otimes|\theta_0|}\right)^{\mathsf{C}_{\theta_0}},$$

where $\Delta(\cdot,\cdot)$ denotes relative Hamming distance. Then, let $\Pr[\Pi_{x',\theta}, \mathsf{Hyb}'_2(b)]$ be the probability that a measurement of $\{\Pi_{x',\theta}, \mathbb{I} - \Pi_{x',\theta}\}$ accepts (returns the outcome associated with $\Pi_{x',\theta}$) in $\mathsf{Hyb}'_2(b)$.

Claim. For any $b \in \{0,1\}$, $\Pr[\Pi_{x',\theta}, \mathsf{Hyb}'_2(b)] = \mathrm{negl}(\lambda)$.

Proof. Consider running $\mathsf{Hyb}'_2(b)$ until $\mathcal{A}_\lambda$ outputs x' and a state on register A' that may be entangled with the challenger's state on register C. Note that we can sample $\theta \leftarrow \{0,1\}^\lambda$ independently since it is no longer in $\mathcal{A}_\lambda$'s view. Then since $\Pi_{x',\theta}$ is diagonal in the Hadamard basis for any (x',θ), we have that

$$\Pr[\Pi_{x',\theta}, \mathsf{Hyb}'_2(b)] = \Pr_{x',\theta,y}\left[y_{\theta_1} = x'_{\theta_1} \wedge \Delta\left(y_{\theta_0}, x'_{\theta_0}\right) \geq 1/2\right],$$

where the second probability is over $\mathcal{A}_\lambda$ outputting x', the challenger sampling $\theta \leftarrow \{0,1\}^\lambda$, and the challenger measuring register C in the Hadamard basis to obtain y. For any fixed string x', this probability can be bound by standard Hoeffding inequalities. For example, in [8, Appendix B.3], it is shown to be bounded by $4e^{-\lambda(1/2)^2/32} = \mathrm{negl}(\lambda)$, which completes the proof.

Now we consider the corresponding event in $\mathsf{Hyb}_2(b)$, denoted $\Pr[\Pi_{x',\theta}, \mathsf{Hyb}_2(b)]$.

Claim. For any $b \in \{0,1\}$, $\Pr[\Pi_{x',\theta}, \mathsf{Hyb}_2(b)] = \mathrm{negl}(\lambda)$.

Proof. This follows by a direct reduction to semantic security of $\{\mathcal{Z}_\lambda(\cdot,\cdot,\cdot)\}_{\lambda\in\mathbb{N}}$ with respect to its first input. The reduction samples $\theta \leftarrow \{0,1\}^\lambda$, $b' \leftarrow \{0,1\}$, prepares λ EPR pairs on registers (A,C), and sends (θ, b', A) to its challenger. It receives either $\mathcal{Z}_\lambda(\theta, b', \mathsf{A})$ or $\mathcal{Z}_\lambda(0^\lambda, b', \mathsf{A})$, which its sends to $\mathcal{A}_\lambda$. After $\mathcal{A}_\lambda$ outputs (x', A'), the reduction measures $\{\Pi_{x',\theta}, \mathbb{I} - \Pi_{x',\theta}\}$ on register C. Note that the complexity of this reduction is equal to the complexity of $\mathcal{A}_\lambda$ plus an extra λ bits of space and an extra linear time operation, so it is still in $\mathscr{A}$. If $\Pr[\Pi_{x',\theta}, \mathsf{Hyb}_2(b)]$ is non-negligible this can be used to distinguish $\mathcal{Z}_\lambda(\theta, b', \mathsf{A})$ from $\mathcal{Z}_\lambda(0^\lambda, b', \mathsf{A})$, due to Section 3.

Finally, we can show the following claim, which completes the proof.

Claim. $\mathsf{Advt}(\mathsf{Hyb}_2) = \mathrm{negl}(\lambda)$.

Proof. First, we note that for any $b \in \{0,1\}$, the global state of $\mathsf{Hyb}_2(b)$ immediately after $\mathcal{A}_\lambda$ outputs x' is within negligible trace distance of a state $\tau^{\mathsf{C},\mathsf{A}'}_{\mathsf{Ideal}}$ in the image of $\mathbb{I} - \Pi_{x',\theta}$. This follows immediately from Section 3 and Gentle Measurement (Lemma 1). Now, consider measuring registers C_{θ_1} of $\tau^{\mathsf{C},\mathsf{A}'}_{\mathsf{Ideal}}$ to determine whether the experiment outputs $\perp$. That is, the procedure measures C_{θ_1} in the Hadamard basis and checks if the resulting string is equal to x'_{θ_1}. There are two options.

- If the measurement fails, then the experiment outputs $\perp$, independent of whether $b = 0$ or $b = 1$, so there is 0 advantage in this case.
- If the measurement succeeds, then we know that the state on register C_{θ_0} is only supported on vectors $H^{\otimes|\theta_0|}|y\rangle$ such that $\Delta(y, x'_{\theta_0}) < 1/2$, since $\tau^{\mathsf{C},\mathsf{A}'}_{\mathsf{Ideal}}$ was in the image of $\mathbb{I} - \Pi_{x',\theta}$. These registers are then measured in the computational basis to produce bits $\{x_i\}_{i:\theta_i=0}$, and the experiment outputs

$\bot$ if $\bigoplus_{i:\theta_i=0} x_i \neq b' \oplus b$ and otherwise outputs the state on register A'. Note that (i) this decision is the *only* part of the experiment that depends on b, and (ii) it follows from Theorem 2 that the bit $\bigoplus_{i:\theta_i=0} x_i$ is *uniformly random and independent* of the register A', which is disjoint (but possibly entangled with) C. Thus, there is also 0 advantage in this case.
Indeed, Theorem 2 says that making a Hadamard basis measurement of a register that is in a superposition of computational basis vectors with relative Hamming weight $< 1/2$ will produce a set of bits $\{x_i\}_{i:\theta_i=0}$ such that $\bigoplus_{i:\theta_i=0} x_i$ is a uniformly random bit, even given potentially entangled quantum side information. We can apply this lemma to our system on $\mathsf{C}_{\theta_0}, \mathsf{A}'$ by considering a change of basis that maps $H^{\otimes|\theta_0|}\,|x'_{\theta_0}\rangle \to |0^{|\theta_0|}\rangle$. That is, the change of basis first applies Hadamard gates, and then an XOR with the fixed string x'_{θ_0}. Applying such a change of basis maps C_{θ_0} to a state that is supported on vectors $|y\rangle$ such that $\omega(y) < 1/2$, and we want to claim that a *Hadamard* basis measurement of the resulting state produces $\{x_i\}_{i:\theta_i=0}$ such that $\bigoplus_{i:\theta_i=0} x_i$ is uniformly random and independent of A'. This is exactly the statement of Theorem 2.

This completes the proof, since we have shown that there exists a single distribution, defined by $\tau^{\mathsf{C},\mathsf{A}'}_{\mathsf{Ideal}}$, that is negligibly close to both $\mathsf{Hyb}_2(0)$ and $\mathsf{Hyb}_2(1)$.

4 Cryptography with Certified Everlasting Security

4.1 Secret sharing

We give a simple construction of a 2-out-of-2 secret sharing scheme where there exists a designated party that the dealer can ask to produce a certificate of deletion of their share. If this certificate verifies, then the underlying plaintext is information theoretically deleted, even given the other share.

Definition. First, we augment the standard syntax of secret sharing to include a deletion algorithm Del and a verification algorithm Ver. Formally, consider a secret sharing scheme $\mathsf{CD\text{-}SS} = (\mathsf{Share}, \mathsf{Rec}, \mathsf{Del}, \mathsf{Ver})$ with the following syntax.

- $\mathsf{Share}(m) \to (s_1, s_2, \mathsf{vk})$ is a quantum algorithm that takes as input a classical message m, and outputs a quantum share s_1, a classical share s_2 and a (potentially quantum) verification key vk.
- $\mathsf{Rec}(s_1, s_2) \to \{m, \bot\}$ is a quantum algorithm that takes as input two shares and outputs either a message m or a $\bot$ symbol.
- $\mathsf{Del}(s_1) \to \mathsf{cert}$ is a quantum algorithm that takes as input a quantum share s_1 and outputs a (potentially quantum) deletion certificate cert.
- $\mathsf{Ver}(\mathsf{vk}, \mathsf{cert}) \to \{\top, \bot\}$ is a (potentially quantum) algorithm that takes as input a (potentially quantum) verification key vk and a (potentially quantum) deletion certificate cert and outputs either $\top$ or $\bot$.

We say that $\mathsf{CD\text{-}SS}$ satisfies correctness of deletion if the following holds.

Definition 1 (Correctness of deletion). $\mathsf{CD\text{-}SS} = (\mathsf{Share}, \mathsf{Rec}, \mathsf{Del}, \mathsf{Ver})$ *satisfies* correctness of deletion *if for any m, it holds with $1 - \mathrm{negl}(\lambda)$ probability over $(s_1, s_2, \mathsf{vk}) \leftarrow \mathsf{Share}(m), \mathsf{cert} \leftarrow \mathsf{Del}(s_1), \mu \leftarrow \mathsf{Ver}(\mathsf{vk}, \mathsf{cert})$ that $\mu = \top$.*

Next, we define certified deletion security for a secret sharing scheme.

Definition 2 (Certified deletion security).
Let $\mathcal{A} = \{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$ denote an unbounded adversary and b denote a classical bit. Consider experiment $\mathsf{EV\text{-}EXP}^{\mathcal{A}}_{\lambda}(b)$ which describes everlasting security given a deletion certificate, and is defined as follows.

- *Sample $(s_1, s_2, \mathsf{vk}) \leftarrow \mathsf{Share}(b)$.*
- *Initialize $\mathcal{A}_\lambda$ with s_1.*
- *Parse $\mathcal{A}_\lambda$'s output as a deletion certificate cert and a residual state on register A'.*
- *If $\mathsf{Ver}(\mathsf{vk}, \mathsf{cert}) = \top$ then output (A', s_2), and otherwise output $\bot$.*

Then $\mathsf{CD\text{-}SS} = (\mathsf{Share}, \mathsf{Rec}, \mathsf{Del}, \mathsf{Ver})$ satisfies certified deletion security *if for any unbounded adversary $\mathcal{A}$, it holds that*

$$\mathsf{TD}\left(\mathsf{EV\text{-}EXP}^{\mathcal{A}}_{\lambda}(0), \mathsf{EV\text{-}EXP}^{\mathcal{A}}_{\lambda}(1)\right) = \mathrm{negl}(\lambda),$$

Corollary 1. *The scheme $\mathsf{CD\text{-}SS} = (\mathsf{Share}, \mathsf{Rec}, \mathsf{Del}, \mathsf{Ver})$ defined as follows is a secret sharing scheme with certified deletion.*

- $\mathsf{Share}(m)$: *sample $x, \theta \leftarrow \{0,1\}^\lambda$ and output*

$$s_1 \coloneqq |x\rangle_\theta, s_2 \coloneqq \left(\theta, b \oplus \bigoplus_{i:\theta_i=0} x_i\right), \quad \mathsf{vk} \coloneqq (x, \theta).$$

- $\mathsf{Rec}(s_1, s_2)$: *parse $s_1 \coloneqq |x\rangle_\theta, s_2 \coloneqq (\theta, b')$, measure $|x\rangle_\theta$ in the θ-basis to obtain x, and output $b = b' \oplus \bigoplus_{i:\theta_i=0} x_i$.*
- $\mathsf{Del}(s_1)$: *parse $s_1 \coloneqq |x\rangle_\theta$ and measure $|x\rangle_\theta$ in the Hadamard basis to obtain a string x', and output $\mathsf{cert} \coloneqq x'$.*
- $\mathsf{Ver}(\mathsf{vk}, \mathsf{cert})$: *parse vk as (x, θ) and cert as x' and output $\top$ if and only if $x_i = x'_i$ for all i such that $\theta_i = 1$.*

Proof. Correctness of deletion follows immediately from the description of the scheme. Certified deletion security, i.e.

$$\mathsf{TD}\left(\mathsf{EV\text{-}EXP}^{\mathcal{A}}_{\lambda}(0), \mathsf{EV\text{-}EXP}^{\mathcal{A}}_{\lambda}(1)\right) = \mathrm{negl}(\lambda)$$

follows by following the proof strategy of Theorem 3. This setting is slightly different than the setting considered in the proof of Theorem 3 since here we consider unbounded $\mathcal{A}_\lambda$ that are not given access to θ while Theorem 3 considers bounded $\mathcal{A}_\lambda$ that are given access to an encryption of θ. However, the proof is almost identical, defining hybrids as follows.

$\mathsf{Hyb}_0(b)$: This is the distribution $\left\{\mathsf{EV\text{-}EXP}_\lambda^{\mathcal{A}_\lambda}(b)\right\}_{\lambda\in\mathbb{N}}$ described above.
$\mathsf{Hyb}_1(b)$: This distribution is sampled as follows.

- Prepare λ EPR pairs $\frac{1}{\sqrt{2}}(|00\rangle+|11\rangle)$ on registers $(\mathsf{C}_1,\mathsf{A}_1),\ldots,(\mathsf{C}_\lambda,\mathsf{A}_\lambda)$. Define $\mathsf{C}:=\mathsf{C}_1,\ldots,\mathsf{C}_\lambda$ and $\mathsf{A}:=\mathsf{A}_1,\ldots,\mathsf{A}_\lambda$.
- Sample $\theta\leftarrow\{0,1\}, b'\leftarrow\{0,1\}$, measure register C in basis θ to obtain $x\in\{0,1\}^\lambda$, and initialize $\mathcal{A}_\lambda$ with register A.
- If $b'=b\oplus\bigoplus_{i:\theta_i=0}x_i$ then proceed as in Hyb_0 and otherwise output $\bot$.

$\mathsf{Hyb}_2(b)$: This is the same as $\mathsf{Hyb}_1(b)$ except that measurement of register C to obtain x is performed after $\mathcal{A}_\lambda$ outputs x' and A'.

Indistinguishability between these hybrids closely follows the proof of Theorem 3. The key difference is that $\mathsf{Hyb}'_2(b)$ is identical to $\mathsf{Hyb}_2(b)$ except that s_2 is set to $(b',0^\lambda)$. Then, $\Pr[\Pi_{x',\theta},\mathsf{Hyb}'_2(b)]=\mathrm{negl}(\lambda)$ follows identically to the proof in Theorem 3, whereas $\Pr[\Pi_{x',\theta},\mathsf{Hyb}_2(b)]=\mathrm{negl}(\lambda)$ follows because the view of $\mathcal{A}_\lambda$ is identical in both hybrids. The final claim, that $\mathsf{Advt}(\mathsf{Hyb}_2)=\mathrm{negl}(\lambda)$ follows identically to the proof in Theorem 3.

Remark 2 (One-time pad encryption). We observe that the above proof, which considers unbounded $\mathcal{A}_\lambda$ who don't have access to θ until after they produce a valid deletion certificate, can also be used to establish the security of a simple one-time pad encryption scheme with certified deletion. The encryption of a bit b would be the state $|x\rangle_\theta$ together with a one-time pad encryption $k\oplus b\oplus\bigoplus_{i:\theta_i=0}x_i$ with key $k\leftarrow\{0,1\}$. The secret key would be (k,θ). Semantic security follows from the one-time pad, while certified deletion security follows from the above secret-sharing proof. This somewhat simplifies the construction of one-time pad encryption with certified deletion of [11], who required a seeded extractor.

4.2 Public-key encryption

In this section, we define and construct post-quantum public-key encryption with certified deletion for classical messages, assuming the existence of post-quantum public-key encryption for classical messages.

Public-Key encryption with certified deletion. First, we augment the standard syntax to include a deletion algorithm Del and a verification algorithm Ver. Formally, consider a public-key encryption scheme $\mathsf{CD\text{-}PKE}=(\mathsf{Gen},\mathsf{Enc},\mathsf{Dec},\mathsf{Del},\mathsf{Ver})$ with syntax

- $\mathsf{Gen}(1^\lambda)\to(\mathsf{pk},\mathsf{sk})$ is a classical algorithm that takes as input the security parameter and outputs a public key pk and secret key sk.
- $\mathsf{Enc}(\mathsf{pk},m)\to(\mathsf{ct},\mathsf{vk})$ is a quantum algorithm that takes as input the public key pk and a message m, and outputs a (potentially quantum) verification key vk and a quantum ciphertext ct.
- $\mathsf{Dec}(\mathsf{sk},\mathsf{ct})\to\{m,\bot\}$ is a quantum algorithm that takes as input the secret key sk and a quantum ciphertext ct and outputs either a message m or a $\bot$ symbol.

– $\mathsf{Del}(\mathsf{ct}) \to \mathsf{cert}$ is a quantum algorithm that takes as input a quantum ciphertext ct and outputs a (potentially quantum) deletion certificate cert.
– $\mathsf{Ver}(\mathsf{vk}, \mathsf{cert}) \to \{\top, \bot\}$ is a (potentially quantum) algorithm that takes as input a (potentially quantum) verification key vk and a (potentially quantum) deletion certificate cert and outputs either $\top$ or $\bot$.

We say that CD-PKE satisfies correctness of deletion if the following holds.

Definition 3 (Correctness of deletion). $\mathsf{CD\text{-}PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Del}, \mathsf{Ver})$ *satisfies* correctness of deletion *if for any m, it holds with $1 - \mathrm{negl}(\lambda)$ probability over $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda), (\mathsf{ct}, \mathsf{vk}) \leftarrow \mathsf{Enc}(\mathsf{pk}, m), \mathsf{cert} \leftarrow \mathsf{Del}(\mathsf{ct}), \mu \leftarrow \mathsf{Ver}(\mathsf{vk}, \mathsf{cert})$ that $\mu = \top$.*

Next, we define certified deletion security. Our definition has multiple parts, which we motivate as follows. The first experiment is the everlasting security experiment, which requires that conditioned on the (computationally bounded) adversary producing a valid deletion certificate, their left-over state is information-theoretically independent of b. However, we still want to obtain meaningful guarantees against adversaries that do not produce a valid deletion certificate. That is, we hope for standard semantic security against arbitrarily malicious but computationally bounded adversaries. Since such an adversary can query the ciphertext generator with an arbitrarily computed deletion certificate, we should include this potential interaction in the definition, and require that the response from the ciphertext generator still does not leak any information about b.[13] Note that, in our constructions, the verification key vk is actually completely independent of the plaintext b, and thus for our schemes this property follows automatically from semantic security.

Definition 4 (Certified deletion security). $\mathsf{CD\text{-}PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Del}, \mathsf{Ver})$ *satisfies* certified deletion security *if for any non-uniform QPT adversary $\mathcal{A} = \{\mathcal{A}_\lambda, |\psi\rangle_\lambda\}_{\lambda\in\mathbb{N}}$, it holds that*

$$\mathsf{TD}\left(\mathsf{EV\text{-}EXP}^{\mathcal{A}}_\lambda(0), \mathsf{EV\text{-}EXP}^{\mathcal{A}}_\lambda(1)\right) = \mathrm{negl}(\lambda),$$

and

$$\left|\Pr\left[\mathsf{C\text{-}EXP}^{\mathcal{A}}_\lambda(0) = 1\right] - \Pr\left[\mathsf{C\text{-}EXP}^{\mathcal{A}}_\lambda(1) = 1\right]\right| = \mathrm{negl}(\lambda),$$

where the experiment $\mathsf{EV\text{-}EXP}^{\mathcal{A}}_\lambda(b)$ considers everlasting security given a deletion certificate, and is defined as follows.

– *Sample $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$ and $(\mathsf{ct}, \mathsf{vk}) \leftarrow \mathsf{Enc}(\mathsf{pk}, b)$.*

[13] One might expect that the everlasting security definition described above already captures this property, since whether the certificate accepts or rejects is included in the output of the experiment. However, this experiment does not include the output of the adversary in the case that the certificate is rejected. So we still need to capture the fact that the *joint* distribution of the final adversarial state and the bit indicating whether the verification passes semantically hides b.

– *Initialize* $\mathcal{A}_\lambda(|\psi_\lambda\rangle)$ *with* pk *and* ct.
– *Parse* $\mathcal{A}_\lambda$*'s output as a deletion certificate* cert *and a residual state on register* A'.
– *If* $\mathsf{Ver}(\mathsf{vk},\mathsf{cert}) = \top$ *then output* A'*, and otherwise output* $\bot$.

and the experiment $\mathsf{C\text{-}EXP}^{\mathcal{A}}_{\lambda}(b)$ *is a strengthening of semantic security, defined as follows.*

– *Sample* $(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$ *and* $(\mathsf{ct},\mathsf{vk}) \leftarrow \mathsf{Enc}(\mathsf{pk},b)$.
– *Initialize* $\mathcal{A}_\lambda(|\psi_\lambda\rangle)$ *with* pk *and* ct.
– *Parse* $\mathcal{A}_\lambda$*'s output as a deletion certificate* cert *and a residual state on register* A'.
– *Output* $\mathcal{A}_\lambda\left(\mathsf{A}', \mathsf{Ver}(\mathsf{vk},\mathsf{cert})\right)$.

Now we can formally define the notion of public-key encryption with certified deletion.

Definition 5 (Public-key encryption with certified deletion). $\mathsf{CD\text{-}PKE} = (\mathsf{Gen},\mathsf{Enc},\mathsf{Dec},\mathsf{Del},\mathsf{Ver})$ *is a secure* public-key encryption scheme with certified deletion *if it satisfies (i) correctness of deletion (Definition 3), and (ii) certified deletion security (Definition 4).*

Then, we have the following corollary of Theorem 3.

Corollary 2. *Given any post-quantum semantically-secure public-key encryption scheme* $\mathsf{PKE} = (\mathsf{Gen},\mathsf{Enc},\mathsf{Dec})$*, the scheme* $\mathsf{CD\text{-}PKE} = (\mathsf{Gen},\mathsf{Enc}',\mathsf{Dec}',\mathsf{Del},\mathsf{Ver})$ *defined as follows is a public-key encryption scheme with certified deletion.*

– $\mathsf{Enc}'(\mathsf{pk},m)$: *sample* $x,\theta \leftarrow \{0,1\}^\lambda$ *and output*

$$\mathsf{ct} := \left(|x\rangle_\theta, \mathsf{Enc}\left(\mathsf{pk}, \left(\theta, b \oplus \bigoplus_{i:\theta_i=0} x_i\right)\right)\right), \quad \mathsf{vk} := (x,\theta).$$

– $\mathsf{Dec}'(\mathsf{sk},\mathsf{ct})$: *parse* $\mathsf{ct} := (|x\rangle_\theta, \mathsf{ct}')$*, compute* $(\theta,b') \leftarrow \mathsf{Dec}(\mathsf{sk},\mathsf{ct}')$*, measure* $|x\rangle_\theta$ *in the* θ*-basis to obtain* x*, and output* $b = b' \oplus \bigoplus_{i:\theta_i=0} x_i$.
– $\mathsf{Del}(\mathsf{ct})$: *parse* $\mathsf{ct} := (|x\rangle_\theta, \mathsf{ct}')$ *and measure* $|x\rangle_\theta$ *in the Hadamard basis to obtain a string* x'*, and output* $\mathsf{cert} := x'$.
– $\mathsf{Ver}(\mathsf{vk},\mathsf{cert})$: *parse* vk *as* (x,θ) *and* cert *as* x' *and output* $\top$ *if and only if* $x_i = x'_i$ *for all* i *such that* $\theta_i = 1$.

Proof. Correctness of deletion follows immediately from the description of the scheme. For certified deletion security, we consider the following:

– First, we observe that

$$\mathsf{TD}\left(\mathsf{EV\text{-}EXP}^{\mathcal{A}}_{\lambda}(0), \mathsf{EV\text{-}EXP}^{\mathcal{A}}_{\lambda}(1)\right) = \mathrm{negl}(\lambda)$$

follows from Theorem 3 and the semantic security of PKE by setting the distribution $\mathcal{Z}_\lambda(\theta,b',\mathsf{A})$ to sample $(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$, and output $(\mathsf{A},\mathsf{Enc}(\mathsf{pk},(\theta,b')))$, and setting the class of adversaries $\mathscr{A}$ to be all non-uniform families of QPT adversaries $\{\mathcal{A}_\lambda,|\psi_\lambda\rangle\}_{\lambda\in\mathbb{N}}$.

– Next, we observe that

$$\left| \Pr\left[\mathsf{C\text{-}EXP}_\lambda^{\mathcal{A}}(0) = 1\right] - \Pr\left[\mathsf{C\text{-}EXP}_\lambda^{\mathcal{A}}(1) = 1\right] \right| = \mathrm{negl}(\lambda)$$

follows from the fact that the encryption scheme remains (computationally) semantically secure even when the adversary is given the verification key x corresponding to the challenge ciphertext, since the bit b remains encrypted with Enc.

This completes our proof.

The notion of certified deletion security can be naturally generalized to consider multi-bit messages, as follows.

Definition 6 (Certified deletion security for multi-bit messages). $\mathsf{CD\text{-}PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Del}, \mathsf{Ver})$ *satisfies* certified deletion security *if for any non-uniform QPT adversary* $\mathcal{A} = \{\mathcal{A}_\lambda, |\psi\rangle_\lambda\}_{\lambda\in\mathbb{N}}$*, it holds that*

$$\mathsf{TD}\left(\mathsf{EV\text{-}EXP}_\lambda^{\mathcal{A}}(0), \mathsf{EV\text{-}EXP}_\lambda^{\mathcal{A}}(1)\right) = \mathrm{negl}(\lambda),$$

and

$$\left| \Pr\left[\mathsf{C\text{-}EXP}_\lambda^{\mathcal{A}}(0) = 1\right] - \Pr\left[\mathsf{C\text{-}EXP}_\lambda^{\mathcal{A}}(1) = 1\right] \right| = \mathrm{negl}(\lambda),$$

where the experiment $\mathsf{EV\text{-}EXP}_\lambda^{\mathcal{A}}(b)$ *considers everlasting security given a deletion certificate, and is defined as follows.*

- *Sample* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$*. Initialize* $\mathcal{A}_\lambda(|\psi_\lambda\rangle)$ *with* pk *and parse its output as* (m_0, m_1)*.*
- *Sample* $(\mathsf{ct}, \mathsf{vk}) \leftarrow \mathsf{Enc}(\mathsf{pk}, m_b)$*.*
- *Run* $\mathcal{A}_\lambda$ *on input* ct *and parse* $\mathcal{A}_\lambda$*'s output as a deletion certificate* cert*, and a residual state on register* A'*.*
- *If* $\mathsf{Ver}(\mathsf{vk}, \mathsf{cert}) = \top$ *then output* A'*, and otherwise output* $\bot$*.*

and the experiment $\mathsf{C\text{-}EXP}_\lambda^{\mathcal{A}}(b)$ *is a strengthening of semantic security, defined as follows.*

- *Sample* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$*. Initialize* $\mathcal{A}_\lambda(|\psi_\lambda\rangle)$ *with* pk *and parse its output as* (m_0, m_1)*.*
- *Sample* $(\mathsf{ct}, \mathsf{vk}) \leftarrow \mathsf{Enc}(\mathsf{pk}, m_b)$*.*
- *Run* $\mathcal{A}_\lambda$ *on input* ct *and parse* $\mathcal{A}_\lambda$*'s output as a deletion certificate* cert*, and a residual state on register* A'*.*
- *Output* $\mathcal{A}_\lambda\,(\mathsf{A}', \mathsf{Ver}(\mathsf{vk}, \mathsf{cert}))$*.*

A folklore method converts any public-key bit encryption scheme to a public-key string encryption scheme, by separately encrypting each bit in the underlying string one-by-one and appending all resulting ciphertexts. Semantic security of the resulting public-key encryption scheme follows by a hybrid argument, where

one considers intermediate hybrid experiments that only modify one bit of the underlying plaintext at a time. We observe that the same transformation from bit encryption to string encryption also preserves certified deletion security, and this follows by a similar hybrid argument. That is, as long as the encryption scheme for bits satisfies certified deletion security for single-bit messages per Definition 4, the resulting scheme for multi-bit messages satisfies certified deletion security according to Definition 6.

In the full version [5], we show how to build on this framework to obtain several advanced primitives with certified everlasting security, including *attribute-based encryption* and *fully-homormphic encryption*.

Acknowledgments

We thank Bhaskar Roberts and Alex Poremba for comments on an earlier draft, and for noting that quantum fully-homomorphic encryption is not necessary for our FHE with certified deletion scheme, classical fully-homomorphic encryption suffices.

D.K. was supported in part by DARPA SIEVE, NSF QIS-2112890 and NSF CNS-2247727. This material is based on work supported by DARPA under Contract No. HR001120C0024. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Government or DARPA.

References

1. California Consumer Privacy Act (CCPA) (2018)
2. Agarwal, A., Bartusek, J., Khurana, D., Kumar, N.: A new framework for quantum oblivious transfer. CoRR **abs/2209.04520** (2022). https://doi.org/10.48550/arXiv.2209.04520, https://doi.org/10.48550/arXiv.2209.04520
3. Ananth, P., Qian, L., Yuen, H.: Cryptography from pseudorandom quantum states. To appear in CRYPTO (2022), https://ia.cr/2021/1663
4. Bartusek, J., Coladangelo, A., Khurana, D., Ma, F.: One-way functions imply secure computation in a quantum world. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology – CRYPTO 2021. pp. 467–496. Springer International Publishing, Cham (2021)
5. Bartusek, J., Khurana, D.: Cryptography with certified deletion. Cryptology ePrint Archive, Paper 2022/1178 (2022), https://eprint.iacr.org/2022/1178, https://eprint.iacr.org/2022/1178
6. Bennett, C.H., Brassard, G.: Quantum cryptography: Public key distribution and coin tossing. In: Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing. pp. 175–179 (1984)
7. Biehl, I., Meyer, B., Wetzel, S.: Ensuring the integrity of agent-based computations by short proofs. In: Rothermel, K., Hohl, F. (eds.) Mobile Agents, Second International Workshop, MA'98, Stuttgart, Germany, September 1998, Proceedings. Lecture Notes in Computer Science, vol. 1477, pp. 183–194. Springer (1998). https://doi.org/10.1007/BFb0057658, https://doi.org/10.1007/BFb0057658

8. Bouman, N.J., Fehr, S.: Sampling in a quantum population, and applications. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 724–741. Springer, Heidelberg (Aug 2010). https://doi.org/10.1007/978-3-642-14623-7_39
9. Brakerski, Z., Vaikuntanathan, V.: Efficient fully homomorphic encryption from (standard) lwe. In: 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science. pp. 97–106 (2011). https://doi.org/10.1109/FOCS.2011.12
10. Brassard, G., Crépeau, C., Jozsa, R., Langlois, D.: A quantum bit commitment scheme provably unbreakable by both parties. In: 34th FOCS. pp. 362–371. IEEE Computer Society Press (Nov 1993). https://doi.org/10.1109/SFCS.1993.366851
11. Broadbent, A., Islam, R.: Quantum encryption with certified deletion. In: Pass, R., Pietrzak, K. (eds.) Theory of Cryptography. pp. 92–122. Springer International Publishing, Cham (2020)
12. Coiteux-Roy, X., Wolf, S.: Proving erasure. In: IEEE International Symposium on Information Theory, ISIT 2019, Paris, France, July 7-12, 2019. pp. 832–836 (2019). https://doi.org/10.1109/ISIT.2019.8849661, https://doi.org/10.1109/ISIT.2019.8849661
13. Crépeau, C., Kilian, J.: Achieving oblivious transfer using weakened security assumptions (extended abstract). In: 29th FOCS. pp. 42–52. IEEE Computer Society Press (Oct 1988). https://doi.org/10.1109/SFCS.1988.21920
14. Crépeau, C., van de Graaf, J., Tapp, A.: Committed oblivious transfer and private multi-party computation. In: Coppersmith, D. (ed.) CRYPTO'95. LNCS, vol. 963, pp. 110–123. Springer, Heidelberg (Aug 1995). https://doi.org/10.1007/3-540-44750-4_9
15. Damgård, I., Fehr, S., Lunemann, C., Salvail, L., Schaffner, C.: Improving the security of quantum protocols via commit-and-open. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 408–427. Springer, Heidelberg (Aug 2009). https://doi.org/10.1007/978-3-642-03356-8_24
16. Dulek, Y., Grilo, A.B., Jeffery, S., Majenz, C., Schaffner, C.: Secure multi-party quantum computation with a dishonest majority. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 729–758. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45727-3_25
17. European Commission: Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance) (2016), https://eur-lex.europa.eu/eli/reg/2016/679/oj
18. Fu, H., Miller, C.A.: Local randomness: Examples and application. Phys. Rev. A **97**, 032324 (Mar 2018). https://doi.org/10.1103/PhysRevA.97.032324, https://link.aps.org/doi/10.1103/PhysRevA.97.032324
19. Garg, S., Goldwasser, S., Vasudevan, P.N.: Formalizing data deletion in the context of the right to be forgotten. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part II. LNCS, vol. 12106, pp. 373–402. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45724-2_13
20. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing. p. 169–178. STOC '09, Association for Computing Machinery, New York, NY, USA (2009). https://doi.org/10.1145/1536414.1536440, https://doi.org/10.1145/1536414.1536440

21. Gentry, C., Sahai, A., Waters, B.: Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 75–92. Springer, Heidelberg (Aug 2013). https://doi.org/10.1007/978-3-642-40041-4_5
22. Gottesman, D.: Uncloneable encryption. Quantum Inf. Comput. **3**, 581–602 (2003)
23. Grilo, A.B., Lin, H., Song, F., Vaikuntanathan, V.: Oblivious transfer is in miniqcrypt. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12697, pp. 531–561. Springer (2021). https://doi.org/10.1007/978-3-030-77886-6_18, https://doi.org/10.1007/978-3-030-77886-6_18
24. Heisenberg, W.: Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik. Zeitschrift fur Physik **43**(3-4), 172–198 (Mar 1927). https://doi.org/10.1007/BF01397280
25. 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.) Advances in Cryptology – ASIACRYPT 2021. pp. 606–636. Springer International Publishing, Cham (2021)
26. 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
27. Hiroka, T., Morimae, T., Nishimaki, R., Yamakawa, T.: Certified everlasting zero-knowledge proof for QMA. CRYPTO (2022), https://ia.cr/2021/1315
28. Kalai, Y.T., Raz, R.: Probabilistically checkable arguments. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 143–159. Springer, Heidelberg (Aug 2009). https://doi.org/10.1007/978-3-642-03356-8_9
29. Katz, J., Thiruvengadam, A., Zhou, H.S.: Feasibility and infeasibility of adaptively secure fully homomorphic encryption. In: Kurosawa, K., Hanaoka, G. (eds.) PKC 2013. LNCS, vol. 7778, pp. 14–31. Springer, Heidelberg (Feb / Mar 2013). https://doi.org/10.1007/978-3-642-36362-7_2
30. Khurana, D., Mughees, M.H.: On statistical security in two-party computation. In: Pass, R., Pietrzak, K. (eds.) Theory of Cryptography - 18th International Conference, TCC 2020, Durham, NC, USA, November 16-19, 2020, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12551, pp. 532–561. Springer (2020). https://doi.org/10.1007/978-3-030-64378-2_19, https://doi.org/10.1007/978-3-030-64378-2_19
31. Kilian, J.: Founding cryptography on oblivious transfer. In: 20th ACM STOC. pp. 20–31. ACM Press (May 1988). https://doi.org/10.1145/62212.62215
32. Kundu, S., Tan, E.Y.Z.: Composably secure device-independent encryption with certified deletion (2020). https://doi.org/10.48550/ARXIV.2011.12704, https://arxiv.org/abs/2011.12704
33. Lo, H.K.: Insecurity of quantum secure computations. Phys. Rev. A **56**, 1154–1162 (Aug 1997). https://doi.org/10.1103/PhysRevA.56.1154, https://link.aps.org/doi/10.1103/PhysRevA.56.1154
34. Lo, H.K., Chau, H.F.: Is quantum bit commitment really possible? Physical Review Letters **78**(17), 3410 (1997)
35. Mayers, D.: Unconditionally secure quantum bit commitment is impossible. Physical review letters **78**(17), 3414 (1997)

36. Mayers, D., Salvail, L.: Quantum oblivious transfer is secure against all individual measurements. In: Proceedings Workshop on Physics and Computation. PhysComp'94. pp. 69–77. IEEE (1994)
37. Morimae, T., Yamakawa, T.: Quantum commitments and signatures without one-way functions. To appear in CRYPTO (2022), https://ia.cr/2021/1691
38. Naor, M.: Bit commitment using pseudo-randomness. In: Brassard, G. (ed.) CRYPTO'89. LNCS, vol. 435, pp. 128–136. Springer, Heidelberg (Aug 1990). https://doi.org/10.1007/0-387-34805-0_13
39. Poremba, A.: Quantum proofs of deletion for learning with errors. Cryptology ePrint Archive, Report 2022/295 (2022), https://ia.cr/2022/295
40. Unruh, D.: Everlasting multi-party computation. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 380–397. Springer, Heidelberg (Aug 2013). https://doi.org/10.1007/978-3-642-40084-1_22
41. Unruh, D.: Revocable quantum timed-release encryption. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 129–146. Springer, Heidelberg (May 2014). https://doi.org/10.1007/978-3-642-55220-5_8
42. Watrous, J.: Zero-knowledge against quantum attacks. In: Kleinberg, J.M. (ed.) 38th ACM STOC. pp. 296–305. ACM Press (May 2006). https://doi.org/10.1145/1132516.1132560
43. Wiesner, S.: Conjugate coding. SIGACT News **15**, 78–88 (1983)
44. 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, https://doi.org/10.1109/18.796385
45. Yao, A.C.C.: Protocols for secure computations (extended abstract). In: 23rd FOCS. pp. 160–164. IEEE Computer Society Press (Nov 1982). https://doi.org/10.1109/SFCS.1982.38
46. Yao, A.C.C.: Security of quantum protocols against coherent measurements. In: 27th ACM STOC. pp. 67–75. ACM Press (May / Jun 1995). https://doi.org/10.1145/225058.225085