IACR Communications in Cryptology
ISSN 3006-5496, Vol. 1, No. 4, 21 pages.

https://doi.org/10.62056/ah2i5w7sf

An efficient combination of quantum error correction and authentication

Yfke Dulek[2,3], Garazi Muguruza[1,3] and Florian Speelman[1,3]

[1] University of Amsterdam, Informatics Institute, Netherlands
[2] CWI, Netherlands
[3] QuSoft, Netherlands

Abstract. When sending quantum information over a channel, we want to ensure that the message remains intact. Quantum error correction and quantum authentication both aim to protect (quantum) information, but approach this task from two very different directions: error-correcting codes protect against probabilistic channel noise and are meant to be very robust against small errors, while authentication codes prevent adversarial attacks and are designed to be very sensitive against any error, including small ones.

In practice, when sending an authenticated state over a noisy channel, one would have to wrap it in an error-correcting code to counterbalance the sensitivity of the underlying authentication scheme. We study the question of whether this can be done more efficiently by combining the two functionalities in a single code. To illustrate the potential of such a combination, we design the threshold code, a modification of the trap authentication code which preserves that code's authentication properties, but which is naturally robust against depolarizing channel noise. We show that the threshold code needs polylogarithmically fewer qubits to achieve the same level of security and robustness, compared to the naive composition of the trap code with any concatenated CSS code. We believe our analysis opens the door to combining more general error-correction and authentication codes, which could improve the practicality of the resulting scheme.

Keywords: quantum cryptography · quantum authentication · quantum error-correction · trap code

1 Introduction

Authentication is one of the most fundamental tasks of modern cryptography – for many applications it is imperative that the integrity of data is preserved, not just against noise and random errors, but even against adversarial attacks. Constructions for message authentication codes (MACs) underlay many important cryptographic protocols that are in constant use for secure internet communication. We study the notion of *quantum authentication*, where instead of wanting to ascertain the integrity of classical data, the data involved consists of qubits.

Starting with the work of Barnum, Crepeau, Gottesman, Smith, and Tapp [BCG+02], several quantum authentication codes have been proposed. In our current work, we will mostly be working with two prominent examples, namely the *Clifford code* and the *trap code*, not going into depth for other examples such as the polynomial code [BCG+06] or the Auth-QFT-Auth scheme [GYZ17]. The Clifford code [ABOE08] constructs a very effective authentication scheme, which involves attaching a number of flag qubits to the

E-mail: g.muguruza@uva.nl (Garazi Muguruza), f.speelman@uva.nl (Florian Speelman)

Received: 2024-10-09 Accepted: 2024-12-03

plaintext, and then scrambling the state using a random Clifford – this turns out to be a very efficient way of guaranteeing security, and it can also be used as a building block for interactive proofs [ABOE08] and multi-party computation [DNS10, DNS12, DGJ+20]. The trap code [BGS13, BW16] constructs a scheme, for which encoding consists of interspersing the plaintext (in an error-correcting code) with so-called traps which try to detect an adversary's attempted modifications. A very interesting property of this authentication scheme is its natural interaction with computation; it is possible to perform some quantum gates 'transversally' on qubits of the ciphertext, which results in a valid authentication of a new ciphertext (with an updated key). This property enabled the trap code to be a crucial ingredient in various results within quantum cryptography, such as the construction of quantum one-time programs [BGS13], a scheme for quantum zero-knowledge proofs for QMA [BJSW16], and verifiable homomorphic encryption [ADSS17]. Also see an extended version of the trap code which supports key recycling and ciphertext authentication [DS18] for more context of this code.

Multiple works have followed the first notions of security for the primitive of quantum authentication of Barnum et al. [BCG+02], which did not consider adversaries entangled with the encrypted message. An important requirement for authentication protocols is a composable security notion, which ensures that the scheme is secure when using it in any arbitrary environment. By using a simulator-based approach to security, several additional desirable properties to the basic functionality have been proven, such as key recycling [HLM16, Por17, GYZ17] or quantum ciphertext authentication [AGM18, DS18]. Additionally, it is possible to study the notion of authentication in the setting of computational security [BMPZ19], including public-key versions of the primitive [AGM21]. In this work we extensively use the Abstract Cryptography (AC) framework introduced by Maurer and Renner [MR11], which views cryptography as a resource theory and has been previously applied successfully to prove security of purity testing based authentication schemes by Portmann [Por17].

Authentication is usually applied to messages that will be transmitted at some point, and such a transmission involves incurring some *error* by the quantum channel which is used. The MACs present in the literature will inevitably reject whenever any error is present in the channel. However, it is possible to first encode this message in a quantum authentication code, and then wrap the result in an *error-correction code* (see also e.g. the discussion by [HLM16] and mainly [Por17]).

Observe that the primitives of quantum authentication and error correction have a conceptual overlap, in the sense that both aim to protect data against modifications. However, in practice there is a large difference in how they are built: authentication codes need to protect against any adversarial attack, and therefore often are extremely sensitive against even minor attempted modifications. For example, if a Pauli operation would be applied to a single qubit that is part of a Clifford-code authenticated state, the encoded plaintext would be almost completely scrambled by having a random n-qubit Pauli operator applied to the entire plaintext. On the other hand, an error-correcting code should be robust against typical (usually low-weight) modifications of the encoded data. Given that the goals of these codes are similar, one might wonder whether this is doing 'double work' in some sense, making the resulting encoded state larger than necessary. We stress that the comparison is only interesting when the outer error-correcting code is indeed doing 'double encoding', as is the case for any code that encodes a single qubit of data like concatenated codes, but does not holds for 'good' error-correcting codes with linear rate and distance.

In this work, we give evidence that this is indeed the case: We construct a code which functions both as a quantum error-correcting code and as a quantum authentication code, and which is more efficient than the naive concatenation of these functionalities would imply. In particular:

- As an example of a combined code, we present the *threshold code*. Even though this is not the main goal of the current work, note that this code preserves several of the useful computational properties of the original trap code, if a CSS code is used as the underlying error-correcting code, having essentially the same encoding procedure as the trap code and only differing in the decoding.
- We show that our scheme is correct and secure, by proving that the resulting code is a good purity testing code. Because of the generality of the AC framework, the same security proof will also imply security under most other security definitions (if these do not require extra properties such as key recycling).
- We compare the resulting scheme to the concatenation of the two primitives separately. If we define efficiency in terms of amount of qubits needed to obtain certain correctness and security for a constant-error quantum depolarizing channel, we show how the resulting scheme is more efficient than applying the codes separately.

2 Preliminaries

2.1 Notation

The single-qubit Pauli matrices given by

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \text{ and } Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = iXZ, \tag{1}$$

form a basis for single-qubit Pauli operations. Note that they are unitaries and any two Pauli operations either commute or anti-commute. An n-qubit Pauli matrix is given by n-fold tensor products of single-qubit Paulis, and we denote the *Pauli group* of n-qubit Pauli matrices by

$$\mathcal{G}_n := \{i^k P_1 \otimes P_2 \otimes \cdots \otimes P_n \ : \text{ where } P_j \in \{I, X, Z, Y\},\ k \in [4]\}. \tag{2}$$

The *weight* of an n-qubit Pauli operation, denoted $\omega(P)$, is the number of non-identity Paulis in the n-fold tensor product. Moreover, we denote by $\omega_X(P)$ and $\omega_Z(P)$ the number of X and Z-Paulis respectively.

The *Clifford group*, $\mathcal{C}_n$, is the group of n-qubit unitaries that leave the Pauli group invariant. That is, given $P \in \mathcal{G}_n$, for all $C \in \mathcal{C}_n$ we have $i^k CPC^\dagger \in \mathcal{G}_n$, where $k \in [4]$.

The logarithm log is considered in base 2, unless specified otherwise.

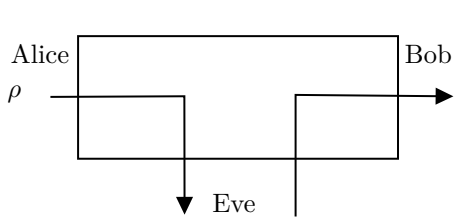
The following variant of Chernoff's bound studies the probability of the majority in a population becoming the minority, and vice versa.

Lemma 1 ([GH01, Lemma 1])**.** *Consider a population set A and a sub-population $B \subset A$. Suppose we pick an integer k such that $0 < k < |A|$ and a random subset $S \subset A$ of size k. Then for any $0 < \gamma \leq 1$ we can bound the relative size of the sub-population in the sample S by*

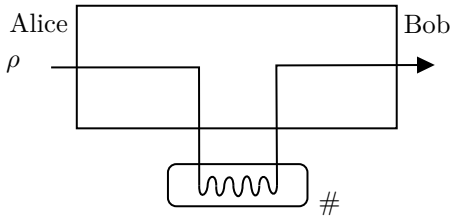
$$P\left[\frac{|S \cap B|}{k} < (1-\gamma)\frac{|B|}{|A|}\right] < \exp\left(-\gamma^2 \frac{|B|}{|A|}\frac{k}{2}\right). \tag{3}$$

2.2 Abstract cryptography

AC views cryptography as a resource theory: a protocols constructs an *ideal resource* from a *real system* by means of a *simulator*. We will describe the basic concepts here, but an in-depth explanation relevant for this work can be found in section 2 and appendix A of Portmann's work [Por17].



(a) Insecure quantum channel with an adversary.



(b) Insecure quantum channel with an adversary replaced by a noisy filter.

Figure 1: Insecure channels with and without filters.

In an n-player setting, a *resource* is an object with n interfaces; allows the players to input and receive messages. We will denote resources by squares, and inputs/outputs from the interfaces by lines intersecting with the squares, see figure 1a. If two resources $\mathcal{C}$ and $\mathcal{K}$ are available to the players, we write $\mathcal{C}||\mathcal{K}$ for the parallel composition of the resources: the resources are simultaneously accessible to the players in any arbitrary order, thus in particular, the order of composition is irrelevant and $\mathcal{C}||\mathcal{K} = \mathcal{K}||\mathcal{C}$.

A *converter* models the local operations that the players can perform in their interfaces. We will denote converters by squares with rounded corners. If a converter σ is connected to the interface i of the resource $\mathcal{C}$, we write $\sigma_i\mathcal{C}$ (or equivalently $\mathcal{C}\sigma_i$), see figure 1. A *protocol* is defined by a set of converters: one for each honest player. On the one hand, an adversary is allowed to perform any operation allowed by quantum mechanics, thus it is essential to prove security against adversaries. On the other hand, for security guarantees it is not enough to show good performance in presence of adversaries, we also need to emulate the presence of no adversary. We do this with a special type of converter, called *filters*, which emulate an honest behavior, for example without any tampering but still with noise. We call *filtered resources* a pair of resource $\mathcal{C}$ and filter $\Diamond_E$, denoted $\mathcal{C}_\Diamond = (\mathcal{C}, \Diamond_E)$.

Definition 1 (Cryptographic security)**.** We say that the protocol $\pi_{AB} = (\pi_A, \pi_B)$ constructs the filtered resource $\mathcal{S}_\Diamond$ from $\mathcal{C}_\#$ within (ε, δ), denoted $\mathcal{C}_\# \xrightarrow{\pi_{AB},(\varepsilon,\delta)} \mathcal{S}_\Diamond$, if the following two conditions hold:

1. In presence of no malicious player, the filtered resources are ε-close to each other
$$d(\pi_{AB}\mathcal{C}_\#, S_\Diamond) \leq \varepsilon.$$
2. In the presence of an adversary, there exists a simulator σ_E, δ-close to the real protocol
$$d(\pi_{AB}\mathcal{C}, \sigma_E\mathcal{S}) \leq \delta.$$

Here the distance d is the supremum over the set of all possible distinguishers allowed by quantum mechanics. If the filtered resources $\mathcal{S}_\Diamond$ and $\mathcal{C}_\#$ are clear from the context, we say that π_{AB} is (ε, δ)-secure, or ε-correct and δ-secure.

We differ from the original definition of cryptographic security in [MR11], where security is defined as the maximum of the two values ε and δ, because these parameters have independent meanings that are interesting to study separately. The ε in item 1 refers to the *correctness* of the protocol. That is, the probability that the protocol running on a noisy channel without adversary will be distinguishable from an ideal channel. The δ in item 2 is the usual *security* in presence of an adversary. Although we might want to consider equal correctness and security in certain scenarios, splitting these two parameters allows us to revisit the proofs from Portmann and understand composability of authentication and

error-correction in terms of cryptographic security parameters. For example, authentication protocols considered in the literature are not correct in presence of noise, i.e. they will always reject with high probability, unless they are wrapped in error-correcting codes, which are correct but not necessarily secure.

Theorem 1 (Serial composition security)**.** *Let the protocols π and π' construct $\mathcal{S}_\Diamond$ from $\mathcal{R}_\#$ and $\mathcal{T}_\square$ from $\mathcal{S}_\Diamond$ within (ε, δ) and (ε', δ') respectively, i.e.*

$$\mathcal{R}_\# \xrightarrow{\pi,(\varepsilon,\delta)} \mathcal{S}_\Diamond \quad \text{and} \quad \mathcal{S}_\Diamond \xrightarrow{\pi',(\varepsilon',\delta')} \mathcal{T}_\square. \tag{4}$$

Then the serial composition $\pi'\pi$ constructs $\mathcal{T}_\square$ from $\mathcal{R}_\#$ within $(\varepsilon + \varepsilon', \delta + \delta')$,

$$\mathcal{R}_\# \xrightarrow{\pi'\pi,(\varepsilon+\varepsilon',\delta+\delta')} \mathcal{T}_\square. \tag{5}$$

Proof. The statement follows directly from the triangle inequality. For ε-correctness we have that

$$d(\pi'\pi\mathcal{R}_\#, \mathcal{T}_\square) \leq d(\pi'\pi\mathcal{R}_\#, \pi'\mathcal{S}_\Diamond) + d(\pi'\mathcal{S}_\Diamond, \mathcal{T}_\square) \leq d(\pi\mathcal{R}_\#, \mathcal{S}_\Diamond) + \varepsilon' \leq \varepsilon + \varepsilon'. \tag{6}$$

Similarly for δ-security, the composed converter $\sigma'\sigma$ is a converter for the composition since

$$d(\pi'\pi\mathcal{R}, \sigma'\sigma\mathcal{T}) \leq d(\pi'\pi\mathcal{R}, \pi'\sigma\mathcal{S}) + d(\pi'\sigma\mathcal{S}, \sigma'\sigma\mathcal{T}) \leq d(\pi\mathcal{R}, \sigma\mathcal{S}) + d(\pi'\mathcal{S}, \sigma'\mathcal{T}) \leq \delta + \delta', \tag{7}$$

where we used commutativity of converters $\alpha\beta\mathcal{C} = \beta\alpha\mathcal{C}$ and the pseudo-metric property $d(\alpha\mathcal{C}, \alpha\mathcal{C}') \leq d(\mathcal{C}, \mathcal{C}')$, see [Mau12]. □

2.3 Quantum error correction

Since quantum information is very sensitive to errors and noise from the environment; quantum error correction is developed as a tool to protect data against errors. A $[[n, k, d]]$ quantum error-correcting code (QECC) is an encoding of k 'logical qubits' (which we wish to protect from errors) into a codeword consisting of n 'physical qubits' (auxiliary qubits), with $n > k$. The distance d is the minimum weight of a Pauli P to convert one valid codeword into another.

After the encoded information is subjected to noise, we perform a collective measurement on the n qubits which will enable us to diagnose the type of error that occurred, called error syndrome. Afterwards, error decoding or recovery is performed, to return to the original state of the code. We say that a $[[n, k, d]]$ QECC can correct t errors if recovery is successful for any superoperator with support on the set of Pauli operators of weight up to t. In any case, we assume that we can always decode, possibly to a different state than the input if more than t errors are present. Moreover, sometimes we are satisfied just with knowing if an error has occurred, without the need to reverse it. We call this the error-detection property of the code. In fact, a QECC with distance d can correct $t = (d-1)/2$ errors. For a more in-depth analysis we refer the reader to standard literature in error correction [NC10, Pre99].

Stabilizer codes introduced by Gottesman [Got96] allow us to describe quantum states in terms of the operators stabilizing them instead of working with the state itself, by means of group theory techniques for the Pauli group. Any two elements of the Pauli group $\mathcal{G}_n$ either commute or anti-commute and square to $\pm I$, which we will use to describe codewords. Given an abelian subgroup S of $\mathcal{G}_n$, we define the *stabilizer code* V_S to be the stable states under the action of elements of S. That is,

$$V_S := \{|\psi\rangle : M|\psi\rangle = |\psi\rangle, \quad M \in S\}. \tag{8}$$

Let us denote by $S_1, \ldots, S_m$ the generators of the stabilizer group $S = \langle S_1, \ldots, S_m \rangle$. Since any Pauli error $P \in \mathcal{G}_n$ either commutes or anti-commutes with each element of the generator, we can define the vector $s_P = (s_{1,P}, \ldots, s_{m,P})$ such that $s_{j,P} = 0$ if P_j commutes with S_j, and $s_{j,P} = 1$ if it anti-commutes. Therefore,

$$S_j P |\psi\rangle = (-1)^{s_{j,P}} P S_i |\psi\rangle = (-1)^{s_{j,P}} P |\psi\rangle, \quad \text{for all } |\psi\rangle \in V_S. \tag{9}$$

We call the vector s the *syndrome* of the error-correcting code. Errors with non-zero syndrome for some element in the stabilizer $M \in S$ can be detected by the QECC – i.e. the ones that anti-commute with some element of the stabilizer. However, commuting errors are undetectable, and will change the code whenever they are not part of the stabilizer. If we denote by $S^\perp$ the set of Paulis that commute with the stabilizer, i.e.

$$S^\perp := \{P \in \mathcal{G}_n \colon PM = MP \text{ for all } M \in S\}, \tag{10}$$

then the set of undetectable errors that change the data non-trivially is $S^\perp \setminus S$.

Purity testing codes are exactly the stabilizer codes that detect any non-trivial Pauli attack with high probability. This property makes them extremely well suited for constructing authentication schemes as we will see in section 2.4.

Definition 2. A set $\{V_k\}_{k \in \mathcal{K}}$ of stabilizer codes, each with respective stabilizer subgroup S_k, is *ε-purity testing* if, when the code is selected uniformly at random, the probability of any Pauli error $P \in \mathcal{G}_n$ acting non-trivially on the data and not being detected is upper bounded by ε. That is,

$$\Pr_{k \in \mathcal{K}} \left(P \in S_k^\perp \setminus S_k\right) \leq \varepsilon. \tag{11}$$

There is also a more general family of codes that will be useful for our analysis.

Definition 3. Let $\alpha \in (0, 1]$. We say that a family of quantum error-correcting codes $[[n, 1, d]]$ with stabilizer group S and threshold p_{th} has *decay of order* α if there exist constants $\kappa, \beta > 0$ such that

$$\Pr\left(\mathbf{X} \in S^\perp \setminus S\right) \leq \kappa \left(p/p_{\text{th}}\right)^{\beta n^\alpha}, \quad \text{when } p < p_{\text{th}}. \tag{12}$$

Note that the distance of a code is uniquely determined by the size $d = \Theta(n^\alpha)$.

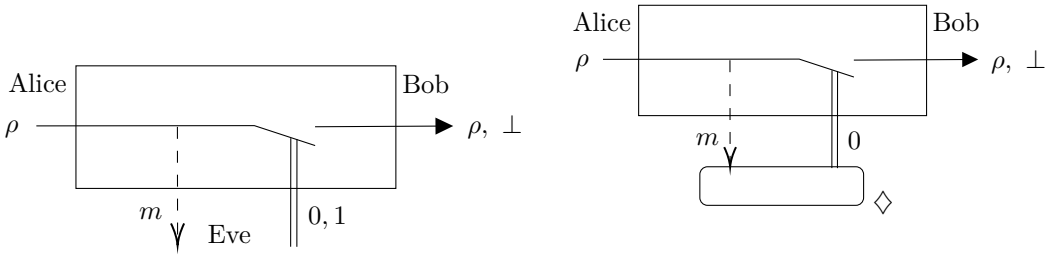
For example, well known *concatenated* codes fall into this category. Given a $[[n, 1, d]]$ QECC, we can recursively encode each encoded qubit in n physical qubits, which can be encoded again such that each layer L of concatenation is a $[[n^L, 1, d^L]]$ QECC, see [Pre99]. After L levels of concatenation, the probability of failed recovery is upper bounded by

$$\Pr\left(\mathbf{X} \in S^\perp \setminus S\right) \leq \binom{n}{t+1}^{-1} \left(\binom{n}{t+1} p\right)^{(t+1)^L}. \tag{13}$$

Note that if $p < p_{\text{th}} := \binom{n}{t+1}^{-1}$, then we can make the failure probability as small as desired by increasing the number of layers. This is, a $[[n^L, 1, d^L]]$ concatenated codes has decay $\alpha := \log_n(t+1)$.

2.4 Quantum authentication

In the context of constructive cryptography, a quantum authentication protocol is expected to construct an *authenticated quantum channel*, $\mathcal{S}$, from nothing but an insecure quantum channel and a secret key source. The goal of a secure quantum channel is to allow Alice to send m qubits to Bob without Eve tampering with the data. On the one hand, they cannot stop Eve from learning that a message has been transmitted nor cutting the communication lines. Hence Eve's actions can be described as a bit 0 when Bob gets the message, and 1



(a) Secure authenticated quantum channel with adversary Eve.

(b) Secure authenticated quantum channel with no adversary present.

Figure 2: Characterization of an authenticated quantum channel $\mathcal{S}_\diamondsuit$.

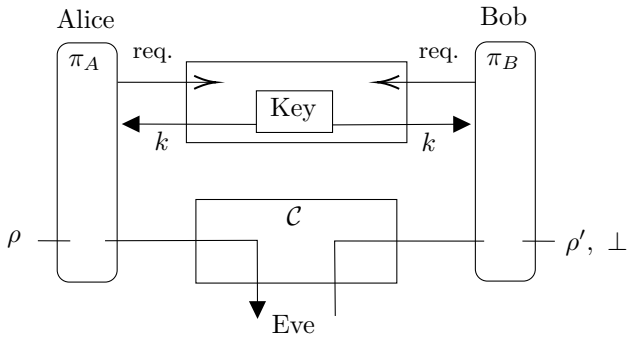


Figure 3: The real system for quantum message authentication.

when he does not. On the other hand, in the presence of no adversary, Eve's interface is substituted by a filter $\diamondsuit_E$ that models an honest behavior, in this case always allowing Bob to receive exactly the message that Alice sent. Figure 2 is a graphical description of the channel $\mathcal{S}_\diamondsuit$.

In order to construct the filtered resource $\mathcal{S}_\diamondsuit$, quantum authentication protocols will use a shared secret key $\mathcal{K}$ and an insecure quantum channel $\mathcal{C}_\#$, here the filter $\diamondsuit_E$ represents an honest behavior instead of an adversary, and the filter $\#_E$ is a noisy channel. After receiving a message ρ, the protocol π_A authenticates it with the key k received from $\mathcal{K}$ and sends the message to the insecure quantum channel $\mathcal{C}$. The protocol π_B upon receiving a message checks its validity with the shared secret key k, and outputs either ρ' or an error message $\perp$. In absence of an adversary, we substitute Eve's interface by a noise filter $\#_E$. Note that for our purposes we are not considering key resources, which greatly simplifies Portmann's descriptions [Por17, Section 3].

A generic way of constructing authentication codes was given by Barnum et al. [BCG+02] using purity-testing codes. In these schemes, the message is first encoded using a $[[n, 1, d]]$ purity-testing error-correcting code, and then encrypted with a quantum one-time pad using the shared secret key, see Protocol 1.

We previously defined a quantum message authentication system as a protocol that constructs an authenticated quantum channel $\mathcal{S}_\diamondsuit$ as in figure 2, from some shared secret key $\mathcal{K}$ and an insecure quantum channel $\mathcal{C}_\#$, where the filter introduces noise. Portmann showed that the scheme from protocol 1 based on purity-testing codes provides quantum authentication protocols, given that the filter is noiseless, denoted $\square_E$.

Theorem 2 ([Por17, Lemma D.1])**.** *Given a δ-purity testing protocol $[[n, 1, d]]$, let $\pi_{AB}^{auth} = (\pi_A, \pi_B)$ denote the converter corresponding to Alice and Bob's protocols 1. Then π_{AB}^{auth} constructs an authenticated quantum channel $\mathcal{S}_\diamondsuit$, given an insecure noiseless quantum channel $\mathcal{C}_\square$ and a secret shared key $\mathcal{K}$ within $(0, \delta^{auth})$, where $\delta^{auth} = \max\{\delta, 2^{-(n-1)}\}$.*

Protocol 1 Quantum 'encode-then-encrypt' message authentication scheme from purity-testing codes.

Setting. At the beginning of the protocol the encoder and decoder have access to secret shared key $(k, l) \in \mathcal{K}$, with $\mathcal{K} := \mathcal{K}_0 \times \mathcal{K}_1$ not necessarily of the same size. Let $\{V_k\}_{k\in\mathcal{K}_0}$ be a family of purity-testing codes.

Encoding.

1: Given input data qubit ρ_d, append the $(n-1)$-qubit syndrome state $|0\rangle\langle 0|^{\otimes(n-1)}$.
2: Encode everything with the purity testing code according to the secret shared key $k \in \mathcal{K}_0$, to obtain $V_k(\rho_d \otimes |0\rangle\langle 0|^{\otimes(n-1)})V_k^\dagger$.
3: Encrypt the message with a quantum one-time pad using the key $l \in \mathcal{K}_1$, the final message is the following

$$\mathrm{Auth}_{k,l}(\rho_d) = P_l V_k(\rho_d \otimes |0\rangle\langle 0|^{\otimes(n-1)})V_k^\dagger P_l. \tag{14}$$

Decoding.

1: First decrypt the data using l.
2: Decode the data according to the error-correcting code V_k given by the key k.
3: Measure the syndrome register in the computational basis. If the measurement outcome is 0, accept the protocol. Else, abort.

That is,

$$\mathcal{C}_\square || \mathcal{K} \xrightarrow{\pi_{AB}^{auth}, (0, \delta^{auth})} \mathcal{S}_\lozenge. \tag{15}$$

2.5 Noisy channels

Given a noisy quantum channel between Alice A and Bob B, where the *noise* is represented by a quantum operation $\mathcal{F}_{A\to B}$, we say that there exists an error-correction protocol π_{AB}^{ecc}, defined by an encoding map $\mathcal{E}_A$ and a decoding map $\mathcal{D}_B$, correcting the errors induced by $\mathcal{F}_{A\to B}$ within $\varepsilon^{\mathrm{ecc}}$, if

$$\frac{1}{2}\left\|\mathcal{D}_B \circ \mathcal{F}_{A\to B} \circ \mathcal{E}_A - I_{A\to B}\right\|_\diamond \leq \varepsilon^{\mathrm{ecc}}. \tag{16}$$

We can rewrite the above statement in the abstract cryptography language.

Lemma 2 ([Por17, Lemma 4.2])**.** *Let $\#_E$ be a filter introducing the noise given by the quantum operation $\mathcal{F}$, and let $\square_E$ be a noiseless filter. If there exists an error correction protocol $\pi_{AB}^{ecc} = (\pi_A^{ecc}, \pi_B^{ecc})$ that corrects the errors induced by $\mathcal{F}$ within ε^{ecc}, then π_{AB}^{ecc} constructs a noiseless channel $\mathcal{C}_\square$, from a noisy channel $\hat{\mathcal{C}}_\#$ within $(\varepsilon^{ecc}, 0)$. That is,*

$$\hat{\mathcal{C}}_\# \xrightarrow{\pi_{AB}^{ecc}, (\varepsilon^{ecc}, 0)} \mathcal{C}_\square. \tag{17}$$

It is now clear what the relevance of splitting the cryptographic security definition in terms of correctness and security is, a direct consequence of theorems 1 and 2, and lemma 2, is that a δ-secure authentication scheme wrapped in an ε-correct error-correcting code constructs an (ε, δ)-secure authenticated quantum channel, instead of an $(\varepsilon + \delta)$-secure from the original composition theorem.

In order to be able to compare explicit schemes, we will restrict to a basic type of noise, typically used in error correction literature, i.i.d. Pauli noise. We will assume that when qubits are sent through a noisy channel, they independently undergo a X, Y or Z Pauli error with probabilities p_X, p_Y and p_Z respectively. This model is interesting not only because it models many interesting real situations, but also because the Pauli operators are a basis of single-qubit operations, and thus protection against i.i.d. Pauli

noise for a single qubit implies protection against any single-qubit error. Moreover, in the 'encode-then-encrypt' authentication scheme, the one-time-pad encryption and the Pauli twirl make any attack become a Pauli attack. This means that for the security proof it is enough to prove security against Pauli attacks [BGS13], which is also the reason why the underlying error-correction codes are required to be purity-testing codes.

Let $\mathcal{F}_n$ denote a quantum noise channel acting on n-qubits independently, which we can write in terms of the basis elements of single-qubit operations $\mathcal{F}_n = \mathcal{F}^{\otimes n}$, where

$$\mathcal{F}(\rho) := (1 - p_X - p_Y - p_Z)\rho + p_X X\rho X^\dagger + p_Y Y\rho Y^\dagger + p_Z Z\rho Z^\dagger. \tag{18}$$

This channel leaves the state untouched with probability $p_I := 1 - p_X - p_Y - p_Z$ and each Pauli operation is applied with probability p_X, p_Y and p_Z respectively. As mentioned earlier, we will assume that the noise is i.i.d. distributed. Therefore, we can write the noise channel acting on a n-qubit register as

$$\mathcal{F}_n(\rho) = \sum_{\substack{k_1+k_2+k_3+k_4=n \\ k_1,k_2,k_3,k_4\geq 0}} \binom{n}{k_1, k_2, k_3, k_4} \prod_{j\in\{I,X,Y,Z\}} p_j^{k_j} \sigma_j^{k_j} \rho (\sigma_j^\dagger)^{k_j}. \tag{19}$$

The depolarizing channel, the most commonly used noise model in error-correction literature [Ter15], is of this type. When a qubit goes through the depolarizing channel, the channel erases the qubit and substitutes it by a completely mixed state $I/2$ with probability p, and leaves the qubit untouched with probability $1 - p$. In notation from equation 19, this is the same as saying that with probability $1 - p$ the qubit is being left untouched, and each Pauli operation will be applied with probability $p/3$. Therefore, the depolarizing noise acting on n-qubits can be written as

$$\begin{aligned} \mathcal{F}_n(\rho) &= \sum_{\substack{k_1+\cdots+k_4=n \\ k_1,\ldots,k_4\geq 0}} \binom{n}{k_1, \ldots, k_4} (1-p)^{k_1} \left(\frac{p}{3}\right)^{n-k_1} \prod_{j\in\{X,Y,Z\}} \sigma_j^{k_j} \rho (\sigma_j^\dagger)^{k_j}, \\ &= (1-p)^n \rho + (1-p)^{n-1} \frac{p}{3} \sum_{k=1}^{n} \sum_{j\in\{I,X,Y,Z\}} \sigma_j^k \rho (\sigma_j^\dagger)^k + \cdots. \end{aligned} \tag{20}$$

3 Explicit composed protocols

Attempting to protect our data against both noise and attacks can be seen in the AC framework of section 2.2 as constructing a noiseless secure channel $\mathcal{C}$ (correct and secure) from a noisy insecure channel $\hat{\mathcal{C}}_\#$. In practice, this is obtained by concatenating an authenticating scheme from section 2.4 with an error-correcting code from section 2.3. Unfortunately, both these constructions involve encoding logical qubits in redundant physical qubits for protection, and thus the dimension of the secure and insecure channels can differ vastly.

In this section we analyse the cost-effectiveness of two of the most used authentication schemes: the trap and Clifford schemes. They are both purity testing code families, see definition 2, i.e. the probability of non-trivial errors being undetected by the codes is low. In short, the trap scheme consists of two error-correcting codes, such that the inner code corrects low-weight noise and the traps detect high-weight attacks, hence the adversary cannot effectively choose a relevant attack without being detected. The Clifford scheme on the other hand is a single error-correcting code that randomizes the weight of the error, actually making it *strong* purity testing, meaning it detects *any* error with high probability.

Taking the amount of qubits necessary as a parameter for efficiency, in this section we analyse the cost-effectiveness of protecting a single qubit of data against both noise and attacks. We study both the trap of the Clifford scheme for the authentication and consider only single-qubit error-correcting codes.

3.1 Trap scheme

Given a fixed $[[n, 1, d]]$ error-correcting code with encoder $\mathrm{Enc}_n \colon \mathbb{C}^2 \to \mathbb{C}^{2^n}$ and decoder $\mathrm{Dec}_n \colon \mathbb{C}^{2^n} \to \mathbb{C}^2$, the trap authentication scheme constructs a set $\{V_k\}_{k\in\mathcal{K}}$ of purity testing codes encoding 1 logical qubit in $3n$ physical qubits, by first encoding each data qubit ρ_d in n physical qubits, appending $2n$ 'traps' to the encoded data (n copies of $|0\rangle\langle 0|$ and another n copies of $|+\rangle\langle +|$); the resulting $3n$-qubit registers are permuted attending to a secret shared key $k \in \mathcal{K}$. The purity-testing codes needed for the encode-then-encrypt scheme in protocol 1 are given by:

$$\begin{aligned} V_k(\rho_d) &= \pi_k(\mathrm{Enc}_n(\rho_d) \otimes |0\rangle\langle 0|^n \otimes |+\rangle\langle +|^n)\pi_k^\dagger \\ &:= \pi_k(I_{2n} \otimes H^{\otimes n})(\mathrm{Enc}_n(\rho_d) \otimes |0\rangle\langle 0|^{2n-1})(I_{2n} \otimes H^{\otimes n})\pi_k^\dagger. \end{aligned} \tag{21}$$

When decoding, first the inverse permutation, according to the secret key, is applied. Finally, the data registers are decoded according to the fixed error-correcting code and the traps are measured in the computational and Hadamard bases respectively. Here we consider the underlying code as error *correcting*, for the sake of fair comparison with later codes, but there is also an error *detection* variant of the trap code [BGS13].

Lemma 3 ([BW16, Theorem 5.2])**.** *The trap code with inner error-correcting code* $[[n, 1, d]]$ *is* $(1/3)^{\frac{d+1}{2}}$*-purity testing.*

Given a purity testing code, the discussion in sections 2.4 and 2.5 ensures us that if there exists an error-correcting protocol correcting the errors induced by the noisy channel within error ε^{ecc}, then composing qubit-wise the trap code encoded qubits with the error-correcting code will give rise to an ε^{ecc}-correct and $(1/3)^{\frac{d+1}{2}}$-secure protocol. The following theorem rephrases the security and correctness of the composed protocol in terms of the number of qubits. For simplicity of the analysis we assume the channel is affected by depolarizing noise.

Proposition 1. *Let* π^{trap} *be the trap authentication scheme with a family* $[[n_{in}, 1, d_{in}]]$ *of inner codes of decay* α_{in}*, and* π^{ecc} *a family* $[[n_{out}, 1, d_{out}]]$ *of outer codes of decay* α_{out}*, with* p_{in} *and* p_{out} *thresholds respectively and* $\kappa_{in} < p_{out}$*. Let* $\#_E$ *be a filter introducing the noise given by the depolarizing channel with channel error* $p < p_{in}$ *and* $p < p_{out}$*. Then to obtain* ε*-correctness and* δ*-security, i.e.*

$$\hat{\mathcal{C}}_{\#}||\mathcal{K} \xrightarrow{\pi^{trap}\pi^{ecc},(\varepsilon,\delta)} \mathcal{S}_\diamondsuit^m, \tag{22}$$

it is sufficient for the total amount of qubits n_{total} *to grow as*

$$n_{total} = \Omega\left(\log(1/\varepsilon)^{1/\alpha_{out}} \log(1/\delta)^{1/\alpha_{in}}\right). \tag{23}$$

Proof. Let us denote by S_{in} the stabilizer subgroup of the inner code and by S the one of the trap code concatenated with the error-correcting code.
Security. Since the inner code uniquely determines the security it is natural to start with it. By lemma 3, given $P \in \mathcal{G}_{n_{\mathrm{in}}}$, δ-security is obtained whenever

$$\Pr_{k\in\mathcal{K}}\left(P \in (S_{\mathrm{in}})_k^\perp \setminus (S_{\mathrm{in}})_k\right) \leq \left(\frac{1}{3}\right)^{\frac{\Theta\left(n_{\mathrm{in}}^{\alpha_{\mathrm{in}}}\right)+1}{2}} \leq \delta. \tag{24}$$

Taking logarithms on both sides we obtain,

$$\frac{\Theta\left(n_{\mathrm{in}}^{\alpha_{\mathrm{in}}}\right)+1}{2} \geq \frac{\log(1/\delta)}{\log(3)} \quad \text{i.e.} \quad n_{\mathrm{in}} \geq \Omega\left(\log(1/\delta)^{1/\alpha_{\mathrm{in}}}\right). \tag{25}$$

Correctness. Note in equation 21 how the first $n_{\text{in}}n_{\text{out}}$ qubits have double encoding, and the last $2n_{\text{in}}n_{\text{out}}$ only one, thus the probability of being detected for $n_{\text{in}}n_{\text{out}}$ of the qubits is not uniquely determined by the outer error-correcting code. Then the error probability is bounded by

$$\begin{aligned}\Pr\big(\mathbf{X} \in S^{\perp} \setminus S\big) &\leq \kappa_{\text{out}} \left(\Pr\big(\mathbf{X} \in S_{\text{in}}^{\perp} \setminus S_{\text{in}}\big)/p_{\text{out}}\right)^{n^{\alpha_{\text{out}}}} + 2n_{in}\kappa_{\text{out}}(p/p_{\text{out}})^{n^{\alpha_{\text{out}}}} \\ &\leq \kappa_{\text{out}} \left(\frac{\kappa_{\text{in}}}{p_{\text{out}}}\right)^{n_{\text{out}}^{\alpha_{\text{out}}}} \left(\frac{p}{p_{\text{in}}}\right)^{n_{\text{in}}^{\alpha_{\text{in}}} n_{\text{out}}^{\alpha_{\text{out}}}} + 2n_{in}\kappa_{\text{out}} \left(\frac{p}{p_{\text{out}}}\right)^{n_{\text{out}}^{\alpha_{\text{out}}}},\end{aligned} \tag{26}$$

thus we achieve ε-correctness whenever $\Pr\big(\mathbf{X} \in S^{\perp} \setminus S\big) \leq \varepsilon$, thus it would be enough for both following conditions to hold

$$\begin{cases} \kappa_{\text{out}} \left(\frac{\kappa_{\text{in}}}{p_{\text{out}}}\right)^{n_{\text{out}}^{\alpha_{\text{out}}}} \left(\frac{p}{p_{\text{in}}}\right)^{n_{\text{in}}^{\alpha_{\text{in}}} n_{\text{out}}^{\alpha_{\text{out}}}} \leq \varepsilon/2 \\ 2n_{in}\kappa_{\text{out}} \left(\frac{p}{p_{\text{out}}}\right)^{n_{\text{out}}^{\alpha_{\text{out}}}} \leq \varepsilon/2. \end{cases} \tag{27}$$

For the first item in equation 27, by taking logarithms we need

$$n_{\text{out}}^{\alpha_{\text{out}}} \log\left(\frac{p_{\text{out}}}{\kappa_{\text{in}}}\right) + n_{\text{in}}^{\alpha_{\text{in}}} n_{\text{out}}^{\alpha_{\text{out}}} \log\frac{p_{\text{in}}}{p} \geq \log\left(\frac{2\kappa_{\text{out}}}{\varepsilon}\right), \tag{28}$$

thus it is enough for any of the following two to hold,

$$n_{\text{out}}^{\alpha_{\text{out}}} \log\left(\frac{p_{\text{out}}}{\kappa_{\text{in}}}\right) \geq \log\left(\frac{2\kappa_{\text{out}}}{\varepsilon}\right) \quad \text{and} \quad n_{\text{in}}^{\alpha_{\text{in}}} n_{\text{out}}^{\alpha_{\text{out}}} \log\frac{p_{\text{in}}}{p} \geq \log\left(\frac{2\kappa_{\text{out}}}{\varepsilon}\right). \tag{29}$$

Note that the first requirement is weaker than the second one, thus it is enough to ask

$$n_{\text{out}} \geq \Omega\left(\log(1/\varepsilon)^{1/\alpha_{\text{out}}}\right). \tag{30}$$

For the second item in equation 27, by taking logarithms we need

$$n_{\text{out}}^{\alpha_{\text{out}}} \log\left(\frac{p_{\text{out}}}{p}\right) \geq \log\left(\frac{4n_{\text{in}}\kappa_{\text{out}}}{\varepsilon}\right) \quad \text{i.e.} \quad n_{\text{out}} \geq \Omega\left(\left[\log(n_{\text{in}}) + \log\left(\frac{1}{\varepsilon}\right)\right]^{1/\alpha_{\text{out}}}\right), \tag{31}$$

but this requirement is weaker than the one in equation 30, thus proving the desired bound. □

3.2 Clifford scheme

The Clifford authentication scheme is based on a set of purity testing unitaries $\{C_k\}_{k\in\mathcal{K}}$ given by the Clifford group. The authentication scheme first appends to the data qubit ρ_d, n 'traps' (n copies of $|0\rangle\langle 0|$) and finally a Clifford element C_k is applied to the resulting $(n+1)$-qubit registers attending to a secret shared key $k \in \mathcal{K}$. Without the need for encryption, the authenticated data is directly given by:

$$\text{Auth}_k(\rho_d) := C_k(\rho_d \otimes |0\rangle\langle 0|^n)C_k^{\dagger}. \tag{32}$$

When decoding, first the inverse Clifford operation, according to the secret key, is applied. Finally, the traps are measured in the computational basis.

It is not difficult to see that the n-trap Clifford authentication scheme is 2^{-n}-secure. However, the Clifford twirl will map any Pauli operation to an arbitrary-weight one, not being able to distinguish between low and high weight operations and hence making it impractical over noisy channels. It is therefore imperative to compose it with an error-correcting code for practical uses.

Proposition 2. *Let π^c be the n_{in}-qubit Clifford authentication scheme, and π^{ecc} a family $[[n_{out}, 1, d_{out}]]$ of outer codes of decay α_{out}, with p_{out} threshold. Let $\#_E$ be a filter introducing the noise given by the depolarizing channel with channel error $p < p_{out}$. Then to obtain ε-correctness and δ-security, i.e.*

$$\hat{\mathcal{C}}_{\#}||\mathcal{K} \xrightarrow{\pi^c\pi^{ecc},(\varepsilon,\delta)} \mathcal{S}_{\diamondsuit}^m, \tag{33}$$

it is sufficient for the total amount of qubits n_{total} to grow as

$$n_{total} = \Omega\left(\log(1/\varepsilon)^{1/\alpha_{out}} \log(1/\delta)\right). \tag{34}$$

Proof. Let us denote by S_c the stabilizer subgroup of the Clifford code and by S the one of the Clifford code concatenated with the error-correcting code.
Security. Since the n_{in}-trap Clifford code is $2^{-n_{\text{in}}}$ purity-testing, we obtain δ-security whenever

$$\Pr_{k\in\mathcal{K}}\left(P \in (S_c)_k^{\perp} \setminus (S_c)_k\right) \leq 2^{-n_{\text{in}}} \leq \delta \quad \text{i.e.} \quad n_{\text{in}} \geq \Omega\left(\log(1/\delta)\right). \tag{35}$$

Correctness. Note that none of the errors can permeate the outer error-correcting code, because the Clifford operation will map it to an arbitrary weight Pauli operation and therefore will be detected by the traps. Hence we achieve ε-correctness whenever

$$\Pr\left(\mathbf{X} \in S^{\perp} \setminus S\right) \leq (n_{\text{in}}+1)\kappa_{\text{out}} \left(\frac{p}{p_{\text{out}}}\right)^{n_{\text{out}}^{\alpha_{\text{out}}}} \leq \varepsilon. \tag{36}$$

Taking logarithms on both sides we obtain,

$$n_{\text{out}}^{\alpha_{\text{out}}} \geq \log\left(\frac{p_{\text{out}}}{p}\right)^{-1} \log\left(\frac{(n_{\text{in}}+1)\kappa_{\text{out}}}{\varepsilon}\right) \quad \text{i.e.} \quad n_{\text{out}} \geq \Omega\left(\log(1/\varepsilon)^{1/\alpha_{\text{out}}}\right). \tag{37}$$

□

4 The threshold authentication scheme

Both Hayden, Leung and Mayers [HLM16] and Portmann [Por17] constructions of composed protocols, it is assumed that the authentication scheme rejects whenever an error is present – which is always the case with very high probability when sending information through noisy channels – and therefore an error-correcting code is necessary to make the schemes useful. However, from the structure of the composition, the number of qubits used in such a construction blows up both with the size of the purity-testing code used in the authentication scheme and the error-correcting code. It is therefore natural to ask if such a composition is even necessary, and if we cannot design a protocol that directly constructs an authenticated quantum channel from a noisy insecure channel and shared secret key. This is exactly what the threshold scheme we propose in this section does.

The threshold scheme is an adaptation of the trap scheme where, with the same encoding, we require Bob to accept the message whenever *low* amount of errors are detected. In other words, we use the traps as they were originally intended, to measure the amount of error present in the encoded data and decide if these errors pertain to noise or an attack. In principle, this should not be enough, as the correctable errors of an error-correcting code grow sub-linearly in the size of the code, while for example for the depolarizing channel the number of errors is linear in the size of the code. However, error-correcting codes with fixed decay actually correct linear amount of errors with *very high probability*, which is enough to form a purity-testing family of codes.

The threshold code is constructed as follows. Given a fixed $[[n, 1, d]]$ error-correcting code with encoder $\mathrm{Enc}_n \colon \mathbb{C}^2 \to \mathbb{C}^{2^n}$ and decoder $\mathrm{Dec}_n \colon \mathbb{C}^{2^n} \to \mathbb{C}^2$, the threshold authentication scheme first encodes each data qubit ρ_d in n physical qubits and then appending $2n$ 'traps' to the encoded data (n copies of $|0\rangle\langle 0|$ and another n copies of $|+\rangle\langle +|$); the resulting $3n$-qubit registers are permuted attending to a secret shared key $k \in \mathcal{K}$. The authenticated states are given by:

$$V_k(\rho_d) := \pi_k(\mathrm{Enc}_n(\rho_d) \otimes |0\rangle\langle 0|^n \otimes |+\rangle\langle +|^n)\pi_k^\dagger. \tag{38}$$

When decoding, first the inverse permutation, according to the secret key, is applied. Finally, the data registers are decoded according to the fixed error-correcting code and the traps are measured in the computational and Hadamard bases respectively. However, *only if less than a threshold* τn, with $\tau \in [0, 2]$, of errors are present does the receiver accept the message. The threshold τn allows us to tune the the amount of error we are willing to accept, depending on the noise of the channel, more efficiently than adding an entire new error-correcting code to each qubit. The explicit construction of the threshold authentication scheme is given in Protocol 2.

Protocol 2 Threshold authentication scheme π_{AB}^{thr}.

Setting. At the beginning of the protocol the encoder and decoder have access to secret shared key $(k, l) \in \mathcal{K}$, with $\mathcal{K} := \mathcal{K}_0 \times \mathcal{K}_1$ not necessarily of the same size. Let $(\mathrm{Enc}_n, \mathrm{Dec}_n)$ be an $[[n, 1, d]]$ error-correcting code and $\{\pi_k\}_{k \in \mathcal{K}_0}$ a family of permutations.

Encoding.

1: Encode the input data qubit into n qubits with the fixed error correcting code $\mathrm{Enc}_n(\rho_d)$.
2: Append $2n$ computational basis states $|0\rangle\langle 0|^{\otimes 2n}$ and apply a Hadamard gate to each of the last n qubits.
3: Apply the permutation π_k to all the qubit registers according to the secret key $k \in \mathcal{K}_0$.
4: Finally, encrypt the message with a quantum one-time pad using the secret key $l \in \mathcal{K}_1$, obtaining thus

$$\mathrm{Auth}_{k,l}(\rho_d) = P_l \pi_k \left(\mathrm{Enc}_n(\rho_d) \otimes |0\rangle\langle 0|^{\otimes n} \otimes |+\rangle\langle +|^{\otimes n}\right) \pi_k^\dagger P_l. \tag{39}$$

Decoding.

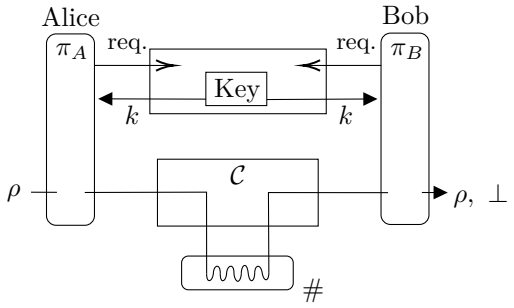
1: First decrypt the data using l.
2: Apply the inverse permutation according to k.
3: Measure the second to last n registers in the computational basis, and the last n registers in the Hadamard basis. If less than a threshold τn of qubits differ from the expected outcome, i.e. $|0\rangle\langle 0|^n \otimes |+\rangle\langle +|^n$, accept the protocol. Else, abort.
4: If the protocol has been accepted, decode the data register with the decoder Dec_n.

Since there is no outer error-correcting code in our protocol, we have to ensure that the threshold scheme constructs directly a noiseless secure quantum channel from nothing but a noisy insecure quantum channel and a shared secret key. We will separate this task in two steps, first proving the correctness and then the security.

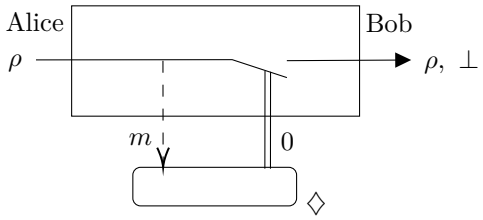
4.1 Correctness

We want to prove that when we use our protocol with a noisy channel, the outcome is nearly indistinguishable from using a noiseless secure channel without adversary, see figure 4.

Proposition 3. *Let $\#_E$ be a filter introducing the noise given by the depolarizing channel with channel error p. Let $[[n, 1, d]]$ be a family of error-correcting codes with threshold*



(a) Threshold protocol with no adversary present.



(b) Authenticated quantum channel with no adversary present.

Figure 4: Comparison between the threshold protocol in a noisy channel and a secure authenticated quantum channel without adversary.

$p_{th} > p$ and decay α. Let π_{AB}^{thr} be the threshold authentication scheme built from these error-correcting codes and with threshold parameter $\tau > 4p/3$ as in protocol 2. Then π_{AB}^{thr} is ε-correct, i.e.

$$d(\pi_{AB}^{thr}(\mathcal{C}_{\#}||\mathcal{K}), \mathcal{S}_{\diamondsuit}) \leq \varepsilon, \tag{40}$$

with

$$\varepsilon = \kappa 2^{-\beta n^{\alpha} \log(p_{th}/p)} + 2^{-n(\tau - 4p/3)^2 \log(e)}. \tag{41}$$

Proof. To prove correctness within ε we have to show that the threshold protocol π_{AB}^{thr} constructs a noiseless secure channel $\mathcal{S}_{\diamondsuit}$ such that the real system transmitted through a noisy channel $\pi_{AB}^{\text{thr}}(\mathcal{C}_{\#}||\mathcal{K})$ cannot be distinguished from the ideal system $\mathcal{S}_{\diamondsuit}$. Note that distinguishability in presence of no adversary is exactly the diamond norm between the identity map and the encoding-noise-decoding map of the threshold code, i.e.

$$d(\pi_{AB}^{\text{thr}}(\mathcal{C}_{\#}||\mathcal{K}), \mathcal{S}_{\diamondsuit}) = \frac{1}{2} \left\| \mathcal{D}^{\text{thr}} \circ \mathcal{F} \circ \mathcal{E}^{\text{thr}} - I \right\|_{\diamond}. \tag{42}$$

Given a random variable $X \in \{I, X, Z, Y\}$, let $\omega(X) \in \{0, 1\}$ be a random variable denoting if the operator describes a non-trivial error on a qubit or not, i.e.

$$\omega(X_j) = \begin{cases} 1 & \text{if } X_j \neq I \\ 0 & \text{if } X_j = I. \end{cases} \tag{43}$$

Let $X_1, \ldots, X_{3n}$ be independent random variables such that the first n fail with probability p and the last $2n$ fail with probability $2p/3$, i.e.

$$\Pr(\omega(X_j) = 1) = \begin{cases} p & \text{for } j = 1, \ldots, n, \\ 2p/3 & \text{for } j = n+1, \ldots, 3n. \end{cases} \tag{44}$$

We do this distinction because we have computational and Hadamard bases traps, thus the probability of rejection is different. We denote by $\mathbf{X}$ the tensor product of the first n variables, i.e. $\mathbf{X} := X_1, \otimes \cdots \otimes X_n$. We achieve ε-correctness whenever the rejection probability of the traps or the failed recovery of the error-correcting code encoding the data qubits is less than ε. That is,

$$\begin{aligned} \Pr\left(\left\{ \mathbf{X} \in S^{\perp} \setminus S \right\} \cup \left\{ \sum_{j=n}^{3n} \omega(X_j) \geq \tau n \right\} \right) &= \Pr\left(\mathbf{X} \in S^{\perp} \setminus S \right) + \Pr\left(\sum_{j=n}^{3n} \omega(X_j) \geq \tau n \right) \\ &\leq \kappa \left(p/p_{\text{th}} \right)^{\beta n^{\alpha}} + \exp\left(-n(\tau - 4p/3)^2 \right) \end{aligned} \tag{45}$$

whenever $p < p_{\text{th}}$, where we used Hoeffding's inequality with $\tau > 4p/3$. □

4.2 Security

With security we mean that, in presence of a malicious player, there exists a simulator in the 'ideal protocol' that is indistinguishable from the 'real protocol'. However, instead of constructing this simulator, it is enough to show that the threshold scheme constructs a set of codes that is purity testing, which will provide us with security by theorem 2. Although Portmann's original proof constructs a secure channel from a noiseless channel, in the security proof the filters are substituted by an adversary, and therefore work for our setting as well.

By setting the threshold properly, we can leverage the fact that error-correcting codes correct a linear amount of errors with high probability to prove that the threshold scheme is purity testing.

Proposition 4. *Let $\#_E$ be a filter introducing the noise given by the depolarizing channel with channel error p. Let $[[n, 1, d]]$ be a family of error-correcting codes with threshold $p_{th} > p$ and decay α. Let π_{AB}^{thr} be the threshold authentication scheme built from these error-correcting codes and with threshold parameter $\tau < 2p_{th}$ as in protocol 2. Then π_{AB}^{thr} is δ-secure, i.e. there exists a simulator σ_E, δ-close to the real protocol*

$$d(\pi_{AB}^{rm}\mathcal{C}, \sigma_E\mathcal{S}) \leq \delta, \tag{46}$$

with

$$\delta = \max\left\{\frac{10\kappa}{9\sqrt{2\pi}\sqrt{3b(1-b)}}\exp\left(-\frac{\ln(n)}{2} - \beta n^{\alpha}\ln(p_{th}/b)\right), \exp\left(-n\frac{3b}{2}\left(1-\frac{\tau}{2b}\right)^2\right)\right\}, \tag{47}$$

where $b := \frac{p_{th}}{2} + \frac{\tau}{4}$.

Proof. We will prove security by showing that the threshold scheme constructs a family of purity-testing codes, this is, that if the permutation key is selected uniformly at random, the probability of any Pauli error $E \in \mathcal{G}_n$ acting non-trivially on the data and not being detected is upper bounded by δ. The threshold code, for a key $k \in \mathcal{K}_0$, is characterized by the codes

$$V_k(\rho_d) = \pi_k(\text{Enc}_n(\rho_d) \otimes |0\rangle\langle 0|^n \otimes |+\rangle\langle +|^n)\pi_k^\dagger. \tag{48}$$

The first n qubits are used to decode the inner error-correcting code, and the last $2n$ are the traps, such that the protocol rejects whenever more than τn non-zero traps are detected. Let us denote by S_{in} the stabilizer subgroup of the inner error-correcting code. For a particular permutation π_k, on the one hand, the set of Paulis that are not detected is

$$S_k^\perp := \{\pi_k^\dagger(P \otimes Q \otimes R)\pi_k \colon P \in S_{\text{in}}^\perp,\ \omega_X(Q) + \omega_Z(R) \leq r\}. \tag{49}$$

On the other hand, since the traps are invariant to Z and X operations respectively, the Paulis that act trivially on the message are

$$S_k := \{\pi_k^\dagger(P \otimes Q \otimes R)\pi_k \colon P \in S_{\text{in}},\ Q \in \{I, Z\}^{\otimes n},\ R \in \{I, X\}^{\otimes n}\}. \tag{50}$$

We can also split the set of permutations in terms of the error correction and the trap detection

$$\begin{aligned}\Pi^0(E) &:= \{\pi \in \Pi_{3n} \colon E = \pi^\dagger(P \otimes T)\pi,\ P \in S_{\text{in}}^\perp \setminus S_{\text{in}},\ T \in \mathcal{G}_{2n}\},\\ \Pi^1(E) &:= \{\pi \in \Pi_{3n} \colon E = \pi^\dagger(P \otimes Q \otimes R)\pi,\ P \in \mathcal{G}_n,\ \omega_X(Q) + \omega_Z(R) \leq \tau n\},\end{aligned} \tag{51}$$

so that we can bound the purity testing parameter by the minimum size of both sets

$$\Pr_{k\in\mathcal{K}_0}\left(E \in S_k^{\perp} \setminus S_k\right) \leq \min\left\{\frac{\left|\Pi^0(E)\right|}{|\Pi_{3n}|}, \frac{\left|\Pi^1(E)\right|}{|\Pi_{3n}|}\right\}. \tag{52}$$

Given an operator $E \in \mathcal{G}_{3n}$, we will divide the proof in two cases attending to its weight, where the weight of an operator is defined as the sum of the weight of the elements in its tensor product,

$$\omega(E) := \sum_{j=1}^{3n} \omega(E_j), \quad \text{where} \quad \omega(E_j) := \begin{cases} 1 & \text{if } E_j \neq I \\ 0 & \text{if } E_j = I. \end{cases} \tag{53}$$

Note that it is enough to bound one of the sets from equation 52 for different weight attacks.

Case 1: $\omega(E) \leq 3np_{\text{th}}$. For low-weight attacks, still linear in the total size of the protocol, we expect the error-correcting code to correct them with high probability. Since the set of Pauli operators acting non-trivially and being undetected is exactly the same as the set of operators that the error-correcting code fails to decode correctly and they are randomized because of the one-time pad, we can rewrite it in terms of random variables. We define a set of i.i.d. random variables $X_1, \ldots, X_{3n}$ such that

$$\Pr(\omega(X_j) = 1) = \frac{\omega(E)}{3n} \quad \text{for } j \in [n]. \tag{54}$$

We denote by $\mathbf{X}$ the tensor product of the first n variables, i.e. $\mathbf{X} := X_1, \otimes \cdots \otimes X_n$. Condition on a fixed amount of registers suffering an error, we have the bound

$$\begin{aligned} \frac{\left|\Pi^0(E)\right|}{|\Pi_{3n}|} &\leq \Pr\left(\mathbf{X} \in S_{\text{in}}^{\perp} \setminus S_{\text{in}} \,\middle|\, \sum_{j=1}^{3n} \omega(X_j) = \omega(E)\right) \leq \frac{\Pr\left(\mathbf{X} \in S_{\text{in}}^{\perp} \setminus S_{\text{in}}\right)}{\Pr\left(\sum_{j=1}^{3n} \omega(X_j) = \omega(E)\right)} \\ &\leq \frac{10\kappa}{9\sqrt{2\pi}(3np_{th})^{\beta n^{\alpha}}} \frac{\omega(E)^{\beta n^{\alpha}}}{\sqrt{\omega(E)(1-\omega(E)/3n)}}. \end{aligned} \tag{55}$$

Case 2: $\omega(E) \geq 3\tau n/2$. High weight attacks will be detected by the traps with high probability, even when a linear amount of them τn are triggered before aborting the protocol. Although we cannot leverage the independence of errors as in the security proof of the trap code, we can apply a sampling variant of the Chernoff bound, see lemma 1. Let us define the total population to be all the registers $A := \{1, \ldots, 3n\}$, and the sub-population the traps $B := \{n, \ldots, 3n\}$. Given a Pauli attack E of weight $\omega(E)$ and a random sample $S \subset A$, with $|S| = \omega(E)$, we have

$$\frac{\left|\Pi^1(E)\right|}{|\Pi_{3n}|} \leq \Pr\left(\sum_{j=n}^{3n} \omega(E_j) < \tau n\right) = \Pr\left(\sum_{j=n}^{3n} \omega(E_j) < (1-\gamma)\frac{|B|}{|A|}\omega\right) \tag{56}$$

$$< \exp\left(-\gamma^2 \frac{|B|}{|A|}\frac{\omega(E)}{2}\right) = \exp\left(-\frac{\omega(E)}{3}\left(1 - \frac{3\tau n}{2\omega(E)}\right)^2\right), \tag{57}$$

where $\gamma = 1 - \frac{3\tau n}{2\omega}$, with $\gamma \in (0,1)$ whenever $\omega(E) > 3\tau n/2$. Finally, since the adversary will pick the optimal weight, we need a bound independent of the weight $\omega(E)$. We can ensure this by choosing a non-empty overlap of the two cases, which holds whenever $\tau < 2p_{th}$, and picking a weight in the middle of the accepted ones e.g. $\omega(E) = 3n\left(\frac{p_{th}}{2} + \frac{\tau}{4}\right) = 3nb$.

This provides an upper bound on the purity-testing parameter

$$\begin{aligned}&\Pr_{k\in\mathcal{K}_0}\left(E\in S_k^{\perp}\setminus S_k\right)\\&\leq\max\left\{\frac{10\kappa}{9\sqrt{2\pi}\sqrt{3b(1-b)}}\exp\left(-\frac{\ln(n)}{2}-\beta n^{\alpha}\ln(p_{th}/b)\right),\exp\left(-n\frac{3b}{2}\left(1-\frac{\tau}{2b}\right)^2\right)\right\}.\end{aligned}\tag{58}$$

□

4.3 Efficiency in terms of qubits

We can combine the correctness and security requirements of the threshold scheme to obtain the sufficient amount of qubits that the threshold scheme requires to obtain (ε,δ)-security. The effectiveness of the threshold scheme lies in the fact that instead of encoding the traps in an error-correcting code to allow some error, we can just tune the threshold parameter τ to the noise scenario whilst being able to detect adversaries. That is, in contrast to the composed authentication and error correction, we can construct a secure quantum channel from a noisy insecure channel and secret key without the need to double encode our qubits in two error-correcting codes.

Theorem 3. *Let $\#_E$ be a filter introducing the noise given by the depolarizing channel with channel error p. Let $[[n,1,d]]$ be a family of error-correcting codes with threshold $p_{th}>p$ and decay α. Let π_{AB}^{thr} be the threshold authentication scheme built from these error-correcting codes and with threshold parameter $\tau\in\left(\frac{4p}{3},2p_{th}\right)$ as in protocol 2. Then for π_{AB}^{thr} to obtain ε-correctness and δ-security, i.e.*

$$\hat{\mathcal{C}}_{\#}||\mathcal{K}\xrightarrow{\pi^{thr},(\varepsilon,\delta)}\mathcal{S}_{\diamondsuit},\tag{59}$$

it is sufficient for the total amount of qubits n_{total} to grow as

$$n_{total}\geq\Omega\left(\max\left\{\log(1/\varepsilon)^{1/\alpha},\log(1/\delta)^{1/\alpha}\right\}\right).\tag{60}$$

Proof. In order to obtain ε-correctness, from proposition 3 we need

$$\kappa 2^{-\beta n^{\alpha}\log(p_{\text{th}}/p)}+2^{-n(\tau-4p/3)^2\log(e)}\leq\varepsilon,\tag{61}$$

thus would be enough if both following conditions held,

$$\begin{cases}\kappa 2^{-\beta n^{\alpha}\log(p_{\text{th}}/p)}\leq\varepsilon/2\\2^{-n(\tau-4p/3)^2\log(e)}\leq\varepsilon/2.\end{cases}\tag{62}$$

For the first item in equation 62, by taking logarithms we need

$$n^{\alpha}\geq\frac{1}{\beta\log(p_{\text{th}}/p)}\log\left(\frac{2\kappa}{\varepsilon}\right),\tag{63}$$

and for the second item in equation 62, also by taking logarithms

$$n\geq\frac{1}{(\tau-4p/3)^2\log(e)}\log(2/\varepsilon).\tag{64}$$

Since the second condition is weaker, both conditions in equation 62 will hold whenever

$$n\geq\Omega(\log(1/\varepsilon)^{1/\alpha}).\tag{65}$$

For δ-security, from proposition 4 we need

$$\max\left\{\frac{10\kappa}{9\sqrt{2\pi}\sqrt{3b(1-b)}}\exp\left(-\frac{\ln(n)}{2}-\beta n^{\alpha}\ln(p_{th}/b)\right),\exp\left(-n\frac{3b}{2}\left(1-\frac{\tau}{2b}\right)^{2}\right)\right\}\leq\delta. \tag{66}$$

Note that the above conditions can be simplified as the following two equations holding

$$\begin{cases}A\exp(-\ln(n)/2)\exp(-Bn^{\alpha})\leq\delta\\ \exp(-nC)\leq\delta\end{cases} \tag{67}$$

for the constants

$$A:=\frac{10\kappa}{9\sqrt{2\pi}\sqrt{3b(1-b)}},\quad B:=\beta\ln(p_{th}/b)\quad\text{and}\quad C:=\frac{3b}{2}\left(1-\frac{\tau}{2b}\right)^{2}. \tag{68}$$

By taking logarithms, we see that the first item in equation 67 holds whenever

$$\ln(n)/2+Bn^{\alpha}\geq\ln(A/\delta),\quad\text{i.e.}\quad n\geq\Omega\left(\log(1/\delta)^{1/\alpha}\right). \tag{69}$$

The second item in equation 67 holds whenever

$$n\geq\frac{1}{C}\ln(1/\delta), \tag{70}$$

a weaker condition than equation 69, thus reaching the desired conclusion. □

5 Conclusions

We studied the combination of authentication and error-correction in a single primitive, and we saw that the size blowups of authentication and error correction are (slightly-more than) *multiplied* in a naively composed protocol to determine the total blowup. As an example of the potential of looking at these properties together, we designed the threshold scheme, for which the resource usage is only dependent on the maximum blowup of the two functionalities.

Code	**Decay of α_{in} and α_{out}**	$\varepsilon=\kappa\delta$ or $\varepsilon=\delta^{\kappa}$ for $\kappa>1$
Threshold	$\max\{O(\ln(1/\varepsilon)^{1/\alpha_{\text{in}}}),O(\ln(1/\delta)^{1/\alpha_{\text{in}}})\}$	$O(\ln(1/\delta)^{1/\alpha_{\text{in}}})$
Trap	$O(\ln(1/\varepsilon)^{1/\alpha_{\text{out}}}\ln(1/\delta)^{1/\alpha_{\text{in}}})$	$O(\ln(1/\delta)^{1/\alpha_{\text{in}}+1/\alpha_{\text{out}}})$
Clifford	$O(\ln(1/\varepsilon)^{1/\alpha_{\text{out}}}\ln(1/\delta))$	$O(\ln(1/\delta)^{1/\alpha_{\text{out}}+1})$

Figure 5: Comparison for $[n,1,d]$ QECC.

Given an inner error correcting code of decay α_{in} and an outer one of decay α_{out}, we see that in the usual scenario where the security is more important than the error, $\delta=\varepsilon^{1/\kappa}$ as in the third column of figure 5, the threshold scheme performs better than both the trap and Clifford scheme.

Note that if we have a ECC code of decay $\alpha=1$, meaning that the distance of the code is linear in the size (as good as it can get), then the Clifford code is not better than the trap code (while the trap code has more functionalities), but the threshold code still have a polylogarithmic lower requirement in number of qubits to obtain the same security and correctness.

We leave as an open question what is the maximum gain in efficiency of combining these functionalities. For instance, it is an interesting question whether it is possible to

make the Clifford code error-robust in a more efficient way, or if the threshold code is the optimal code. Our analysis opens the door to combining more general error-correction and authentication codes, which could improve the practicality of the resulting scheme.

In the current work we only considered the notions of information-theoretic security where the integrity of the plaintext is important, and we do not study key recycling. It could be interesting to combine some of these notions – for instance, to construct a computationally-secure scheme for authentication which also functions as error-correcting code in an efficient way.

Additionally, a code which is both error-correcting and authenticating is in some sense the *opposite* of ciphertext authentication. Therefore it could be interesting to consider if there is a natural way of combining these functionalities, and what the maximum amount of key recycling possible is.

References

[ABOE08] Dorit Aharonov, Michael Ben-Or, and Elad Eban. Interactive Proofs For Quantum Computations. *Preprint*, 2008. doi:https://arxiv.org/abs/0810.5375.

[ADSS17] Gorjan Alagic, Yfke Dulek, Christian Schaffner, and Florian Speelman. Quantum fully homomorphic encryption with verification. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 438–467. Springer, Cham, December 2017. doi:10.1007/978-3-319-70694-8_16.

[AGM18] Gorjan Alagic, Tommaso Gagliardoni, and Christian Majenz. Unforgeable quantum encryption. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part III*, volume 10822 of *LNCS*, pages 489–519. Springer, Cham, April / May 2018. doi:10.1007/978-3-319-78372-7_16.

[AGM21] Gorjan Alagic, Tommaso Gagliardoni, and Christian Majenz. Can you sign a quantum state? *Quantum*, 5:603, 2021. doi:10.22331/q-2021-12-16-603.

[BCG+02] H. Barnum, C. Crepeau, D. Gottesman, A. Smith, and A. Tapp. Authentication of quantum messages. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 449–458, 2002. arXiv:0205128, doi:10.1109/SFCS.2002.1181969.

[BCG+06] Michael Ben-Or, Claude Crépeau, Daniel Gottesman, Avinatan Hassidim, and Adam Smith. Secure multiparty quantum computation with (only) a strict honest majority. In *47th FOCS*, pages 249–260. IEEE Computer Society Press, October 2006. doi:10.1109/FOCS.2006.68.

[BGS13] Anne Broadbent, Gus Gutoski, and Douglas Stebila. Quantum one-time programs - (extended abstract). In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 344–360. Springer, Berlin, Heidelberg, August 2013. doi:10.1007/978-3-642-40084-1_20.

[BJSW16] Anne Broadbent, Zhengfeng Ji, Fang Song, and John Watrous. Zero-knowledge proof systems for QMA. In Irit Dinur, editor, *57th FOCS*, pages 31–40. IEEE Computer Society Press, October 2016. doi:10.1109/FOCS.2016.13.

[BMPZ19] Fabio Banfi, Ueli Maurer, Christopher Portmann, and Jiamin Zhu. Composable and finite computational security of quantum message transmission. In Dennis Hofheinz and Alon Rosen, editors, *TCC 2019, Part I*, volume 11891 of *LNCS*,

pages 282–311. Springer, Cham, December 2019. doi:10.1007/978-3-030-36030-6_12.

[BW16] Anne Broadbent and Evelyn Wainewright. Efficient simulation for quantum message authentication. In Anderson C. A. Nascimento and Paulo Barreto, editors, *ICITS 16*, volume 10015 of *LNCS*, pages 72–91. Springer, Cham, August 2016. doi:10.1007/978-3-319-49175-2_4.

[DGJ+20] Yfke Dulek, Alex B. Grilo, Stacey Jeffery, Christian Majenz, and Christian Schaffner. Secure multi-party quantum computation with a dishonest majority. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 729–758. Springer, Cham, May 2020. doi:10.1007/978-3-030-45727-3_25.

[DNS10] Frédéric Dupuis, Jesper Buus Nielsen, and Louis Salvail. Secure two-party quantum evaluation of unitaries against specious adversaries. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 685–706. Springer, Berlin, Heidelberg, August 2010. doi:10.1007/978-3-642-14623-7_37.

[DNS12] Frédéric Dupuis, Jesper Buus Nielsen, and Louis Salvail. Actively secure two-party evaluation of any quantum operation. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 794–811. Springer, Berlin, Heidelberg, August 2012. doi:10.1007/978-3-642-32009-5_46.

[DS18] Yfke Dulek and Florian Speelman. Quantum ciphertext authentication and key recycling with the trap code. In *13th Conference on the Theory of Quantum Computation, Communication and Cryptography*, Leibniz International Proceedings in Informatics, pages 1:1–1:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. arXiv:1804.02237, doi:10.4230/LIPIcs.TQC.2018.1.

[GH01] Andrew V. Goldberg and Jason D. Hartline. Competitive Auctions for Multiple Digital Goods. In *European Symposium on Algorithms*, pages 416–427. Springer Berlin Heidelberg, 2001. doi:10.1007/3-540-44676-1_35.

[Got96] Daniel Gottesman. Class of quantum error-correcting codes saturating the quantum hamming bound. *Phys. Rev. A*, 54:1862–1868, 1996. doi:10.1103/PhysRevA.54.1862.

[GYZ17] Sumegha Garg, Henry Yuen, and Mark Zhandry. New security notions and feasibility results for authentication of quantum data. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part II*, volume 10402 of *LNCS*, pages 342–371. Springer, Cham, August 2017. doi:10.1007/978-3-319-63715-0_12.

[HLM16] Patrick Hayden, Debbie W. Leung, and Dominic Mayers. The Universal Composable Security of Quantum Message Authentication with Key Recyling. *Preprint*, 2016. doi:https://arxiv.org/abs/1610.09434.

[Mau12] Ueli Maurer. Constructive Cryptography – A New Paradigm for Security Definitions and Proofs. In *Theory of Security and Applications*, Lecture Notes in Computer Science, pages 33–56. Springer Berlin Heidelberg, 2012. doi:10.1007/978-3-642-27375-9.

[MR11] Ueli Maurer and Renato Renner. Abstract Cryptography. In *The Second Symposium on Innovations in Computer Science*, pages 1–21. Tsinghua University Press, 2011. doi:10.1007/978-3-642-27375-9_3.

[NC10] Michael A. Nielsen and Isaac L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, Cambridge, New York, 10th anniversary ed edition, 2010. doi:10.1119/1.1463744.

[Por17] Christopher Portmann. Quantum authentication with key recycling. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part III*, volume 10212 of *LNCS*, pages 339–368. Springer, Cham, April / May 2017. doi:10.1007/978-3-319-56617-7_12.

[Pre99] John Preskill. Lecture notes for Physics 219: Quantum computation. *Caltech Lecture Notes*, 1999. URL: http://theory.caltech.edu/~preskill/ph219/ph219_2020-21.html.

[Ter15] Barbara M. Terhal. Quantum Error Correction for Quantum Memories. *Reviews of Modern Physics*, 87(2):307–346, 2015. arXiv:1302.3428, doi:10.1103/RevModPhys.87.307.