

Proving Erasure

Xavier Coiteux-Roy
Faculty of Informatics
Università della Svizzera italiana
Lugano, Switzerland
xavier.coiteux.roy@usi.ch

Stefan Wolf
Faculty of Informatics
Università della Svizzera italiana
Lugano, Switzerland
stefan.wolf@usi.ch

Abstract—It seems impossible to certify that a remote hosting service does not leak its users’ data — or does quantum mechanics make it possible? We investigate if a server hosting data can information-theoretically prove its definite deletion using a “BB84-like” protocol. To do so, we first rigorously introduce an alternative to privacy by encryption: *privacy delegation*. We then apply this novel concept to provable deletion and remote data storage. For both tasks, we present a protocol, show its partial security, and display its vulnerability to eavesdropping attacks targeting only a few bits.

Index Terms—quantum cryptography, information-theoretic security, privacy amplification, provable deletion, data storage, nonlocality.

I. INTRODUCTION

As the story goes [1], some sailors who did not have access to secure private storage on ships would tie their bag of belongings with the “thief knot.” This particular knot resembled the reef knot, one of the most common knots, so that anyone not knowing the secret would tie back the bag the wrong way. The canny sailors would then detect when someone went through their belongings because the thief knot they tied would be altered.

This folktale inspires us to introduce in Section II a novel, relaxed, notion of privacy: *privacy delegation*. Privacy delegation does not prevent eavesdropping but it makes such an act inevitably detectable. It is useful in applications where we cannot aspire to have perfect encryption in the sense of Shannon [2].

A typical application is data storage (Section III-A): Information-theoretically secure encryption of data is impossible unless one keeps a secret key at least as long as the message that is remotely stored [2], defeating the purpose of storing it in the first place. So how can a hosting server meaningfully certify it protected one’s privacy?

This is related to the question of provable erasure (Section III-B): Can one verify some information was indeed deleted? These two tasks seem at first sight impossible, at least when restricting ourselves to classical physics. We use quantum theory instead and follow the path of Bennett and Brassard in 1984 [3]. We present a straightforward protocol in Section IV and sketch a proof of its partial security in Section IV-C, before showing in Section IV-D it fails to be

unconditionally secure against some limited attacks. We are, in Section IV-E, left with the question of how to fix it.

II. DELEGATING PRIVACY

A. Privacy Delegation: an Alternative Standard for Privacy

The general framework of privacy delegation is the following. After a certain protocol (*e.g.*, data storage), the prover/server presents a proof to the verifier/user. If the proof is accepted by the verifier, they are sure the prover respected their privacy and that none of their data was leaked. On the other hand, the rejection of the proof by the verifier alarms them that their data might have been compromised. Fig. 1 summarizes the idea of delegating privacy.

We will define privacy delegation in the language of modern cryptography [4] through the notion of conditional indistinguishability: We will say the protocol is secure if, once the privacy certificate has been produced by the prover \mathcal{P} and accepted by the verifier \mathcal{V} , no eavesdropping by an adversary \mathcal{A} can offer any advantage in discriminating whether the now-completed privacy-delegation protocol was run on one message or another, given they are the same length.

Definition 1. A *privacy-delegation protocol* is information-theoretically secure if and only if for any prover \mathcal{P} and adversary \mathcal{A} :

$$\mathbb{P}[\text{CERT}^{\mathcal{P}}(n)] \cdot (\mathbb{P}[\text{DISCR}^{\mathcal{A}} | \text{CERT}^{\mathcal{P}}(n)] - 1/2) < \text{negl}(n),$$

(security of privacy-delegation)

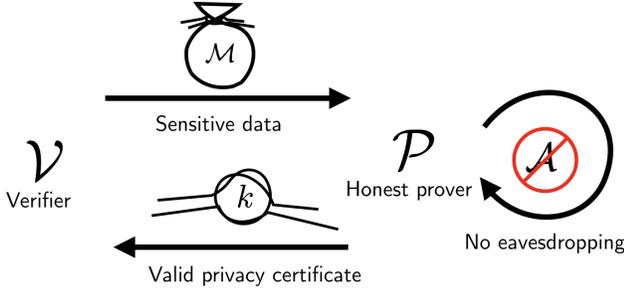
where n is the security parameter and $\text{negl}(n)$ is a negligible function, meaning smaller than the inverse of any polynomial of n for sufficiently large n .

CERT is the certification experiment and is passed (*i.e.*, it evaluates to true) if and only if the verifier \mathcal{V} accepts the *privacy certificate* produced by the prover \mathcal{P} .

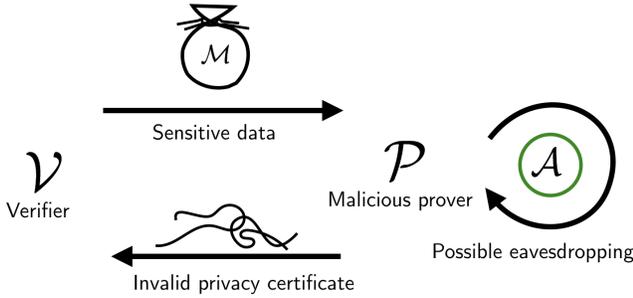
DISCR is the discrimination experiment, and it is passed if and only if, when given at the beginning of the privacy-delegation protocol, either the legitimate message or a dummy one (with equal probability), the adversary \mathcal{A} guesses successfully which one it is. While not necessary, it is convenient to pose \mathcal{P} and \mathcal{A} as the same entity.

Finally, the compact formulation of the privacy-delegation security definition reflects that the privacy-delegation protocol is secure as soon as either term is negligible.

Note how, in presence of a prover generating a valid privacy certificate with probability 1, the above definition of secure privacy delegation reduces to Shannon’s notion of information-theoretic security against eavesdropping. Any scheme which is perfectly secured in Shannon’s sense can be seen as a trivial privacy-delegation scheme for which all certificates are valid.



(a) A valid privacy certificate guarantees that no leaks about the sensitive message occurred.



(b) It is arbitrarily unlikely to produce a valid privacy certificate if eavesdropping has occurred.

Fig. 1: Privacy delegation is a novel approach to privacy.

We conclude our definition of privacy delegation by requiring one additional property: It is desirable that a privacy-delegation protocol fails with non-negligible probability only when the prover is dishonest.

Definition 2. A privacy-delegation protocol is correct if and only if there exists $\mathcal{P}_{\text{honest}}$ such as:

$$P[\text{CERT}^{\mathcal{P}_{\text{honest}}}(n)] > 1 - \text{negl}(n) . \quad (\text{correctness})$$

B. Privacy Delegation Is an Alarm Bell

We shall stress the role of privacy delegation to detect information leaks; it is not by itself an encryption method: It offers no security against a malicious prover other than announcing its behavior. In this sense, privacy delegation is more an alarm bell than a lock.

It can be useful in bringing a concept of liability to a hosting server. When a server leaks its users’ data, whether by malice or negligence, its privacy certificate will be rejected, and the server can subsequently be boycotted or taken to court.

Note that privacy delegation is different from digital watermarking [5], even if both techniques can be used to detect leaking agents. Digital watermarking mixes the sensitive data with hidden identification information in a way the source of the leak can be identified once the leaked data surfaces, while privacy delegation enables to detect misbehavior even if no abnormal traces of the sensitive information have been observed. Privacy delegation offers in that sense stronger protection, even if neither approach directly prevents the leakage of data.

III. APPLICATIONS

A. Remote storage

An important application of privacy delegation is the remote storage of classical information. In this setting, a user wants to upload classical data to a server and be able to retrieve it later, while the server wants to prove to the user it protected its data from any eavesdropping — *both in the past and the future*. We explicit the general form of this task.

The task of remote storage with privacy delegation:

- 1) A user (verifier \mathcal{V}) wants to store the classical message \mathcal{M} on a server (prover \mathcal{P}). The user can manipulate quantum states.
- 2) \mathcal{V} generates a key k at random and sends quantum encoding $\rho(\mathcal{M}, k)$ to \mathcal{P} . The state $\rho(\mathcal{M}, k)$ depends on the specific protocol.
- 3) Time passes. Eventually the user \mathcal{V} asks the server \mathcal{P} for its data $\rho(\mathcal{M}, k)$ back and wants to be assured no copies were made.
- 4) \mathcal{P} sends back a quantum state ρ' which will also act as the privacy certificate. If they are honest, $\rho' = \rho(\mathcal{M}, k)$.
- 5) \mathcal{V} examines the privacy certificate (*i.e.*, ρ'). If they accept it and the protocol is secure, neither \mathcal{P} nor any eavesdropper \mathcal{A} have any information about \mathcal{M} .

Note that the last step is not a full authentication of the quantum state $\rho(\mathcal{M}, k)$ but only of its privacy-delegation layer. It is, however, easy to guarantee the integrity of the classical data by adding on top of the privacy-delegation protocol a Wegman-Carter classical authentication scheme [6]. These schemes are well-known and can be information-theoretically secure even with a short key.

Because of the nature of remote storage with privacy, it is clear that if the user must keep a secret key, it must at least be shorter than the message to be stored. This is impossible if we aim for arbitrarily perfect secrecy, as stated by Shannon’s theorem¹, and it is why we look for privacy delegation instead of encryption.

Privacy delegation differs from, but is not incompatible with (as we develop next), the standard approach which is to aim for computational security. Computational security is reached through classical encryption relying on assumptions on the

¹Is it really impossible to circumvent Shannon’s theorem using quantum mechanics? After all, quantum key extension and quantum key recycling are already possible [7].

computing power of the adversary and the hardness of certain mathematical problems [8], [9].

B. Provable Deletion

Provable deletion can be framed as a general case of remote storage. In this scenario, the user/verifier is not interested in retrieving their remotely stored data, but simply orders for its definitive erasure. The server/prover subsequently produces a token which, if accepted by the user, certifies the erasure. It is of special interest because classically, there is no way for the server to prove it did not secretly keep a backup of the data as the act of copying classical information is undetectable.

The task of provable deletion with privacy delegation:

- 1) A user (verifier \mathcal{V}) wants to store the classical message \mathcal{M} on a server (prover \mathcal{P}). The user can manipulate quantum states.
- 2) \mathcal{V} generates a key k at random and sends quantum encoding $\rho(\mathcal{M}, k)$ to \mathcal{P} . The state $\rho(\mathcal{M}, k)$ depends on the specific protocol.
- 3) Time passes. Eventually the user \mathcal{V} asks the server \mathcal{P} to provably delete all of its data \mathcal{M} .
- 4) \mathcal{P} applies some operations on $\rho(\mathcal{M}, k)$ and provides to \mathcal{V} a privacy certificate \mathcal{W} .
- 5) \mathcal{V} examines \mathcal{W} , whose acceptance proves the erasure of \mathcal{M} .

We note here that we are analyzing the provable deletion of purely classical information. The no-go theorem for deleting *arbitrary* quantum states [10] does, therefore, not apply.

C. Combining Privacy Delegation and Computational Security

In cases where standard privacy is desirable, meaning when privacy concerns require more than leak detection, privacy delegation can be complemented by classical encryption with computational security. Such a combination has multiple advantages. First, the storage is still meaningful as both techniques only necessitate short keys. Second, it makes possible the concept of recalling encrypted data. Standard classical encryption relies on assumptions that need not hold forever: Computing power grows exponentially, and the underlying hard mathematical problems could at any time be solved more easily than currently believed. There is no guarantee of everlasting security with standard classical encryption because any encrypted message can be stored by an adversary until its encryption becomes obsolete [11]. As such, there is always a risk, even if one uses strong up-to-date encryption, in uploading data to a server that could leak it. If, however, the server is able to produce a valid privacy certificate when the user wants to recall data to change its encryption, then the user can be sure that her or his data is still perfectly safe.

Finally, an adversary scanning massive amounts of data looking for weak encryptions, obsolete standards and/or valuable information can be quickly detected by the privacy-delegation scheme. This makes the hosting network more

secure as a whole, since an adversary will not necessarily know what type of computational encryption they are attacking before they attack it in a detectable way.

IV. IMPLEMENTATION: A NAIVE PROTOCOL

We explore how to implement privacy delegation in the information-theoretically secure way of Section II. We start with a naive protocol which splits into two versions: remote storage (STORAGE) and provable deletion (ERASURE). We then show how they are partially secure whereas an eavesdropper can still gain *some* small amount of information even when privacy should be certified.

A. The Encoding

Both versions start the same way. The idea is for the user to preemptively, at random positions, sprinkle the m bits plain message in the rectilinear basis $\{|0\rangle, |1\rangle\}$ with n random “trap bits” in the diagonal basis $\{|+\rangle, |-\rangle\}$. Fig. 2 illustrates the idea.

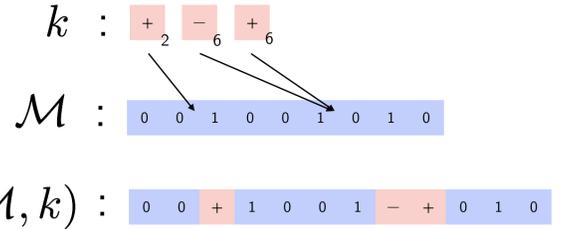


Fig. 2: The message is in the rectilinear basis and is interceded with bits in the diagonal basis whose ulterior integrity certifies the then-recovered privacy of the message.

The secret key k , used later by the user to validate the privacy certificate provided by the server, then consists of the n random check bits s_0, s_1, \dots, s_t and their positions. The length of the secret key is, therefore (with the approximation valid when $m \gg n$):

$$\text{len}(k) = n + \log \binom{m+n}{n} \approx n \log m \quad (1)$$

Note that the encoded message is not, and needs not be, securely encrypted in the traditional sense: An adversary can read almost all of it without requiring the secret key. However, when it does so, it destroys the information needed to conclude successfully the privacy delegation: The eavesdropping can be detected². The security parameter n constrains the minimal length m of the message meaningfully stored: $m \gg n$.

B. The Privacy Delegation

The protocol now splits into two versions according to the chosen task.

²This is different from quantum sealing [12], because we require here that the secret key be shorter than the message and do not insist on the message being totally readable by someone not having the key.

a) *Remote-Storage Version*: In the version STORAGE, the server sends back the whole quantum state to the user, who then checks the integrity of the sprinkled bits and separate them from the meaningful data. If the user's measurements on the sprinkled bits match their secret key, they accept the privacy certificate. Otherwise, they reject it and accuse the server of having leaked their information.

Note that the *integrity* of the information stored on the server is not guaranteed by that basic protocol even when the privacy delegation succeeds because an adversary could for example flip in the rectilinear basis (*i.e.*, apply an X-gate) every bit without making CERT fail. As mentioned before we are, however, not concerned by this kind of attack because it can be prevented in a straightforward way using a Wegman-Carter classical authentication scheme.

b) *Provable-Deletion Version*: In the version ERASURE, the user asks the server to delete all of its data by measuring it in the diagonal basis and publicly announcing the result. If the server behaved honestly and did not leak any data, the output should be completely random except for the sprinkled check bits. The user accepts the privacy certificate if these values correspond to the secret key, and reject it otherwise. They ignore all non-sprinkled bits.

The proof of erasure comes from the server being forced to measure in the diagonal basis bits that are encoded in the rectilinear basis. This gives a series of uniformly random bits while destroying the original information. Since the diagonal-basis measurements are announced publicly by the prover, they cannot rewind the protocol later: The information which was encoded in the rectilinear basis is lost forever.

C. Partial Security

Both versions of the naive protocol offer arbitrarily high partial privacy-delegation security for sufficiently long messages, meaning that a valid certificate information-theoretically guarantees that a very limited amount of information was leaked. We aim to give the intuition behind this statement by proving it for non-coherent attacks. We show the privacy certificate will be rejected with arbitrarily high probability in presence of enough eavesdropping.

Theorem 1. *For both STORAGE and ERASURE, the probability of any prover \mathcal{P} producing a valid privacy certificate in presence of an eavesdropper $\bar{A}(r)$ doing a rectilinear projective measurement on r qubits is given by:*

$$\mathbb{P}[\text{CERT}^{\mathcal{P}|\bar{A}(r)}] \leq 2^{-rn/(m+n)+\epsilon} + \exp(-2\epsilon^2 r), \quad (2)$$

where m is the message length, n the number of sprinkled check bits, and ϵ a security parameter.

Proof. The probability to pass the certification given the number $K = k$ of randomly sprinkled check bits that were measured by the eavesdropper is 2^{-k} . K follows a hypergeometric distribution (it is as if the eavesdropper had done classical

sampling). The exact probability of passing the certification is thus:

$$\mathbb{P}[\text{CERT}^{\mathcal{P}|\bar{A}(r)}] = \sum_{k=0}^r 2^{-k} \frac{\binom{n}{k} \binom{m}{r-k}}{\binom{m+n}{r}} \quad (3)$$

The upper bound follows from Hoeffding's inequality [13], [14].

$$\mathbb{P}[|K - \mu(K)| \geq \epsilon] \leq 2 \exp(-2\epsilon^2 r) \quad (4)$$

with $\mu(K) = rn/(m+n)$ since K is hypergeometric. \square

D. An Attack Leaking Partial Information

An attack on a limited number of bits is, however, still possible. More precisely, an adversary can pass with non-negligible probability both a discrimination experiment (*e.g.*, discriminating a message starting by 0 from an uniformly random one) and the certification experiment.

Theorem 2. *STORAGE and ERASURE are insecure against an eavesdropper \bar{A} measuring only, in the rectilinear basis, the first bit of the quantumly encoded message and a prover $\bar{\mathcal{P}}$ proceeding honestly otherwise.*

Proof.

$$\begin{aligned} & \mathbb{P}[\text{DISCR}^{\bar{A}}(n) | \text{CERT}^{\bar{\mathcal{P}}}(n)] - 1/2 \\ &= (1 - n/(n+m))/4 \in \Theta(1/n) \\ & \mathbb{P}[\text{CERT}^{\bar{\mathcal{P}}}(n)] = 1 - n/(2(n+m)) \in \Theta(1/n) \\ \therefore & \mathbb{P}[\text{CERT}^{\bar{\mathcal{P}}}(n)] \\ & \cdot \left(\mathbb{P}[\text{DISCR}^{\bar{A}} | \text{CERT}^{\bar{\mathcal{P}}}(n)] - 1/2 \right) \not\leq \text{negl}(n) . \quad \square \end{aligned}$$

This does not mean that the protocol is useless, as it is already desirable to restrict the amount of stored data that can be leaked without the user becoming aware. This is especially so if the data are already encrypted with computational security. However, for our protocol to be more than only partially secure, the protocol applied on any message should be indistinguishable from the one applied on any other one. It is important to note the weakness here is also present in naive implementations of BB84. We will discuss next how this problem is, however, usually resolved in the case of BB84, and what this implies for our privacy-delegation protocol.

E. Privacy Amplification by Public Discussion

In its original version [3], the BB84 quantum key-distribution protocol was also susceptible to the weak attack suggested in Proposition 2. There was, however, a fix: privacy amplification by public discussion [15]–[17]. These now well-known schemes aim to reduce the amount of information an eavesdropper can have about the private key. Two parties do so by assessing through public discussion how much information the eavesdropper can have about their key and by agreeing accordingly on a randomly selected error-correcting code or a hashing function from a universal class. When successful (if Eve does not have too much information), the result is a shorter but fully private key. Public discussion between Alice

and Bob is achieved through a perfectly authenticated, but not private, channel.

In our storage scenario, public discussion is unavailable as it would necessitate communication between a user's past and future selves. Communication from the past to the future could still be simulated by the user if they keep some secret information, but the amount of this information they would need to privately store to apply standard privacy-amplification schemes would be longer than the message they want to store. Privacy amplification by public discussion as it is usually done fails, therefore, to extend partial privacy-delegation security to full information-theoretic privacy-delegation security in the storage setting. Could there be an alternative way to go from partial to total security in the privacy-delegation storage setting? This is our main open question.

V. CONCLUSION

We approached this work with one question in mind: Does quantum mechanics make provable deletion of classical data achievable? This leads us to define rigorously *privacy delegation*, an alternative to the ideal of privacy through encryption, and to formalize how to detect leaks during data storage.

We suggest that quantum-provable deletion could be possible by providing a naive privacy-delegation scheme for both the remote-storage and provable-deletion problems. The question remains, however, open as our scheme offers only partial security: It fails against restricted attacks on targeted bits. This shortcoming is known from quantum key distribution; unfortunately, the fix used there, namely privacy amplification by public discussion, cannot be applied in our setting.

To conclude, we emphasize that even if the protocols presented are out of reach of current quantum technologies — quantum memory is hard —, there is real value in devising and analyzing this kind of theoretical puzzles: Formal definitions give focus and direction to future research, while attempts to solve difficult problems under novel constraints invariably spark new ideas that ultimately turn into the building blocks needed to shrink the domain of the mathematical, cryptological and physical *impossibles*.

ACKNOWLEDGMENT

We would like to thank Alberto Montana, Arne Hansen, and Cecilia Boschini for helpful discussions.

REFERENCES

- [1] C. Ashley, *The Ashley Book of Knots*. Doubleday Books, 1993.
- [2] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28, no. 4, pp. 656–715, 1949.
- [3] C. H. Bennett and G. Brassard, "Quantum cryptography: public key distribution and coin tossing int.," in *Conf. on Computers, Systems and Signal Processing (Bangalore, India, Dec. 1984)*, 1984, pp. 175–9.
- [4] Y. Lindell and J. Katz, *Introduction to modern cryptography*. Chapman and Hall/CRC, 2014.
- [5] H. Honsinger, "Digital watermarking," *Journal of Electronic Imaging*, vol. 11, 2002. [Online]. Available: <https://doi.org/10.1117/1.1494075>
- [6] M. N. Wegman and J. L. Carter, "New hash functions and their use in authentication and set equality," *Journal of Computer and System Sciences*, vol. 22, no. 3, pp. 265–279, 1981.
- [7] I. Damgård, T. B. Pedersen, and L. Salvail, "A quantum cipher with near optimal key-recycling," in *Annual International Cryptology Conference*. Springer, 2005, pp. 494–510.
- [8] N. F. Pub, "197: Advanced encryption standard (AES)," *Federal Information Processing Standards Publication*, vol. 197, no. 441, p. 0311, 2001.
- [9] E. Barker, "Sp 800-67 rev. 2, recommendation for triple data encryption algorithm (tdea) block cipher," *NIST special publication*, vol. 800, p. 67, 2017.
- [10] A. K. Pati and S. L. Braunstein, "Impossibility of deleting an unknown quantum state," *Nature*, vol. 404, no. 6774, p. 164, 2000.
- [11] Y. Aumann, Y. Z. Ding, and M. O. Rabin, "Everlasting security in the bounded storage model," *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1668–1680, June 2002.
- [12] H. Bechmann-Pasquinucci, "Quantum seals," *International Journal of Quantum Information*, vol. 1, no. 02, pp. 217–224, 2003.
- [13] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American statistical association*, vol. 58, no. 301, pp. 13–30, 1963.
- [14] N. J. Bouman and S. Fehr, "Sampling in a quantum population, and applications," in *Annual Cryptology Conference*. Springer, 2010, pp. 724–741.
- [15] U. M. Maurer, "Secret key agreement by public discussion from common information," *IEEE transactions on information theory*, vol. 39, no. 3, pp. 733–742, 1993.
- [16] C. H. Bennett, G. Brassard, and J.-M. Robert, "Privacy amplification by public discussion," *SIAM Journal on Computing*, vol. 17, no. 2, pp. 210–229, 1988.
- [17] C. H. Bennett, G. Brassard, C. Crépeau, and U. M. Maurer, "Generalized privacy amplification," *IEEE Transactions on Information Theory*, vol. 41, no. 6, pp. 1915–1923, 1995.