Structure-Preserving and Re-randomizable RCCA-secure Public Key Encryption and its Applications

Antonio Faonio[1], Dario Fiore[1], Javier Herranz[2], and Carla Ràfols[3]

[1] IMDEA Software Institute ⋆ ,
[2] Cybercat and Universitat Politècnica de Catalunya ⋆⋆
[3] Cybercat and Universitat Pompeu Fabra ⋆⋆⋆

Abstract. Re-randomizable RCCA-secure public key encryption (Rand-RCCA PKE) schemes reconcile the property of re-randomizability of the ciphertexts with the need of security against chosen-ciphertexts attacks. In this paper we give a new construction of a Rand-RCCA PKE scheme that is perfectly re-randomizable. Our construction is structure-preserving, can be instantiated over Type-3 pairing groups, and achieves better computation and communication efficiency than the state of the art perfectly re-randomizable schemes (e.g., Prabhakaran and Rosulek, CRYPTO'07). Next, we revive the Rand-RCCA notion showing new applications where our Rand-RCCA PKE scheme plays a fundamental part: (1) We show how to turn our scheme into a *publicly-verifiable* Rand-RCCA scheme; (2) We construct a malleable NIZK with a (variant of) simulation soundness that allows for re-randomizability; (3) We propose a new UC-secure Verifiable Mix-Net protocol that is secure in the common reference string model. Thanks to the structure-preserving property, all these applications are efficient. Notably, our Mix-Net protocol is the most efficient universally verifiable Mix-Net (without random oracle) where the CRS is an uniformly random string of size independent of the number of senders. The property is of the essence when such protocols are used in large scale.

1 Introduction

Security against chosen ciphertext attacks (CCA) is considered by many the gold standard for public key encryption (PKE). Since the seminal paper of Micali, Rackoff and Sloan [35], the research community has spent a great effort on

⋆ First and second authors are supported by the Spanish Government through the projects Datamantium (ref. RTC-2016-4930-7), SCUM (RTI2018-102043-B-I00), and ERC2018-092822, and by the Madrid Regional Government under project BLOQUES (ref. S2018/TCS-4339).

⋆⋆ The work of the third author is partially supported by Spanish Government through project MTM2016-77213-R.

⋆⋆⋆ The fourth author was supported by a Marie Curie "UPF Fellows" Postdoctoral Grant and by Project RTI2018-102112-B-I00 (AEI/FEDER,UE).

this fundamental topic by both interconnecting different security notions and producing a large body of efficient public encryption schemes.

Challenging the overwhelming agreement that CCA security is **the** right notion of security for PKE, a paper of Canetti, Krawczyk and Nielsen [7] showed that for many use cases a weaker security notion than CCA security is already sufficient. More in details, the paper introduced the notion of Replayable CCA (RCCA) and showed that the notion is sufficient to realize a variant of the public key encryption functionality in the universal composability (UC) model of Canetti [4] where only replay attacks, namely attacks in which the data could be maliciously repeated, can be mounted by the adversary.

In a nutshell, the main fundamental difference between RCCA security and CCA security is that, in a RCCA secure scheme (which is not CCA secure) an adversary is able to maul the challenge ciphertext to obtain new decryptable ciphertexts, the only limitation is that the adversary still cannot break the integrity of the underlying plaintext. To explain this with an example, in a RCCA secure PKE scheme an adversary might append an extra 0 at the end of the ciphertext and still be able to obtain a valid decryption of the mauled ciphertext (to the same plaintext), on the other hand, for a CCA secure PKE, this attack should by definition result into an invalid decryption.

Later, Groth [24] showed that the capability to maul a ciphertext to obtain a new ciphertext which decrypts to the same plaintext should be seen as a feature and not a weakness. In his paper, he introduced the notion of re-randomizable RCCA (Rand-RCCA) PKE, namely a RCCA-secure PKE which comes with an algorithm that re-randomizes the ciphertexts in a way that cannot be linked.

PKE schemes that are both re-randomizable and RCCA-secure have been shown to have several applications, such as: anonymous and secure message transmissions (see Prabhakaran and Rosulek [41]), Mix-Nets (see Faonio and Fiore [16], and Pereira and Rivest [39]), Controlled Functional Encryption (see Naveed *et al.* [38]), and one-round message-transmission protocols with reverse firewalls (see Dodis, Mironov, and Stephens-Davidowitz [13]).

When it comes to constructing these objects, if we look at the literature it is striking to observe that there are extremely efficient constructions of schemes that are only RCCA-secure but not re-randomizable (e.g., Cramer-Shoup [10] or Phan-Pointcheval [40]), or are re-randomizable but only CPA-secure (e.g., ElGamal [14]). In contrast, when the two properties are considered in conjunction, a considerable gap in the efficiency of the schemes seems to arise. More in concrete, the most efficient Rand-RCCA scheme in the standard model of [41] has ciphertexts of 20 groups elements,[4] while, for example, the celebrated Cramer-Shoup PKE [10] has ciphertexts of only 4 groups elements.

In the following paragraphs we state the main contributions of our work.

[4] A recent work of Faonio and Fiore [16] takes this down to 11 group elements at the price of achieving a strictly weaker notion of re-randomizability, in the random oracle model.

Rand-RCCA PKE. Our first contribution is a new structure-preserving[5] Rand-RCCA PKE scheme which significantly narrows the efficiency gap described above. The scheme is secure under the Matrix Diffie-Hellman Assumption (MDDH) in bilinear groups, and for its strongest instantiation, namely, under the Symmetric External Diffie-Hellman Assumption (SXDH), has ciphertexts of 6 groups elements (3 elements in $\mathbb{G}_1$, 2 elements in $\mathbb{G}_2$ and 1 element in $\mathbb{G}_T$).

From a practical perspective, the advantage of a re-randomizable PKE over a standard (non-re-randomizable) PKE strikes when the re-randomizable PKE scheme is part of a larger protocol. To this end, we notice that the structure-preserving property is indeed vital as it allows for modularity and easy integration, which are basic principles for protocol design. However, we can substantiate further our assertion by giving three applications where structure-preserving Rand-RCCA PKE schemes are essential.

Publicly-verifiable Rand-RCCA PKE. Our first application is a publicly-verifiable (pv) Rand-RCCA PKE scheme. A PKE scheme is publicly verifiable when the validity of a ciphertext can be checked without the secret key. This property is for example convenient in the setting of threshold decryption with CCA security [43,5], as the task, roughly speaking, reduces to first publicly check the validity of the ciphertext and then CPA-threshold-decrypt it. Very roughly speaking, we can obtain our pv-Rand-RCCA PKE scheme by appending a Groth-Sahai (GS) NIZK proof [26] of the validity of the ciphertext. We notice that the ciphertext of our Rand-PKE scheme contains[6] an element in $\mathbb{G}_T$. The verification equation does not admit a GS NIZK proof, but only NIWI. We overcome this problem by constructing an additional commitment type for elements in $\mathbb{G}_T$. This gives us a *new* general technique that extends the class of pairing product equations which admit GS NIZK proofs, enlarging therefore the notion of structure preserving. The latter is a contribution of independent interest which might have applications in the field of structure-preserving cryptography in general.

Controlled-Malleable NIZKs. Our second application is a general framework for true-simulation extractable (tSE) and re-randomizable (more generally, controlled-malleable) NIZK systems. The notion of tSE-NIZK was introduced by Dodis *et al.* [12] and found a long series of applications (see for example [20,11,18]). Briefly, the notion assures soundness of the NIZK proofs even when the adversary gets to see simulated NIZK proofs for *true* statements of its choice. In comparison with simulation-extractable (SE) NIZKs (see [42,25]), tSE-NIZKs are considerably more efficient and keep many of the benefits which motivated the introduction of SE-NIZKs[7]. However, if one would like a *controlled malleable*

[5] A scheme is structure preserving if all its public materials, such as messages, public keys, etc. are group elements and the correctness can be verified via pairing-product equations.

[6] In the lingo of structure-preserving cryptography, the scheme is not *strongly* structure preserving.

[7] As an example, tSE-NIZKs are sufficient for the CCA2-secure Naor-Yung PKE of Sahai [42], simulation-sound (SS) NIZKs were introduced in the same paper with exactly this application in mind.

tSE-NIZK, the only available scheme is an SE-NIZK obtained through the general result of Chase *et al.* [8], which is not very efficient. As main result, we scale down the framework of Chase *et al.* to true-simulation extractability, and by using our new Rand-RCCA PKE we construct a new re-randomizable tSE-NIZK scheme. Compared to [8], our scheme can handle a more restricted class of relations and transformations,[8] but our proofs are significantly more efficient. For example, for simple re-randomizable NIZK proofs our tSE NIZKs have an overhead of the order of *tens* more pairing operations for verification, opposed to an overhead of the order of *hundreds* more pairing operations for verification of the simulation-extractable with controlled malleability NIZK systems of [8]. The overhead is computed as the difference with the adaptive sound Groth-Sahai NIZK proof for the same statement.

Mix-Net. Our third application is a universally verifiable and UC-secure Mix-Net based on our pv-Rand-RCCA PKE scheme. Recently, Faonio and Fiore [16] gave a new paradigm to obtain UC-secure verifiable Mix-Net protocols based on Rand-RCCA PKE scheme. Their construction makes use of a non-publicly verifiable Rand-RCCA PKE scheme and obtains a weaker notion of security called *optimistic* (*àla* Golle *et al.* [23]). More in details, the mixing paradigm of [16] is conceptually simple: a mixer receives a list of Rand-RCCA ciphertexts and outputs a randomly permuted list of re-randomized ciphertexts together with a simple NIZK proof that they informally dub "loose shuffling". Such "loose shuffling" proof guarantees that if all the ciphertexts correctly decrypt then the output list is a shuffle of the input one. Hence, in their scheme, cheating can be caught at decryption time, that is after the last mixer returned its list. The problem is that, cheating might be caught too late, thus, their scheme is only optimistic secure. Namely, the scheme is an universal verifiable mix-net optimized to quickly produce a correct output when all the mixers run the protocol correctly. If instead one or more mixers cheat, then no privacy is guaranteed but one can "back up" to a different, slow, mix-net execution.

In this paper, we show that by leveraging the public verifiability of the Rand-RCCA PKE scheme we can obtain a simple design for Mix-Net protocols. In fact, since it is possible to publicly check that a mixer did not invalidate any ciphertext, the proof of loose shuffling turns out to be, indeed, a proof of shuffle.

Interestingly, our use of publicly verifiable ciphertexts come with additional benefits. As mentioned in the paragraph above, our pv-RCCA-PKE scheme can support threshold decryption very easily, and more efficiently than Faonio and Fiore [16]. Finally, our protocol can be fully instantiated in the standard model, whereas the one in [16] rely on non-programmable random oracles.

Most notably, our protocol is the *first efficient universally verifiable Mix-Net in the common random string model*, namely where the common reference string

[8] Yet, our framework is powerful enough for the application of controlled-malleable CCA security of Chase *et al.* Interestingly, we can obtain another pv-Rand-RCCA PKE through their paradigm, although less efficient than our construction. We believe that analyzing what other kinds of CM-CCA notions are supported by our scheme is interesting future work.

is a (small) uniformly random string. In fact, a popular approach to achieve a universally verifiable Mix-Net is to use a NIZK proof of shuffle. However, the most efficient protocols for this task either rely on random oracles to become non-interactive (such as the protocol of Bayer and Groth [1] or Verificatum [46]), or need a structured common reference string (as is the case for the most efficient state-of-the-art NIZK proof of shuffle of Fauzi *et al.* [19]). Furthermore, the common reference string of [19] has size that depends on the number of senders (which in practical scenarios can be huge), whereas our common reference string is made by a number of group elements that is linear in the number of mixers.

Our Mix-Net protocol is proved secure based only on general properties of the pv-Rand-RCCA PKE scheme, and can be instantiated with other schemes in literature (for example with the schemes in [34,8]).

Controlled-Malleable Smooth Projective Hash Functions. At the core of our Rand-RCCA PKE scheme is a new technique that can be seen as a re-randomizable version of smooth projective hash functions (SPHFs) [10]. Given the pervasive use of SPHFs in cryptographic constructions, we believe that our technique may find more applications in the realm of re-randomizable cryptographic primitives. For this reason, we formalize our technique as a primitive called *controlled-malleable SPHF*. Briefly, we define it as an SPHF with tags that allows to re-randomize both instances and tags (inside appropriate spaces), and for which soundness (i.e., smoothness) holds even if the adversary can see a hash value for an invalid instance. We elaborate on this notion in the full version of this paper [17].

Comparison with Related Work. If we consider the state of the art of Rand-RCCA PKE schemes, the most relevant works are the work of Groth, which introduced the notion of Rand-RCCA PKE scheme [24], the aforementioned scheme of Prabhakaran and Rosulek [41], the Rand-RCCA PKE scheme of Chase *et al.* derived from their malleable NIZK systems [8], and two recent works of Libert, Peters and Qian [34] and of Faonio and Fiore [16]. In Table 1 we offer a comparison, in terms of security and functionality properties, of our schemes of Sec. 3 ($\mathcal{PKE}_1$) and Sec. 4 ($\mathcal{PKE}_2$) against previous schemes.

From a technical point of view, the scheme of [41] and our scheme $\mathcal{PKE}_1$, although both based on the Cramer-Shoup paradigm, have little in common. The main differences are: (1) a different design to handle the tags (see next section); (2) a different approach for the re-randomization of the ciphertext. In particular, the Rand-PKE scheme of [41] uses the double-strand technique of Golle *et al.* [22] to re-randomize the ciphertext, while our re-randomization technique, as far as we know, is novel. Furthermore, the scheme of [41] works in two special groups, $\hat{\mathbb{G}}$ and $\tilde{\mathbb{G}}$ that are the subgroups of quadratic residues of $\mathbb{Z}^*_{2q+1}$ and $\mathbb{Z}^*_{4q+3}$ respectively, for a prime q such that $(q, 2q+1, 4q+3)$ is a sequence of primes (a Cunningham Chain of the first kind of length 3).

In Table 2 we compare the efficiency of our new schemes (in the most efficient instantiation with $k = 1$) with the most efficient ones among the Rand-RCCA schemes: the ones in [41] and [16] for the case of secret verifiability, and the scheme in [34] for publicly verifiable Rand-RCCA encryption.

PKE	Group Setting	Assumption	Model	Struc. Pres.	Pub. Ver.	Re-Rand
[24] Groth	–	DDH	GGM			perfect
[41] PR07	Cunn.	DDH	std			perfect
[8,34] CKLM12, LPQ17	Bilin.	SXDH	std	✓	✓	perfect
[16] FF18	–	DDH	NPRO			weak
$\mathcal{PKE}_1$	Bilin.	$\mathcal{D}_k$-MDDH	std	✓*		perfect
$\mathcal{PKE}_2$	Bilin.	$\mathcal{D}_k$-MDDH	std	✓*	✓	perfect

Table 1. Comparison of the properties of a selection of Rand-RCCA-secure PKE schemes. For group setting, – means any group where the assumption holds; Cunn. refers to a pair of groups whose prime orders form a Cunningham chain (see [41]); Bil. stands for bilinear groups. For model, GGM refers to generic group and NPRO refers to non-programmable random oracle. * the structure-preserving property of the two schemes in this paper is not strict, since ciphertexts contain some elements in $\mathbb{G}_T$.

Among the schemes with private verifiability, the most efficient one is that in [16], but its re-randomizability property is weak and the security is in the random oracle model. Among the other two, our scheme $\mathcal{PKE}_1$ is more efficient than that in [41], because the special groups $\tilde{\mathbb{G}}$ required in [41] are large, at least 3072 bits for a security level of 128 bits. Turning to comparing with publicly verifiable schemes, the computational costs for the scheme in [34], in the table, are roughly approximate, because not all the exact computations in the algorithms of the scheme (involving Groth-Sahai proofs) are explicitly described. The size of the ciphertexts reported in [34] is $34|\mathbb{G}_1| + 18|\mathbb{G}_2|$. After personal communication with the authors, we realized that this number is not correct; the correct one is $42|\mathbb{G}_1| + 20|\mathbb{G}_2|$. Our scheme $\mathcal{PKE}_2$ is the most efficient Rand-RCCA scheme with public verifiability up to date: ciphertext size is comparable to that in [34] whereas the computational costs are significantly lower. Even for ciphertext size, ours is comparable to [34] only due to the size of the 4 $\mathbb{G}_T$ elements in our scheme. Besides that, our ciphertexts have many fewer group elements, which is conceptually simpler and, we believe, leaves hope for further improvements. For the two publicly verifiable schemes, the number of pairings required for decryption can be decreased, at the cost of increasing the number of exponentiations, by applying the batching techniques in [28]. The resulting number would be 22P for $\mathcal{PKE}_2$ and something between 40P and 50 P for the scheme in [34].

Technical Overview. We recall that the main technical contributions of this paper are: (1) a new technique for Rand-RCCA PKE scheme (which we also formalize in terms of SPHFs), (2) a new general technique that extends significantly the class of pairing product equations which admits GS NIZK proofs, and (3) a new technique for standard-model UC-secure verifiable Mix-Nets. For space reason, in this technical overview we concentrate on (1).

PKE	Enc ≈ Rand	Dec	$\lvert\mathsf{C}\rvert$	$\lvert\mathsf{pk}\rvert$
PR07	22 $\tilde{E}$	32 $\tilde{E}$	$20\tilde{\mathbb{G}}$	$11\tilde{\mathbb{G}}$
FF18	16 E	18 E	$11\mathbb{G}$	$11\mathbb{G}$
$\mathcal{PKE}_1$	$4E_1{+}5E_2{+}2E_T{+}5\mathsf{P}$	$8E_1{+}4E_2{+}4\mathsf{P}$	$3\mathbb{G}_1{+}2\mathbb{G}_2{+}\mathbb{G}_T$	$7\mathbb{G}_1{+}7\mathbb{G}_2{+}2\mathbb{G}_T$
LPQ17	$79E_1{+}64E_2$	$1E_1{+}142\mathsf{P}$	$42\mathbb{G}_1{+}20\mathbb{G}_2$	$11\mathbb{G}_1{+}16\mathbb{G}_2$
$\mathcal{PKE}_2$	$35E_1{+}31E_2{+}6E_T{+}5\mathsf{P}$	$2E_1{+}46\mathsf{P}$	$12\mathbb{G}_1{+}11\mathbb{G}_2{+}4\mathbb{G}_T$	$8\mathbb{G}_1{+}8\mathbb{G}_2$

Table 2. Efficiency comparison among the best Rand-RCCA-secure PKE schemes; only the last two rows include schemes with public verifiability. For our schemes we consider $k = 1$, so based on SXDH assumption. We use $\tilde{\mathbb{G}}$ for the special groups used in [41], $\mathbb{G}$ for standard DDH groups as considered in [16], and then groups in asymmetric bilinear pairings $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ as considered both in [34] and in this work. Similarly, we denote as $E, \tilde{E}, E_1, E_2, E_T$ the cost of an exponentiation in groups $\mathbb{G}, \tilde{\mathbb{G}}, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$, respectively. Finally, P denotes the cost of computing a bilinear pairing.

A common technique of many CCA-secure PKE schemes in the standard model consists in explicitly labeling each ciphertext produced by the encryption algorithm with a unique tag. Some notable examples of CCA-secure PKE schemes that use tags are the Cramer-Shoup PKE [10], the tag-based PKE of Kiltz [32], and IBE-to-CCA transform of Canetti, Halevi and Katz [6].

Unfortunately, unique tags are not a viable option when designing a re-randomizable PKE scheme. In fact, a ciphertext and its re-randomization would share the same tag, and so they could be trivially linked by an attacker. The main consequence is that many well-known techniques in CCA security cannot be easily exported in the context of Rand-RCCA security. A remarkable exception is the work on Rand-RCCA PKE of Prabhakaran and Rosulek [41]. In this work, the authors managed to reconcile tags and re-randomizability with an ingenious technique: the tag for a new ciphertext is computed as a re-randomizable encoding of the plaintext itself, the tag is then encrypted and attached to the rest of the ciphertext. The decryptor first decrypts the tag and then uses it to check the validity of the payload ciphertext. More in details, the PKE scheme follows the Cramer-Shoup paradigm, therefore their tag (more accurately, a part of their tag) is a $\mathbb{Z}_q$ element (for a properly chosen q). Unfortunately, the restriction on the type of the tags implies that the scheme can be instantiated only in special groups $\mathbb{G}$ of prime order q where the DDH assumption simultaneously holds for both $\mathbb{Z}_q$ and $\mathbb{G}$. Conclusively, the main drawback is a quite large ciphertext size.

We use bilinear-pairing cryptography to overcome the problem of the tags in $\mathbb{Z}_q$. Our starting point is the structure-preserving CCA-PKE of Camenisch *et al.* [3]. Briefly, their PKE scheme is based on the Cramer-Shoup paradigm, with the main twist of performing the validity check in $\mathbb{G}_T$. This trick allows to move the tags from $\mathbb{Z}_q$ to the source group. We give a brief description of the ideas underlying our PKE scheme. We use the implicit notation of Escala *et al.* [15], that uses additive notation for groups and where elements in $\mathbb{G}_i$, are denoted as $[a]_i := a\mathcal{P}_i$ where $\mathcal{P}_i$ is a generator for $\mathbb{G}_i$. The PKE scheme of [3] uses Type-1

pairing groups (where $\mathbb{G}_1 = \mathbb{G}_2$) which are less efficient and secure than Type-3 pairing groups (where no efficient isomorphism from $\mathbb{G}_2$ to $\mathbb{G}_1$ is known to exist). As a first step, we convert their scheme to Type-3 pairing groups; however, for simplicity, in this overview we present the Type-1 version.

Following the blue print of Cramer and Shoup, a ciphertext of the PKE scheme of Camenisch *et al.* consists of three elements: a vector $[\mathbf{c}]_1 \in \mathbb{G}_1^3$ which we call the *instance* (for the DLIN problem described by a matrix $[\mathbf{D}]_1 \in \mathbb{G}_1^{3\times 2}$), an element $[p]_1$ which we call the *payload*, and an element $[\pi]_T$ which we call the *hash*. Together, the instance and the payload form the *tag*, that we denote as $[\mathbf{x}]_1 = [(\mathbf{c}^\top, p)^\top]_1$. The hash is, briefly speaking, a tag-based designated-verifier zero-knowledge proof of the randomness of $[\mathbf{c}]_1$ (namely, that $[\mathbf{c}]_1 = [\mathbf{D}]_1 \cdot \mathbf{r}$). The main difference is that in Cramer-Shoup PKE the tag is computed as a collision-resistant hash of $[\mathbf{x}]_1$, while in our scheme the is the value $[\mathbf{x}]_1$ itself. More in details, the public key material consists of $[\mathbf{D}^*]_1 = [(\mathbf{D}^\top, (\mathbf{a}^\top\mathbf{D})^\top)^\top]_1$, $[\mathbf{f}^\top\mathbf{D}]_T$, and $[\mathbf{F}^\top\mathbf{D}]_1$, where $\mathbf{a}, \mathbf{f} \in \mathbb{Z}_q^3$ and $\mathbf{F} \in \mathbb{Z}_q^{3\times 4}$ are uniformly random, and the encryption algorithm on message $[m]_1$ computes the tag as $[\mathbf{x}]_1 = [\mathbf{D}^*]_1 \cdot \mathbf{r} + [(\mathbf{0}^\top, m)^\top]_1$, and the proof of consistency as $([\mathbf{f}^\top\mathbf{D}]_T + [(\mathbf{F}^\top\mathbf{D})^\top \cdot \mathbf{x}]_T) \cdot \mathbf{r}$, where the addend $[(\mathbf{F}^\top\mathbf{D})^\top \cdot \mathbf{x}]_T$ can be efficiently computed using the pairing. Using the terminology of SPHFs, the hash of the instance $[\mathbf{c}]_1$ and tag $[\mathbf{x}]_1$ is produced using the projective hash algorithm which takes as input the witness $\mathbf{r}$ for $[\mathbf{c}]_1 \in \mathit{span}([\mathbf{D}])$, the tag $[\mathbf{x}]_1$ and the projection key $([\mathbf{f}^\top\mathbf{D}]_T, [\mathbf{F}^\top\mathbf{D}]_1)$. The decryption procedure can re-compute the hash as $e(\mathbf{f}^\top[\mathbf{c}]_1, [1]_1) + e([\mathbf{x}]_1, \mathbf{F}^\top[\mathbf{c}]_1)$, without the knowledge of the witness $\mathbf{r}$ but only using the hash key $(\mathbf{f}, \mathbf{F})$.

To validly re-randomize a ciphertext, the goal would be to compute, using only public information, a new ciphertext where the tag is of the form $[\mathbf{x}'] = [\mathbf{D}^*](\mathbf{r} + \hat{\mathbf{r}}) + [(\mathbf{0}^\top, m)^\top]_1$ (and therefore the instance is of the form $[\mathbf{c}'] = [\mathbf{D}](\mathbf{r} + \hat{\mathbf{r}})$) and the hash is of the form $([\mathbf{f}^\top\mathbf{D}]_T + [(\mathbf{F}^\top\mathbf{D})^\top\mathbf{x}']_T)(\mathbf{r} + \hat{\mathbf{r}})$. However, computing such a re-randomization of the hash is actually infeasible since the scheme is CCA secure.

To overcome this problem, our idea is to reveal enough information about the secret key so as to allow re-randomizability while keeping the scheme secure. To this end, our first observation is to rewrite the equation defining the re-randomized hash considering what we know about $\mathbf{x}'$. Specifically, we use the fact that $(\mathbf{F}^\top\mathbf{D})^\top\mathbf{x}' = (\mathbf{F}^\top\mathbf{D})^\top(\mathbf{x} + \mathbf{D}^*\hat{\mathbf{r}}) = (\mathbf{F}^\top\mathbf{D})^\top\mathbf{x} + (\mathbf{F}^\top\mathbf{D})^\top\ \mathbf{D}^*\hat{\mathbf{r}}$. So the re-randomized hash can be decomposed in three addends as:

$$[\mathbf{f}^\top\mathbf{D} + (\mathbf{F}^\top\mathbf{D})^\top\mathbf{x}]_T(\mathbf{r} + \hat{\mathbf{r}}) \ + \ [(\mathbf{F}^\top\mathbf{D})^\top(\mathbf{D}^*\hat{\mathbf{r}})]_T\hat{\mathbf{r}} \ + \ [(\mathbf{F}^\top\mathbf{D})^\top(\mathbf{D}^*\hat{\mathbf{r}})]_T\mathbf{r}$$

Notice that the first and the second addends can be easily computed knowing the randomizer $\hat{\mathbf{r}}$, the hash $[\pi]_T$ and thanks to the pairing function. So only the third addend is missing.

The second key observation is that we can include the value $[\mathbf{F}\mathbf{D}^*]_1$ in the public key. It is easy to check that, due to the bilinearity of the pairing function, we can compute the missing part as a function of tag $\mathbf{x}$, the randomizer $\hat{\mathbf{r}}$ and this extra piece of information. The third addend can be rewritten as:

$$[(\mathbf{F}^\top\mathbf{D})^\top(\mathbf{D}^*\hat{\mathbf{r}})]_T\mathbf{r} = [\mathbf{D}^\top\mathbf{F}\mathbf{D}^*\hat{\mathbf{r}}]_T\mathbf{r} = [(\mathbf{r}^\top\mathbf{D}^\top)(\mathbf{F}\mathbf{D}^*)\hat{\mathbf{r}}]_T = [\mathbf{x}^\top(\mathbf{F}\mathbf{D}^*\hat{\mathbf{r}})]_T$$

(The last equation can be computed using the pairing $e([\mathbf{x}]_1, [\mathbf{FD}^*]\hat{\mathbf{r}})$.) However, at first look, it is not clear why the scheme should still be secure. To understand it, let us strip away all the computational pieces of the scheme, keeping only the information-theoretic core. In a nutshell, the (one-time simulation) soundness property of the hash boils down to the fact that the function $f(\mathbf{x}) = \mathbf{f} + \mathbf{F} \cdot \mathbf{x}$ is pair-wise independent, meaning that, with knowledge of $f(\mathbf{x})$ one cannot predict $f(\mathbf{x}')$ for $\mathbf{x} \neq \mathbf{x}'$ better than guessing it. However, once we publish the value $\mathbf{FD}^*$ we lose this property. Indeed, given $f(\mathbf{x})$ and $\mathbf{FD}^*$, now we can easily compute the function f over all the points in the affine space $\{\mathbf{x}' \mid \mathbf{x}' = \mathbf{x} + \mathbf{D}^*\mathbf{r}, \mathbf{r} \in \mathbb{Z}_q^2\}$. On one hand, this is good as it allows us to re-randomize. On the other hand, we should prove that one cannot do more than this honest manipulation. Our main technical lemma shows that for any $\mathbf{x}'$ outside this affine space we still have pair-wise independence, i.e., the value $f(\mathbf{x}')$ is unpredictable.

2 Preliminaries and Definitions

A function is negligible in λ if it vanishes faster than the inverse of any polynomial in λ, we write $f(\lambda) \in \mathtt{negl}(\lambda)$ when f is negligible in λ. An asymmetric bilinear group is a tuple $\mathcal{G}$ is a tuple $(q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \mathcal{P}_1, \mathcal{P}_2)$, where $\mathbb{G}_1, \mathbb{G}_2$ and $\mathbb{G}_T$ are groups of prime order q, the elements $\mathcal{P}_1, \mathcal{P}_2$ are generators of $\mathbb{G}_1, \mathbb{G}_2$ respectively, $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an efficiently computable, non-degenerate bilinear map, and there is no efficiently computable isomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$. Let GGen be some probabilistic polynomial time algorithm which on input 1^λ, where λ is the security parameter returns a description of an asymmetric bilinear group $\mathcal{G}$. Elements in $\mathbb{G}_i$, are denoted in implicit notation as $[a]_i := a\mathcal{P}_i$, where $i \in \{1, 2, T\}$ and $\mathcal{P}_T := e(\mathcal{P}_1, \mathcal{P}_2)$. Every element in $\mathbb{G}_i$ can be written as $[a]_i$ for some $a \in \mathbb{Z}_q$, but note that given $[a]_i$, $a \in \mathbb{Z}_q$ is in general hard to compute (discrete logarithm problem). Given $a, b \in \mathbb{Z}_q$ we distinguish between $[ab]_i$, namely the group element whose discrete logarithm base $\mathcal{P}_i$ is ab, and $[a]_i \cdot b$, namely the execution of the multiplication of $[a]_i$ and b, and $[a]_1 \cdot [b]_2 = [a \cdot b]_T$, namely the execution of a pairing between $[a]_1$ and $[b]_2$. Vectors and matrices are denoted in boldface. We extend the pairing operation to vectors and matrices as $e([\mathbf{A}]_1, [\mathbf{B}]_2) = [\mathbf{A}^\top \cdot \mathbf{B}]_T$. $span(\mathbf{A})$ denotes the linear span of the columns of $\mathbf{A}$.

Let ℓ, k be positive integers. We call $\mathcal{D}_{\ell,k}$ a matrix distribution if it outputs (in PPT time, with overwhelming probability) matrices in $\mathbb{Z}_q^{\ell \times k}$. We define $\mathcal{D}_k := \mathcal{D}_{k+1,k}$. Our results will be proven secure under the following decisional assumption in $\mathbb{G}_\gamma$, for some $\gamma \in \{1, 2\}$.

Definition 1 (Matrix Decisional Diffie-Hellman Assumption in $\mathbb{G}_\gamma$, [15]). *The $\mathcal{D}_{\ell,k}$-MDDH assumption holds if for all non-uniform PPT adversaries A,*

$$|\Pr[\mathsf{A}(\mathcal{G}, [\mathbf{A}]_\gamma, [\mathbf{Aw}]_\gamma) = 1] - \Pr[\mathsf{A}(\mathcal{G}, [\mathbf{A}]_\gamma, [\mathbf{z}]_\gamma) = 1]| \in \mathtt{negl}(\lambda),$$

where the probability is taken over $\mathcal{G} = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \mathcal{P}_1, \mathcal{P}_2) \leftarrow \mathsf{GGen}(1^\lambda)$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}, \mathbf{w} \leftarrow \mathbb{Z}_q^k, [\mathbf{z}]_\gamma \leftarrow \mathbb{G}_\gamma^\ell$ and the coin tosses of adversary A.

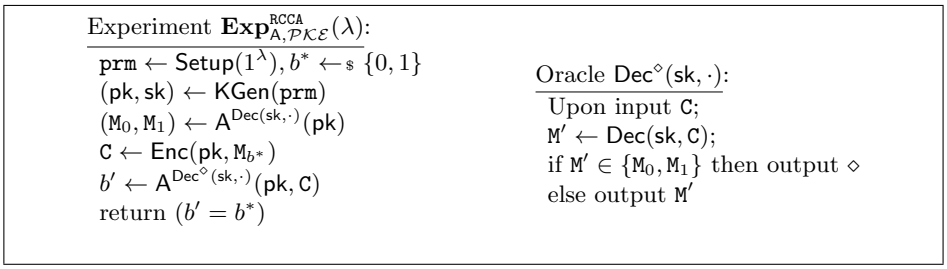

Fig. 1: The RCCA Security Experiment.

2.1 Re-randomizable RCCA PKE

A re-randomizable PKE (Rand-PKE) scheme $\mathcal{PKE}$ is a tuple of five algorithms: (I) $\mathsf{Setup}(1^\lambda)$ upon input the security parameter λ produces public parameters $\mathtt{prm}$, which include the description of the message and ciphertext space $\mathcal{M}, \mathcal{C}$. (II) $\mathsf{KGen}(\mathtt{prm})$ upon input the parameters $\mathtt{prm}$, outputs a key pair $(\mathsf{pk},\mathsf{sk})$; (III) $\mathsf{Enc}(\mathsf{pk},\mathtt{M})$ upon inputs a public key pk and a message $\mathtt{M} \in \mathcal{M}$, outputs a ciphertext $\mathtt{C} \in \mathcal{C}$; (IV) $\mathsf{Dec}(\mathsf{pk},\mathsf{sk},\mathtt{C})$ upon input the secret key sk and a ciphertext $\mathtt{C}$, outputs a message $\mathtt{M} \in \mathcal{M}$ or an error symbol $\perp$; (V) $\mathsf{Rand}(\mathsf{pk},\mathtt{C})$ upon inputs a public key pk and a ciphertext $\mathtt{C}$, outputs another ciphertext $\mathtt{C}'$.

The RCCA security notion is formalized with a security experiment similar to the CCA security one except that in RCCA the decryption oracle (called the guarded decryption oracle) can be queried on any ciphertext and, when decryption leads to one of the challenge messages $\mathtt{M}_0, \mathtt{M}_1$, it answers with a special symbol $\diamond$ (meaning "same").

Definition 2 (Replayable CCA Security, [7]). *Consider the experiment* $\mathbf{Exp}^{\mathtt{RCCA}}$ *in Fig. 1, with parameters* λ*, an adversary* A*, and a PKE scheme* $\mathcal{PKE}$*. We say that* $\mathcal{PKE}$ *is* indistinguishable secure under replayable chosen-ciphertext attacks *(RCCA-secure) for any PPT adversary* A*:*

$$\mathbf{Adv}^{\mathtt{RCCA}}_{\mathsf{A},\mathcal{PKE}}(\lambda) := \left| \Pr\left[\mathbf{Exp}^{\mathtt{RCCA}}_{\mathsf{A},\mathcal{PKE}}(\lambda) = 1\right] - \frac{1}{2} \right| \in \mathtt{negl}(\lambda).$$

We formally define perfect re-randomizability in the full version of this paper [17]. Here we give a simplified description of the notion. The notion of perfect re-randomizability consists of three conditions: (i) the re-randomization of a valid ciphertext and a fresh ciphertext (for the same message) are equivalently distributed; (ii) the re-randomization procedure maintains correctness, meaning the randomized ciphertext and the original decrypt to the same value, in particular, invalid ciphertexts keep being invalid; (iii) it is hard to find a valid ciphertext that is not in the support of the encryption scheme. The last condition, coupled with the first one, implies that for any (possibly malicious) ciphertext that decrypts correctly the distribution of the re-randomized ciphertext and a fresh ciphertext are statistically close. This stronger property is particularly useful in applications, like our Mix-Net of Sec. 6, where we need to re-randomize adversarially chosen ciphertexts.

$\underline{\mathbf{Exp}^{\mathtt{der\text{-}priv}}_{\mathsf{A},\mathcal{NIZK}}}$:
$\mathtt{prm}_G \leftarrow\!\!\$\ \mathsf{Setup}_G(1^\lambda)$; $b^* \leftarrow\!\!\$\ \{0,1\}$;
$(\mathsf{crs}, tp_e, tp_s) \leftarrow \mathsf{Init}(\mathtt{prm}_G)$;
$(x, w, \pi, T) \leftarrow \mathsf{A}(\mathsf{crs}, tp_s)$; Assert $\mathsf{V}(\mathsf{crs}, x, \pi) = 1$;
If $b^* = 0$ then $\pi' \leftarrow\!\!\$\ \mathsf{P}(\mathsf{crs}, T_x(x), T_w(w))$;
else $\pi' \leftarrow\!\!\$\ \mathsf{ZKEval}(\mathsf{crs}, \pi, T)$;
$b \leftarrow \mathsf{A}(\pi')$;
Output $b = b^*$.

Fig. 2: The security experiments for the derivation privacy.

Definition 3 (Public Verifiability). *$\mathcal{PKE} = (\mathsf{Setup}, \mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Rand})$ is a public key scheme with publicly verifiable ciphertexts if there is a deterministic algorithm Ver which, on input $(\mathsf{pk}, \mathtt{C})$ outputs an error symbol $\perp$ whenever $\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}, \mathtt{C}) = \perp$, else it outputs valid.*

2.2 Malleable NIZKs

Recall that a non-interactive zero-knowledge proof system (NIZK) is a tuple $(\mathsf{Init}, \mathsf{P}, \mathsf{V})$ of PPT algorithms. Briefly, the algorithm Init upon input group parameters outputs a common reference string and, possibly, trapdoor information (we will consider algorithms that outputs a trapdoor tp_e for extraction and a trapdoor tp_s for simulation). We use the definitional framework of Chase *et al.* [8] for malleable proof systems. For simplicity of the exposition we consider only the unary case for transformations (see the aforementioned paper for more details). Let $T = (T_x, T_r)$ be a pair of efficiently computable functions, that we refer as a *transformation*.

Definition 4 (Admissible transformations, [8]). *An efficient relation $\mathcal{R}$ is* closed *under a transformation $T = (T_x, T_w)$ if for any $(x, w) \in \mathcal{R}$ the pair $(T_x(x), T_w(w)) \in \mathcal{R}$. If $\mathcal{R}$ is closed under T then we say that T is an* admissible *for $\mathcal{R}$. Let $\mathcal{T}$ be a set of transformations, if for every $T \in \mathcal{T}$, T is admissible for $\mathcal{R}$, then $\mathcal{T}$ is* allowable *set of transformations.*

Definition 5 (Malleable NIZK, [8]). *Let $\mathcal{NIZK} = (\mathsf{Init}, \mathsf{P}, \mathsf{V})$ be a NIZK for a relation $\mathcal{R}$. Let $\mathcal{T}$ be an allowable set of transformations for $\mathcal{R}$. The proof system is* malleable with respect to $\mathcal{T}$ *if there exists an PPT algorithm ZKEval that on input $(\mathsf{crs}, T, (x, \pi))$, where $T \in \mathcal{T}$ and $\mathsf{V}(\mathsf{crs}, x, \pi) = 1$ outputs a valid proof π' for the statement $x' = T_x(x)$.*

We would like the property that two NIZK proofs where one is derived from the other cannot be linked. This is formalized with the notion of *derivation privacy*.

Definition 6. *Let $\mathcal{NIZK} = (\mathsf{Init}, \mathsf{P}, \mathsf{V}, \mathsf{ZKEval})$ be a malleable NIZK argument for a relation $\mathcal{R}$ and an allowable set of transformations $\mathcal{T}$. We say that $\mathcal{NIZK}$ is* derivation private *if for any PPT adversary A we have that*

$$\mathbf{Adv}^{\mathtt{der\text{-}priv}}_{\mathsf{A},\mathcal{NIZK}}(\lambda) := \left| \Pr\left[\mathbf{Exp}^{\mathtt{der\text{-}priv}}_{\mathsf{A},\mathcal{NIZK}}(1^\lambda) = 1\right] - \tfrac{1}{2} \right| \in \mathtt{negl}(\lambda)$$

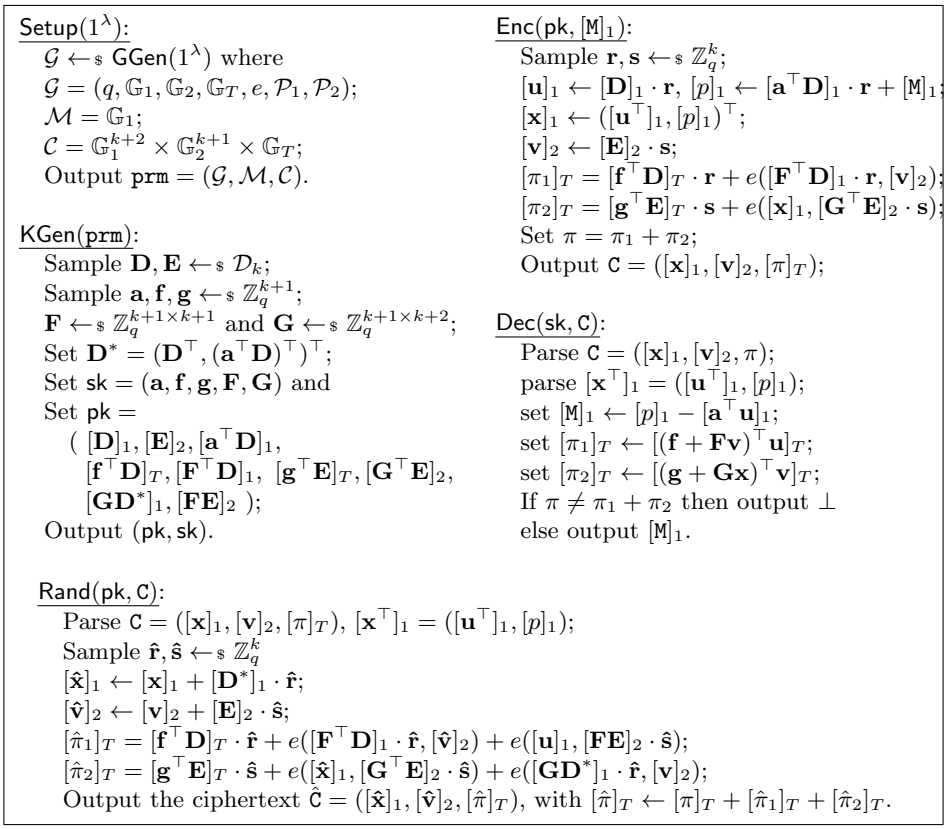

Fig. 3: Our Rand-RCCA encryption scheme $\mathcal{PKE}_1$ based on the $\mathcal{D}_k$-MDDH assumption for $k \in \mathbb{N}^*$.

where $\mathbf{Exp}^{\mathtt{der\text{-}priv}}$ *is the game described in Fig. 2. Moreover we say that* $\mathcal{NIZK}$ *is* perfectly derivation private *(resp.* statistically derivation private*) when for any (possibly unbounded) adversary the advantage above is 0 (resp. negligible).*

Finally, we assume that an adversary cannot find a verifying proof for a valid statement which is not in the support of the proof generated by the proving algorithm. We notice that this property is true for both GS proof systems and for quasi-adaptive proof system of Kiltz and Wee [33]. In particular, for GS proofs, for any commitment to the witness, the prover generates a proof that is uniformly distributed over the set of all the possible valid proofs. On the other hand, the proofs of Kiltz and Wee are unique, therefore the condition is trivially true.

3 Our Rand-RCCA PKE scheme

We present our scheme in Fig. 3. We refer to the introduction for an informal exposition of our techniques. We notice that the check in the decryption proce-

dure can be efficiently computed using the pairing function and the knowledge of $\mathbf{f}, \mathbf{F}, \mathbf{g}, \mathbf{G}$. In the next paragraphs we first show correctness of the scheme, secondly, we give an information-theoretic lemma which is the basic core of the security of our PKE scheme, then we proceed with the RCCA-security of the scheme.

Correctness of decryption. For correctness of decryption, it is easy to see that for a honestly generated ciphertext $([\mathbf{x}]_1, [\mathbf{v}]_2, [\pi]_T) \leftarrow_\$ \mathsf{Enc}(\mathsf{pk}, [\mathtt{M}]_1)$, the first line of decryption $[p]_1 - [\mathbf{a}^\top \mathbf{u}]_1$ yields $[\mathtt{M}]_1$. Hence, we are left with showing that the test $[\pi]_T = [(\mathbf{f} + \mathbf{F}\mathbf{v})^\top \mathbf{u}]_T + [(\mathbf{g} + \mathbf{G}\mathbf{x})^\top \mathbf{v}]_T$ is satisfied:

$$\begin{aligned} \pi = \pi_1 + \pi_2 &= (\mathbf{f}^\top \mathbf{D})\mathbf{r} + (\mathbf{F}^\top \mathbf{D}\mathbf{r})^\top \mathbf{v} + (\mathbf{g}^\top \mathbf{E})\mathbf{s} + \mathbf{x}^\top (\mathbf{G}^\top \mathbf{E})\mathbf{s} \\ &= (\mathbf{f} + \mathbf{F}\mathbf{v})^\top \mathbf{u} + (\mathbf{g} + \mathbf{G}\mathbf{x})^\top \mathbf{v} \end{aligned} \tag{1}$$

Before analyzing the perfect re-randomizability and RCCA security of the scheme we state and prove a powerful information-theoretic lemma. Very informally speaking, the lemma proves that the smooth projective hash proof system at the core of our scheme remains sound even if the adversary gets to see a proof for an instance of its choice. As we want to allow for re-randomization, we relax the notion of soundness by requiring that the instance forged by the adversary does not lie in the set of possible re-randomizations of its query.

Lemma 1. *Let k be a positive integer. For any matrices $\mathbf{D} \in \mathbb{Z}_q^{k+1\times k}, \mathbf{E} \in \mathbb{Z}_q^{k+1\times k}$ and any (possibly unbounded) adversary A:*

$$\Pr\left[\begin{array}{c} \mathbf{u} \notin \mathit{span}(\mathbf{D}) \\ (\mathbf{v} - \mathbf{v}^*) \notin \mathit{span}(\mathbf{E}) \\ z = (\mathbf{f} + \mathbf{F}\mathbf{v})^\top \mathbf{u} \end{array} \middle| \begin{array}{l} \mathbf{f} \leftarrow_\$ \mathbb{Z}_q^{k+1}, \mathbf{F} \leftarrow_\$ \mathbb{Z}_q^{k+1\times k+1}; \\ (z, \mathbf{u}, \mathbf{v}) \leftarrow_\$ \mathsf{A}^{\mathcal{O}(\cdot)}(\mathbf{D}, \mathbf{E}, \mathbf{D}^\top \mathbf{f}, \mathbf{D}^\top \mathbf{F}, \mathbf{F}\mathbf{E}) \end{array}\right] \leq 1/q,$$

where the adversary outputs a single query $\mathbf{v}^*$ to $\mathcal{O}(\cdot)$ which returns $\mathbf{f} + \mathbf{F} \cdot \mathbf{v}^*$.

Proof. Let $\mathbf{K} = \big(\mathbf{f}, \mathbf{F}\big) \in \mathbb{Z}_q^{k+1\times k+2}$. We can rewrite the information that the adversary sees about $\mathbf{f}, \mathbf{F}$ in matrix form:

$$\big(\mathbf{D}, \mathbf{E}, \mathbf{D}^\top \mathbf{f}, \mathbf{D}^\top \mathbf{F}, \mathbf{F}\mathbf{E}, \mathbf{f} + \mathbf{F} \cdot \mathbf{v}^*\big) = \left(\mathbf{D}, \mathbf{E}, \mathbf{D}^\top \mathbf{K}, \mathbf{K}\begin{pmatrix}\mathbf{0}\\ \mathbf{E}\end{pmatrix}, \mathbf{K}\begin{pmatrix}1\\ \mathbf{v}^*\end{pmatrix}\right).$$

We now have to argue that $z = \mathbf{u}^\top \mathbf{K}\begin{pmatrix}1\\ \mathbf{v}\end{pmatrix}$ is independent of the adversary's view when $\mathbf{u} \notin \mathit{span}(\mathbf{D})$ and $(\mathbf{v} - \mathbf{v}^*) \notin \mathit{span}(\mathbf{E})$. Without loss of generality we assume the matrices $\mathbf{D}, \mathbf{E}$ to be full rank. Otherwise this means there is a redundancy in the information provided to the adversary and this clearly does not give him more chances of being successful. Define the following matrices:

$$\tilde{\mathbf{D}} = \big(\mathbf{D}, \mathbf{u}\big) \in \mathbb{Z}_q^{k+1\times k+1}, \qquad \tilde{\mathbf{E}} = \begin{pmatrix}\mathbf{0}, & 1, & 1\\ \mathbf{E}, & \mathbf{v}^*, & \mathbf{v}\end{pmatrix} \in \mathbb{Z}_q^{k+2\times k+2}.$$

By the condition that $\mathbf{u} \notin span(\mathbf{D})$ and $(\mathbf{v} - \mathbf{v}^*) \notin span(\mathbf{E})$, $\tilde{\mathbf{D}}$ and $\tilde{\mathbf{E}}$ are invertible matrices.

Let us consider the matrix $\mathbf{Z} = \tilde{\mathbf{D}}^\top \mathbf{K} \tilde{\mathbf{E}} \in \mathbb{Z}_q^{k+1 \times k+2}$ and the information that the adversary has on this matrix. Note that for $z_{k+1,k+2}$, namely the term in last row and last column of $\mathbf{Z}$, the following holds:

$$z_{k+1,k+2} = \mathbf{u}^\top \mathbf{K} \begin{pmatrix} 1 \\ \mathbf{v} \end{pmatrix} = z.$$

Since the view of the adversary contains invertible matrix $\tilde{\mathbf{E}}$, knowledge of $\mathbf{D}^\top \mathbf{K}$ (in the view of the adversary) is equivalent to knowledge of $\mathbf{D}^\top \mathbf{K} \tilde{\mathbf{E}}$, which are the first k rows of $\mathbf{Z}$.

Similarly, let $\hat{\mathbf{E}}$ be the first $k+1$ columns of $\tilde{\mathbf{E}}$, since $\tilde{\mathbf{D}}$ is invertible and is known by the adversary, knowledge of $\mathbf{K}\hat{\mathbf{E}}$ (in the view of the adversary) is equivalent to knowledge of $\tilde{\mathbf{D}}^\top \mathbf{K}\hat{\mathbf{E}}$, the first $k+1$ columns of $\mathbf{Z}$. Therefore, the view of the adversary includes all the matrix $\mathbf{Z}$ except for $z_{k+1 \times k+2}$.

On the other hand, since $\tilde{\mathbf{D}}$ and $\tilde{\mathbf{E}}$ are invertible matrices, if we see $\mathbf{Z} = \tilde{\mathbf{D}}^\top \mathbf{K} \tilde{\mathbf{E}} \in \mathbb{Z}_q^{k+1 \times k+2}$ as a system of equations with unknown $\mathbf{K}$, there exists a unique solution $\mathbf{K}$ for any choice of $\mathbf{Z}$, namely, $\mathbf{K} = (\tilde{\mathbf{D}}^\top)^{-1} \mathbf{Z} \tilde{\mathbf{E}}^{-1}$.

Therefore, from the point of view of the adversary, every value of $z_{k+1 \times k+2} \in \mathbb{Z}_q$ is equally likely, since $\mathbf{K} \leftarrow_\$ \mathbb{Z}_q^{k+1 \times k+2}$ is sampled uniformly at random. This concludes the proof.

Security. For space reason we prove perfect re-randomizability in the full version of this paper [17]. We prove that the security of the scheme reduces to the $\mathcal{D}_k$-MDDH assumption. Below we state the main theorem:

Theorem 1. *For any matrix distribution $\mathcal{D}_k$ such that the $\mathcal{D}_k$-MDDH assumption holds for the groups $\mathbb{G}_1$ and $\mathbb{G}_2$ generated by GGen, the Rand-PKE scheme $\mathcal{PKE}_1$ described above is RCCA-secure.*

Proof. We start by describing a sequence of hybrid games. For readability purposes, we underline the main differences between each consecutive hybrid. In hybrids $\mathbf{H}_0$ and from $\mathbf{H}_3$ until $\mathbf{H}_7$ we progressively change the way the decryption procedure works. In the description of the games, the changes correpond to the underlined formulae. We summarize the main changes in Fig. 4.

Hybrid $\mathbf{H}_0$. This hybrid experiment is equivalent to the RCCA experiment described in Fig. 1 but the oracle $\mathsf{Dec}^\diamond$ is instantiated with a slightly different decryption procedure. Decryption proceeds exactly as in the description of the PKE scheme, except that, before setting each variable $\mathtt{M}, \pi_1, \pi_2$ it additionally checks if the variable was not set already. For future reference, we label these commands as the decryption rule (*).

Notice that, in this hybrid, this change is merely syntactical, as at each invocation of the decryption procedure all the three variables are unset. The hybrid $\mathbf{H}_0$ is equivalent to the experiment $\mathbf{Exp}^{\mathtt{RCCA}}_{\mathsf{A},\mathcal{PKE}}(\lambda)$ of Fig. 1.

Hybrid $\mathbf{H}_1$. The hybrid $\mathbf{H}_1$ is the same as $\mathbf{H}_0$ but it computes the challenge ciphertext $\mathtt{C}^* = ([\mathbf{x}^*]_1, [\mathbf{v}^*]_2, [\pi^*]_T)$ by using the secret key. Let $\mathbf{x}^*$ be $((\mathbf{u}^*)^\top, p^*)^\top$ and $\pi^* = \pi_1^* + \pi_2^*$.

$$[\mathbf{u}^*]_1 \leftarrow [\mathbf{D}]_1 \cdot \mathbf{r}^*, \quad \underline{[p^*]_1 \leftarrow \mathbf{a}^\top \cdot [\mathbf{u}^*]_1 + [\mathtt{M}_{b^*}]_1} \text{ where } \mathbf{r}^* \leftarrow\$ \mathbb{Z}_q^k$$
$$[\mathbf{v}^*]_2 \leftarrow [\mathbf{E}]_2 \cdot \mathbf{s}^* \text{ where } \mathbf{s}^* \leftarrow\$ \mathbb{Z}_q^k$$
$$[\pi_1^*]_T \leftarrow \underline{e([\mathbf{u}^*]_1, [\mathbf{f}]_2 + \mathbf{F} \cdot [\mathbf{v}^*]_2)}, \quad \underline{[\pi_2^*]_T \leftarrow e([\mathbf{g}]_1 + \mathbf{G} \cdot [\mathbf{x}^*]_1, [\mathbf{v}^*]_2)}.$$

Notice that $[\pi_1^*]_T$ and $[\pi_2^*]_T$ can be efficiently computed using the secret key and the pairing function. The only differences introduced are in the way we compute $[p^*]_1$ and $[\pi^*]_T$. However, notice that such differences are only syntactical, as, by the correctness of the scheme, we compute exactly the same values the hybrid $\mathbf{H}_0$ would compute.

Hybrid $\mathbf{H}_2$. The hybrid $\mathbf{H}_2$ is the same as $\mathbf{H}_1$ but the challenger, upon challenge messages $[\mathtt{M}_0]_1, [\mathtt{M}_1]_1 \in \mathbb{G}_1$, computes the challenge ciphertext $\mathtt{C}^* = ([\mathbf{x}^*]_1, [\mathbf{v}^*]_2, [\pi^*]_T)$ where $\mathbf{x}^*$ is $((\mathbf{u}^*)^\top, p^*)^\top$ by sampling :

$$\underline{\mathbf{u}^* \leftarrow\$ \mathbb{Z}_q^{k+1} \setminus span(\mathbf{D})} \qquad \underline{\mathbf{v}^* \leftarrow\$ \mathbb{Z}_q^{k+1} \setminus span(\mathbf{E})}.$$

The hybrids $\mathbf{H}_1$ and $\mathbf{H}_2$ are computationally indistinguishable. This follows by applying the $\mathcal{D}_k$-MDDH Assumption on $[\mathbf{D}, \mathbf{u}^*]_1$ in $\mathbb{G}_1$ and $[\mathbf{E}, \mathbf{v}^*]_2$ in $\mathbb{G}_2$, respectively, and then a standard statistical argument to show that sampling $\mathbf{u}^*$ uniformly at random in $\mathbb{Z}_q^{k+1}$ is statistically close to sampling it at random in $\mathbb{Z}_q^{k+1} \setminus span(\mathbf{D})$. The reduction is straightforward and is omitted.

From now on, we prove that each pair of consecutive hybrids is statistically close. In particular, this means that the hybrids (and in principle also the adversary) are allowed to run in unbounded time.

Hybrid $\mathbf{H}_3$. The hybrid $\mathbf{H}_3$ is the same as $\mathbf{H}_2$ but adds the following decryption rules that upon input a ciphertext $([\mathbf{u}]_1, [p]_1, [\mathbf{v}]_2, [\pi]_T)$:

(i) If $\mathbf{u} = \mathbf{D}\mathbf{r}$ for some $\mathbf{r} \in \mathbb{Z}_q^k$, then compute
$$\underline{[\pi_1]_T \leftarrow [(\mathbf{f}^\top \mathbf{D} + \mathbf{v}^\top \mathbf{F}^\top \mathbf{D})]_T \cdot \mathbf{r}} \qquad \underline{[\mathtt{M}]_1 \leftarrow [p]_1 - [\mathbf{a}^\top \mathbf{D}]_1 \cdot \mathbf{r}}$$

(ii) If $\mathbf{v} = \mathbf{E}\mathbf{s}$ for some $\mathbf{s} \in \mathbb{Z}_q^k$, letting $\mathbf{x} = (\mathbf{u}^\top, p)^\top$, then compute:
$$\underline{[\pi_2]_T \leftarrow [(\mathbf{g}^\top \mathbf{E} + \mathbf{x}^\top \mathbf{G}^\top \mathbf{E})]_T \cdot \mathbf{s}}$$

Specifically, in the first rule the decryption of $\mathtt{M}$ and π_1 are computed using the public key components $[\mathbf{a}^\top \mathbf{D}]_1, [\mathbf{f}^\top \mathbf{D}]_T$ and $[\mathbf{F}^\top \mathbf{D}]_1$ instead of the secret key components $\mathbf{a}, \mathbf{f}, \mathbf{F}$ for all the ciphertexts with $\mathbf{u} \in span(\mathbf{D})$. Recall that this strategy is not efficient, but it is possible because the simulator does not need to run in polynomial time (since we want to argue the games are statistically close). If $\mathbf{v} = \mathbf{E}\mathbf{s}$, then by the second rule, the hybrid computes the proof π_2 using only the components $[\mathbf{g}^\top \mathbf{E}]_T$ and $[\mathbf{G}^\top \mathbf{E}]_2$ of the public key.

We notice that, again by correctness of the PKE scheme, the computation of π_1, π_2 and $\mathtt{M}$ in the hybrids $\mathbf{H}_3$ and $\mathbf{H}_2$ is equivalent. In particular, let π_1' be

Procedure $\mathsf{Dec}^*(\mathsf{sk}, \mathtt{C})$:
Parse $\mathtt{C} = ([\mathbf{x}]_1, [\mathbf{v}]_2, [\pi]_T)$ and $[\mathbf{x}^\top]_1 = ([\mathbf{u}^\top]_1, [p]_1)$
(i) If $\mathbf{u} \in span(\mathbf{D})$, let $\mathbf{u} = \mathbf{D}\mathbf{r}$ then
$[\mathtt{M}]_1 \leftarrow [p - \mathbf{a}^\top \mathbf{D}\mathbf{r}]_1$;
$[\pi_1]_T \leftarrow [(\mathbf{f}^\top \mathbf{D} + \mathbf{v}^\top \mathbf{F}^\top \mathbf{D})\mathbf{r}]_T$;
(ii) If $\mathbf{v} \in span(\mathbf{E})$, let $\mathbf{v} = \mathbf{E}\mathbf{s}$ then
$[\pi_2]_T \leftarrow [(\mathbf{g}_0^\top \mathbf{E} + \mathbf{x}^\top \mathbf{G}^\top \mathbf{E})\mathbf{s}]_T$;
(iii) If $\mathbf{u} \notin span(\mathbf{D})$ and $(\mathbf{v} - \mathbf{v}^* \notin span(\mathbf{E})$ or $\mathbf{v}^*$ unset) then output $\perp$.
(iv) If $\mathbf{v} \notin span(\mathbf{E})$ and $(\mathbf{x} - \mathbf{x}^* \notin span(\mathbf{D}^*)$ or $\mathbf{u}^*$ unset) then output $\perp$.
(v) If $\mathbf{x} - \mathbf{x}^* \in span(\mathbf{D}^*)$ and $\mathbf{v} - \mathbf{v}^* \in span(\mathbf{E})$ then
$\mathtt{M} \leftarrow \diamond$;
$[\pi_1]_T \leftarrow [\pi^*]_T + [(\mathbf{f}^\top \mathbf{D} + \tilde{\mathbf{v}}^\top \mathbf{F}^\top \mathbf{D})\tilde{\mathbf{x}}]_T$
$[\pi_2]_T \leftarrow [(\mathbf{g}_0^\top \mathbf{E} + \tilde{\mathbf{x}}^\top \mathbf{G}^\top \mathbf{E})\tilde{\mathbf{x}}]_T$
(*) If $[\mathtt{M}]_1$ is unset set $[\mathtt{M}]_1 \leftarrow [p]_1 - \mathbf{a}^\top [\mathbf{u}]$;
(*) If $[\pi_1]_T$ is unset set $[\pi_1]_T \leftarrow [(\mathbf{f} + \mathbf{F}\mathbf{v})^\top \mathbf{u}]_T$;
(*) If $[\pi_2]_T$ is unset set $[\pi_2]_T \leftarrow [(\mathbf{g}_0 + \mathbf{G}\mathbf{x})^\top \mathbf{v}]_T$;
If $[\pi]_T = [\pi_1]_T + [\pi_2]_T$ output $\mathtt{M}$ else $\perp$.

Fig. 4: The decryption procedure in the hybrids experiment. The decryption procedure of the hybrid $\mathbf{H}_0$ executes only the rules (*) and the last decryption check. The decryption procedure of the hybrid $\mathbf{H}_3$ additionally executes (i) and (ii). The decryption procedure of the hybrid $\mathbf{H}_4$ additionally executes (iii). The decryption procedure of the hybrid $\mathbf{H}_5$ additionally executes (iv). The decryption procedure of the hybrid $\mathbf{H}_6$ additionally executes (v). The decryption procedure of the hybrid $\mathbf{H}_7$ stops to execute the rules (*).

the proof as computed in $\mathbf{H}_2$, then $[\pi_1']_T = [(\mathbf{f} + \mathbf{F}\mathbf{v})^\top \mathbf{u}]_T = [(\mathbf{f} + \mathbf{F}\mathbf{v})^\top \mathbf{D}\mathbf{r}]_T = [(\mathbf{f}^\top \mathbf{D} + \mathbf{v}^\top \mathbf{F}^\top \mathbf{D})]_T \cdot \mathbf{r} = [\pi_1]_T$. (An equivalent derivation holds for π_2 and $\mathtt{M}$.) The difference is then only syntactical.

Hybrid $\mathbf{H}_4$. The hybrid $\mathbf{H}_4$ is the same as $\mathbf{H}_3$ but adds the following decryption rule, on input a ciphertext $\mathtt{C} = ([\mathbf{u}]_1, [p]_1, [\mathbf{v}]_2, [\pi]_T)$:

(iii) If $\mathbf{u} \notin span(\mathbf{D})$ and $(\mathbf{v} - \mathbf{v}^* \notin span(\mathbf{E})$ or $\mathbf{v}^*$ is unset) then output $\perp$.

Recall that the challenge ciphertext is $\mathtt{C}^* = ([\mathbf{u}^*]_1, [p^*]_1, [\mathbf{v}^*]_2, [\pi]_T)$. Notice that we check either if $\mathbf{v} - \mathbf{v}^* \notin span(\mathbf{E})$ or $\mathbf{v}^*$ is unset. We do so to handle simultaneously the decryption queries before and after the challenge ciphertext is computed. In particular, before the challenge ciphertext is computed the decryption rule simply checks if $\mathbf{u} \notin span(\mathbf{D})$ (as in the classical Cramer-Shoup proof strategy).

We show in Lemma 3 that $\mathbf{H}_4$ is statistically close to $\mathbf{H}_3$. Here we continue describing the hybrid games.

Hybrid $\mathbf{H}_5$. The hybrid $\mathbf{H}_5$ is the same as $\mathbf{H}_4$ but adds the following decryption rule, on input a ciphertext $\mathtt{C} = ([\mathbf{x}]_1, [\mathbf{v}]_2, [\pi]_T)$:

(iv) If $\mathbf{v} \notin span(\mathbf{E})$ and $(\mathbf{x} - \mathbf{x}^* \notin span(\mathbf{D}^*)$ or $\mathbf{x}^*$ is unset) then output $\perp$.

We show that $\mathbf{H}_5$ is statistically close to $\mathbf{H}_4$ in the full version of this paper [17]. The proof of the lemma is almost identical to the proof of Lemma 3.

Hybrid $\mathbf{H}_6$. The hybrid $\mathbf{H}_6$ is the same as $\mathbf{H}_5$ but adds the following decryption rule, on input a ciphertext $\mathtt{C} = ([\mathbf{x}]_1, [\mathbf{v}]_2, [\pi]_T)$:

(v) If $\mathbf{x} - \mathbf{x}^* \in \mathit{span}(\mathbf{D}^*)$ and $\mathbf{v} - \mathbf{v}^* \in \mathit{span}(\mathbf{E})$ then let $\tilde{\mathbf{r}}, \tilde{\mathbf{s}}$ be such that $\mathbf{x} - \mathbf{x}^* = \tilde{\mathbf{x}} = \mathbf{D}\tilde{\mathbf{r}}$ and $\mathbf{v} - \mathbf{v}^* = \tilde{\mathbf{v}} = \mathbf{E}\tilde{\mathbf{s}}$, and compute $[\pi_1]_T, [\pi_2]_T$ as follows:

$$[\pi_1]_T \underline{\leftarrow [\pi^*]_T + [(\mathbf{f}^\top \mathbf{D} + \tilde{\mathbf{v}}^\top \mathbf{F}^\top \mathbf{D})\tilde{\mathbf{x}}]_T}, \quad [\pi_2]_T \underline{\leftarrow [(\mathbf{g}\mathbf{E} + \tilde{\mathbf{x}}^\top \mathbf{G}^\top \mathbf{E})\tilde{\mathbf{v}}]_T},$$

This hybrid is equivalent to $\mathbf{H}_5$. The conditions of the decryption rule (v) imply that, if the proof π is correct, then the ciphertext $\mathtt{C}$ is a re-randomization of $\mathtt{C}^*$.

Hybrid $\mathbf{H}_7$. The hybrid $\mathbf{H}_7$ is the same as $\mathbf{H}_6$ but its decryption procedure does not execute the rules (*) introduced in the hybrid $\mathbf{H}_0$.

In Lemma 4 we show that $\mathbf{H}_7$ and $\mathbf{H}_6$ are identically distributed, while in the following we prove that the challenge bit b^* is perfectly hidden.

Lemma 2. $\Pr[\mathbf{H}_7 = 1] = \frac{1}{2}$.

Proof. We notice that in $\mathbf{H}_7$ the decryption procedure does not use the secret key $\mathbf{a}$ to perform the decryption; this can be easily confirmed by inspection of the decryption procedure in Fig. 4. Notice also that given the value $\mathbf{a}^\top \mathbf{D}$ the random variable $\mathbf{a}^\top \cdot \mathbf{u}^*$ is uniformly distributed. Thus, both the challenge ciphertext $\mathtt{C}^*$ and the answers of the decryption oracle are independent of the bit b^*.

Lemma 3. *The hybrids $\mathbf{H}_4$ and $\mathbf{H}_3$ are statistically close.*

Proof. We prove the statement with a hybrid argument over the number of decryption queries of the adversary. Let the hybrid $\mathbf{H}_{3,i}$ be the experiment that answers the first i-th oracle queries as in $\mathbf{H}_4$ (namely, considering the decryption rule (iii)) and answers the remaining queries as in $\mathbf{H}_3$. Let Q_D be the number of decryption queries performed by the adversary A. It is easy to check that $\mathbf{H}_{3,0} \equiv \mathbf{H}_3$ and $\mathbf{H}_{3,Q_D} \equiv \mathbf{H}_4$.

On the other hand $\mathbf{H}_{3,i}$ and $\mathbf{H}_{3,i+1}$ differ when the $(i+1)$-th ciphertext $\mathtt{C} = (([\mathbf{u}]_1, [p]_1), [\mathbf{v}]_2, [\pi]_T)$ is such that "$\mathbf{u} \notin \mathit{span}(\mathbf{D})$ and $((\mathbf{v} - \mathbf{v}^*) \notin \mathit{span}(\mathbf{E})$ or $\mathbf{v}^*$ is unset)", but the decryption oracle (as it would be computed in $\mathbf{H}_3$) outputs a value different from $\perp$. In particular, the latter implies that the proof $[\pi]_T$ verifies correctly. Let $\mathtt{Sound}_i$ be such event. To conclude the proof of the lemma we prove that $\Pr[\mathtt{Sound}_i] \leq 1/q$. Then a standard union bound gives us that the statistical distance between $\mathbf{H}_4$ and $\mathbf{H}_3$ is at most Q_D/q, which is negligible.

We reduce an adversary A that causes event $\mathtt{Sound}_i$ to occur into an adversary A' for the game of Lemma 1. Namely, we define an adversary A' for the experiment in the lemma which internally simulates the experiment $\mathbf{H}_{3,i+1}$ running with the adversary A.

$\underline{\text{Adversary } \mathsf{A}'(\mathbf{D}, \mathbf{E}, \mathbf{f}^\top \mathbf{D}, \mathbf{F}^\top \mathbf{D}, \mathbf{F}\mathbf{E})}$ with oracle access to $\mathcal{O}$:

1. Sample $\mathbf{a} \leftarrow_\$ \mathbb{Z}_q^{k+1}, \mathbf{g} \leftarrow_\$ \mathbb{Z}_q^{k+1}, \mathbf{G} \leftarrow_\$ \mathbb{Z}_q^{k+1 \times k+2}$.

2. Set the public key as: $\mathsf{pk} = \begin{pmatrix} [\mathbf{D}]_1, [\mathbf{E}]_2, [\mathbf{a}^\top \mathbf{D}]_1, [\mathbf{f}^\top \mathbf{D}]_T, [\mathbf{F}^\top \mathbf{D}]_1, \\ [\mathbf{g}^\top \mathbf{E}]_T, [\mathbf{G}^\top \mathbf{E}]_2, [\mathbf{G}\mathbf{D}^*]_1, [\mathbf{F}\mathbf{E}]_2 \end{pmatrix}$ as described by the key generation algorithm and set the secret key $\mathsf{sk} = (\mathbf{a}, \cdot, \mathbf{g}, \cdot, \mathbf{G})$.
3. Run the adversary A with input the public key pk. Answer the j-th decryption oracle query with ciphertext $\mathtt{C} = ([\mathbf{u}]_1, [p]_1, [\mathbf{v}]_2, [\pi]_T)$ as follows:
   (a) If $j \leq i$ and $\mathbf{u} \in span(\mathbf{D})$ compute, let $\mathbf{u} = \mathbf{D}\mathbf{r}$:
   $$[\mathtt{M}]_1 \leftarrow [p - \mathbf{a}^\top \mathbf{D} \cdot \mathbf{r}]_1, \qquad [\pi_1]_T \leftarrow [(\mathbf{f}^\top \mathbf{D} + \mathbf{v}^\top \cdot \mathbf{F}^\top \mathbf{D})]_T \cdot \mathbf{r},$$
   $$[\pi_2]_T \leftarrow [(\mathbf{g} + \mathbf{G} \cdot \mathbf{x})^\top \cdot \mathbf{v}]_T$$
   If $\pi = \pi_1 + \pi_2$ then answer with $[\mathtt{M}]_1$, else anwer $\perp$;
   (b) If $\mathbf{u} \notin span(\mathbf{D})$ answer $\perp$;
   (c) If $j = i + 1$ then stop and return $(\pi - (\mathbf{g} + \mathbf{G}\mathbf{x})^\top \mathbf{v}, \mathbf{u}, \mathbf{v})$.
4. Eventually, A outputs $[\mathtt{M}_0]_1, [\mathtt{M}_1]_1$. Sample $\mathbf{v}^* \leftarrow_\$ \mathbb{Z}_q^{k+1} \setminus span(\mathbf{E})$, and sample $\mathbf{u}^* \leftarrow_\$ \mathbb{Z}_q^{k+1} \setminus span(\mathbf{D})$, query the oracle $\mathcal{O}$ with the element $\mathbf{v}^*$ and receive $\mathit{\Pi} = \mathbf{f} + \mathbf{F} \cdot \mathbf{v}^*$. Set $p^* = \mathbf{a}^\top \mathbf{u}^* + \mathtt{M}_{b^*}$ and $\mathbf{x}^* = ((\mathbf{u}^*)^\top, p^*)^\top$, and:
   $$[\pi^*]_T \leftarrow [\mathit{\Pi}^\top \cdot \mathbf{u}^* + (\mathbf{g} + \mathbf{G}\mathbf{x}^*)^\top \mathbf{v}]_T \tag{2}$$
   and send to the adversary the challenge ciphertext $\mathtt{C}^* = ([\mathbf{c}^*]_1, [p^*]_1, [\mathbf{v}]_2, [\pi^*]_T)$.
5. Answer the j-th decryption oracle query with ciphertext $\mathtt{C} = ([\mathbf{u}]_1, [p]_1, [\mathbf{v}]_2, [\pi]_T)$ as follows:
   (a) If $j \leq i$ and $\mathbf{u} \in span(\mathbf{D})$ execute the same as in step 3a.
   (b) If $j \leq i$ and $\mathbf{u} \notin span(\mathbf{D})$ do as follows:
      i. if $(\mathbf{v}^* - \mathbf{v}) \in span(\mathbf{E})$ let $\mathbf{v} = \mathbf{v}^* + \mathbf{E}\boldsymbol{\gamma}$, compute
      $$[\pi_1]_T \leftarrow [(\mathit{\Pi} + \mathbf{F}\mathbf{E}\boldsymbol{\gamma})^\top)\mathbf{u}]_T, \quad [\pi_2]_T \leftarrow [(\mathbf{g}^\top + \mathbf{G}\mathbf{x})^\top \mathbf{v}]_T$$
      if $\pi = \pi_1 + \pi_2$ then answer $[p - \mathbf{a}^\top \cdot \mathbf{u}]_1$ else answer $\perp$.
      ii. if $(\mathbf{v}^* - \mathbf{v}) \notin span(\mathbf{E})$ then output $\perp$.
   (c) If $j = i + 1$ then stop and return $(\pi - (\mathbf{g} + \mathbf{G}\mathbf{x})^\top \mathbf{v}, \mathbf{u}, \mathbf{v})$.

We show that the adversary perfectly simulates the hybrid $\mathbf{H}_{3,i}$ up to the i-th decryption query. By inspection, it is easy to check that up to step 3, the simulation is perfect[9].

More interestingly, at step 4 the adversary A' uses its oracle to compute $\mathit{\Pi} = \mathbf{f} + \mathbf{F}\mathbf{v}^*$. Thanks to this information the adversary can compute the challenge ciphertext exactly as the hybrid experiment would do as shown in eq. 2. After this step, the adversary A' can easily answer the decryption queries whenever $j \leq i$ and $\mathbf{u} \in span(\mathbf{D})$ or $\mathbf{u} \notin span(\mathbf{D})$ and $(\mathbf{v}^* - \mathbf{v}) \notin span(\mathbf{E})$. We show

[9] The adversary computes π_2 in step 3a as the original decryption procedure would do, but by the modification in $\mathbf{H}_1$ we are assured that this is equivalent.

that the answers for the decryption queries where $j \leq i$, $\mathbf{u} \notin span(\mathbf{D})$ and $(\mathbf{v}^* - \mathbf{v}) \in span(\mathbf{E})$ are distributed exactly as in the hybrid experiment, in fact:

$$(\Pi + \mathbf{FE}\boldsymbol{\gamma})^\top \mathbf{u} = \mathbf{f}^\top \mathbf{u} + (\mathbf{Fv}^*)^\top \mathbf{u} + (\mathbf{FE}\boldsymbol{\gamma})^\top \mathbf{u} = \mathbf{f}^\top \mathbf{u} + (\mathbf{F}(\mathbf{v}^* + \mathbf{E}\boldsymbol{\gamma}))^\top \mathbf{u} = (\mathbf{f} + \mathbf{Fv})^\top \mathbf{u}.$$

Finally, by definition of $\mathtt{Sound}_i$, the adversary A at the $(j+1)$-th query outputs a ciphertext that would correctly decrypt in the hybrid experiment and where $\mathbf{u} \notin span(\mathbf{D})$ and $(\mathbf{v}^* - \mathbf{v}) \notin span(\mathbf{E})$ with probability $\Pr[\mathtt{Sound}_i]$. Since the ciphertext correctly decrypts, it means that $\pi = (\mathbf{f} + \mathbf{Fv})^\top \mathbf{u} + (\mathbf{g} + \mathbf{Gx})^\top \mathbf{v}$, therefore the output of A' is a valid guess for the experiment of Lemma 1. However, the adversary A' can win with probability at most $1/q$, and thus the lemma follows.

Lemma 4. *The hybrids* $\mathbf{H}_6$ *and* $\mathbf{H}_7$ *are identically distributed.*

Proof. We prove this lemma by showing that in $\mathbf{H}_6$ the decryption procedure never executes the lines with rules (*). To do this, for any ciphertext queried to the decryption oracle we partition over all possible cases and show that the decryption procedure used for the oracle queries either sets the values $\mathtt{M}, \pi_1, \pi_2$ (and thus the rules (*) are not executed) or it stops before reaching those rules as it outputs $\perp$ or $\diamond$. Let $\mathtt{C} = ([\mathbf{x}]_1, [\mathbf{v}]_2, [\pi]_T)$ be the ciphertext queried to the oracle, where $[\mathbf{x}^\top]_1 = ([\mathbf{u}^\top]_1, [p]_1)$. We consider all the possible alternatives:

- $\mathbf{u} \in span(\mathbf{D})$: notice that in this case, by the rule (i), $\mathtt{M}$ and π_1 are set;
  - $\mathbf{v} \in span(\mathbf{E})$: notice that in this case, by rule (ii), π_2 is also set. Therefore, since in this branch $\mathtt{M}, \pi_1, \pi_2$ are set, the rules (*) are not executed.
  - $\mathbf{v} \notin span(\mathbf{E})$: in this case we enter rule (iv) and thus decryption stops and outputs $\perp$. To see why this rule is entered, notice that either $\mathbf{u}^*$ is unset, or, if it is set, then $\mathbf{u}^* \notin span(\mathbf{D})$, and so $\mathbf{x} - \mathbf{x}^* \notin span(\mathbf{D}^*)$.
- $\mathbf{u} \notin span(\mathbf{D})$, in this case the output could be either $\diamond$ or $\perp$, more in details:
  - $\mathbf{v}^*$ is unset: by rule (iii) decryption stops and outputs $\perp$.
  - $\mathbf{v}^*$ is set and $(\mathbf{v} - \mathbf{v}^*) \notin span(\mathbf{E})$: by rule (iii) decryption outputs $\perp$.
  - $\mathbf{v}^*$ is set and $(\mathbf{v} - \mathbf{v}^*) \in span(\mathbf{E})$:
    - $(\mathbf{x} - \mathbf{x}^*) \notin span(\mathbf{D}^*)$: notice that since $\mathbf{v}^* \notin span(\mathbf{E})$ then it must be that $\mathbf{v} \notin span(\mathbf{E})$. Hence, rule (iv) is entered and decryption outputs $\perp$.
    - $(\mathbf{x} - \mathbf{x}^*) \in span(\mathbf{D}^*)$: rule (v) is entered, decryption outputs $\diamond$, so $\mathtt{M}, \pi_1, \pi_2$ are set, and thus the rules (*) are not executed.

4 Our Publicly-Verifiable Rand-RCCA PKE

Here we show that our RCCA scheme from the previous section can be turned into a publicly verifiable one. Very informally, the idea is to append a malleable proof (essentially a GS proof) that $[\pi]_T$ is well formed. The decryption procedure of the publicly verifiable scheme can simply check the validity of the proof and then CPA-decrypt the ciphertext $[\mathbf{x}]_1$. Let $\mathcal{PKE}_1 = (\mathsf{KGen}_1, \mathsf{Enc}_1, \mathsf{Dec}_1, \mathsf{Rand}_1)$

$\underline{\mathsf{KGen}_2(\mathtt{prm})}$:
$(\mathsf{pk}', \mathsf{sk}') \leftarrow_\$ \mathsf{KGen}'(\mathtt{prm})$, $\mathsf{crs} \leftarrow \mathsf{Init}(\mathtt{prm})$;
Parse $\mathsf{sk}' = (\mathbf{a}, \mathbf{f}, \mathbf{F}, \mathbf{g}, \mathbf{G})$;
Set $\mathsf{sk} = (\mathbf{a}, \mathsf{crs})$, $\mathsf{pk} = (\mathsf{pk}', \mathsf{crs})$;
Output $(\mathsf{pk}, \mathsf{sk})$.

$\underline{\mathsf{Rand}_2(\mathsf{pk}, \mathtt{C})}$:
Parse $\mathtt{C} = ([\mathbf{x}]_1, [\mathbf{v}]_2, \Pi)$,
$T \leftarrow_\$ \mathcal{T}$, (with associated $\hat{\mathbf{r}}, \hat{\mathbf{s}} \in \mathbb{Z}_q^k$)
$\hat{\mathbf{x}} = \mathbf{x} + \mathbf{D}^* \cdot \hat{\mathbf{r}}$;
$\hat{\mathbf{v}} = \mathbf{v} + \mathbf{E} \cdot \hat{\mathbf{s}}$;
$\hat{\Pi} = \mathsf{ZKEval}(\mathsf{crs}, T, ([\mathbf{x}]_1, [\mathbf{v}]_2), \Pi)$;
Output $([\hat{\mathbf{x}}]_1, [\hat{\mathbf{v}}]_2, \hat{\Pi})$.

$\underline{\mathsf{Enc}_2(\mathsf{pk}, [\mathtt{M}]_1)}$:
$\mathbf{r}, \mathbf{s} \leftarrow_\$ \mathbb{Z}_q^k$;
$([\mathbf{x}]_1, [\mathbf{v}]_2, [\pi]_T) \leftarrow \mathsf{Enc}'(\mathsf{pk}, [\mathtt{M}]_1; \mathbf{r}, \mathbf{s})$;
$\Pi \leftarrow_\$ \mathsf{P}(\mathsf{crs}, ([\mathbf{x}]_1, [\mathbf{v}]_2), ([\pi]_T, \mathbf{r}, \mathbf{s}))$;
Output $\mathtt{C} = ([\mathbf{x}]_1, [\mathbf{v}]_2, \Pi)$.

$\underline{\mathsf{Dec}_2(\mathsf{sk}, \mathtt{C})}$:
Parse $\mathtt{C} = ([\mathbf{x}]_1, [\mathbf{v}]_2, \Pi)$;
if $\mathsf{V}(\mathsf{crs}, ([\mathbf{x}]_1, [\mathbf{v}]_2), \Pi) = 1$
output $(-\mathbf{a}^\top, 1) \cdot [\mathbf{x}]_1$;
else output $\perp$.

$\underline{\mathsf{Ver}(\mathsf{pk}, \mathtt{C})}$:
Parse $\mathtt{C} = ([\mathbf{x}]_1, [\mathbf{v}]_2, \Pi)$;
Output $\mathsf{V}(\mathsf{crs}, ([\mathbf{x}]_1, [\mathbf{v}]_2), \Pi)$.

Fig. 5: Our publicly-verifiable re-randomizable RCCA encryption scheme $\mathcal{PKE}_2$. The NIZK is for the relation $\mathcal{R}_{\mathcal{PKE}_1}$ and transformation $\mathcal{T}_{\mathcal{PKE}_1}$.

be the scheme of Sec. 3 and let $\mathcal{NIZK} = (\mathsf{Init}, \mathsf{P}, \mathsf{V}, \mathsf{ZKEval})$ be a malleable NIZK system for membership in the relation defined below:

$$\mathcal{R}_{\mathcal{PKE}_1} = \left\{ ([\mathbf{x}]_1, [\mathbf{v}]_2), ([\pi]_T, \mathbf{r}, \mathbf{s}) : [\pi]_T = [(\mathbf{f} + \mathbf{F}\mathbf{v})^\top \mathbf{u} + (\mathbf{g} + \mathbf{G}\mathbf{x})^\top \mathbf{v}]_T \right\},$$

and with allowable set of transformations:

$$\mathcal{T}_{\mathcal{PKE}_1} = \left\{ T : \begin{array}{c} \exists \hat{\mathbf{r}}, \hat{\mathbf{s}} \in \mathbb{Z}_q^k : \\ \end{array} \begin{array}{l} T_x([\mathbf{x}]_1, [\mathbf{v}]_2) = ([\hat{\mathbf{x}}]_1, [\hat{\mathbf{v}}]_2) \\ T_w([\pi]_T, \mathbf{r}, \mathbf{s}) = ([\hat{\pi}]_T, \mathbf{r} + \hat{\mathbf{r}}, \mathbf{s} + \hat{\mathbf{s}}) \\ ([\hat{\mathbf{x}}]_1, [\hat{\mathbf{v}}]_2, [\hat{\pi}]_T) = \mathsf{Rand}_1(\mathsf{pk}, ([\mathbf{x}]_1, [\mathbf{v}]_2, [\pi]_T); \hat{\mathbf{r}}, \hat{\mathbf{s}}) \end{array} \right\}.$$

We write $T \leftarrow_\$ \mathcal{T}_{\mathcal{PKE}_1}$ for the operation that samples the uniquely defined $\hat{\mathbf{r}}, \hat{\mathbf{s}}$ associated to the transformation T. The pv-Rand-PKE scheme $\mathcal{PKE}_2 = (\mathsf{Init}, \mathsf{KGen}_2, \mathsf{Enc}_2, \mathsf{Dec}_2, \mathsf{Rand}_2, \mathsf{Ver})$ is described in Fig. 5. We defer the proof of the following theorem in the full version of this paper [17].

Theorem 2. *If the $\mathcal{NIZK}$ is adaptive sound and perfect derivation private then the pv-Rand-PKE scheme $\mathcal{PKE}_2$ described in Fig. 5 is publicly verifiable, perfect re-randomizable and RCCA-secure.*

Malleable NIZK. The equations we would like to prove do not admit Groth-Sahai NIZK proofs [26], but only NIWI. We overcome this problem by developing a new technique that extends the class of pairing product equations which admit GS NIZK proofs. This technique is *per se* a result of independent interest.

More in detail, we produce an additional commitment to $[\pi]_T$, using a new commitment type defined over $\mathbb{G}_T$ with good bilinear properties. This allows us to construct a NIZK proof that the ciphertext is valid with perfect completeness and soundness and composable zero-knowledge. The latter notion refers to the fact that if the common reference string is defined in a "witness indistinguishable mode", the proof system is perfect zero-knowledge. By replacing $[\pi_T]$ in

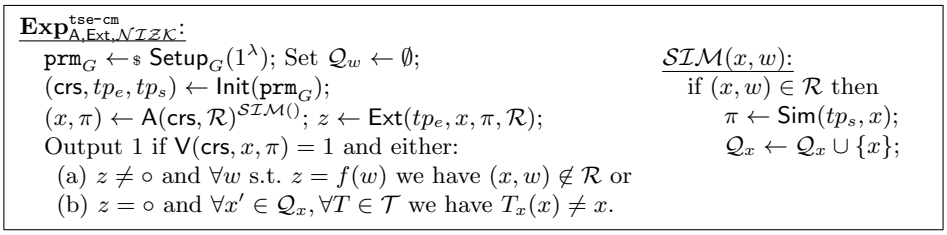

Fig. 6: The security experiments for the NIZK argument system.

the ciphertext by its commitment, in the witness indistinguishable mode we can simulate a proof of validity of the ciphertext by setting $\pi = 0$ and in an undetectable manner. The proof will be correctly distributed because of the perfect zero-knowledge property in these modes.

All the details on how to compute the proof are given in the full version of this paper [17]. Beyond GS Proofs, it also makes use of the QANIZK proof of membership in linear spaces [29,30,33]. The size of the ciphertexts for the SXDH instantiation of the publicly verifiable scheme is $12|\mathbb{G}_1| + 11|\mathbb{G}_2| + 4|\mathbb{G}_T|$. The number of pairings for verification is 32 for the GS proof and 14 for the argument of linear spaces, which can be reduced to $8 + 14$ by batch verifying the GS equation using the techniques of [28].

5 Malleable and True-Simulation Extractable NIZK

In this section we show an application of our Rand-RCCA scheme to build a malleable and true-simulation extractable NIZK.

True-Simulation Extractability. We recall the notion of true-simulation f-extractability (f-tSE-NIZK, for short) of Dodis *et al.*[12]. The notion is a weakening of the concept of simulation extractability where the extractor can compute a function of the witness and the adversary sees simulated proofs only for true statements. Here, we give a variation of the notion that allows for re-randomizability (and malleability). Consider the experiment described in Fig. 6, the main difference respect to the notion of [12], is that the winning condition (b) allows the extractor to give up and output a special symbol $\circ$. The restriction is that the extractor can safely do this without losing the game only when the proof π produced by the adversary is derived from a simulated proof.

Definition 7. *Let f be an efficiently computable function, let $\mathcal{NIZK} = (\mathsf{Init}, \mathsf{P}, \mathsf{V})$ be a NIZK argument for a relation $\mathcal{R}$, and consider the experiment* $\mathbf{Exp}^{\mathtt{tse\text{-}cm}}$ *described in Fig. 6. We say that $\mathcal{NIZK}$ is* true-simulation controlled-malleable f-extractable *(f-tSE-cm) iff there exists a PPT algorithm* Ext *such that for all PPT* A *we have that*

$$\mathbf{Adv}^{\mathtt{tse-cm}}_{\mathsf{A},\mathsf{Ext},\mathcal{NIZK}}(\lambda) := \Pr\left[\mathbf{Exp}^{\mathtt{tse\text{-}cm}}_{\mathsf{A},\mathsf{Ext},\mathcal{NIZK}}(1^\lambda) = 1\right] \in \mathtt{negl}(\lambda).$$

Construction. The construction follows the blueprint of Dodis *et al.* [12] with the twist that we use a Rand RCCA-PKE scheme instead of a CCA-PKE scheme. Our compiler works for a special class of tuples, consisting of a function f, an NP relation $\mathcal{R}$ and a transformation $\mathcal{T}$, that we define below:

Definition 8. *A tuple $(f, \mathcal{R}, \mathcal{T})$, where f is efficiently computable, $\mathcal{R}$ is an NP-relation and $\mathcal{T}$ is an admissible transformation for $\mathcal{R}$, is* suitable *if:*

1. *there exists an efficiently computable decision procedure g such that for any (x, w) the function $g(x, f(w)) = 1$ if and only if $(x, w) \in \mathcal{R}$;*
2. *For any $T \in \mathcal{T}$ and any $(x, w) \in \mathcal{R}$ the transformation of the witness is invariant respect to the function f, namely $f(w) = f(T_w(w))$.*

The restrictions above still allow for many interesting malleabilities. For example, the condition (2) clearly applies to re-randomizable NIZKs, as in this case $T_w(\cdot)$ is the identity function. Condition (1) holds in all those cases where the relation $\mathcal{R}$ can be sampled together with a trapdoor information that allows to compute w from x. The condition (1) applies also to the NIZKs of [12]. More importantly, the conjunction of (1) and (2) allows to efficiently check the condition (b) of the security experiment, which makes the tSE-cm NIZK primitive easier to use.

Let $\mathcal{PKE} = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Rand})$ be a Rand-RCCA PKE scheme, we additionally assume there exists an integer $\ell \in \mathbb{N}$ such that the random coins of both the encryption procedure and the re-randomization procedure are in $\mathbb{Z}_q^\ell$ and that, for any $\mathsf{pk}, \mathtt{M}$, given $\mathsf{Rand}(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mathtt{M}; \rho_0); \rho_1) = \mathsf{Enc}(\mathsf{pk}, \mathtt{M}; \rho_0 + \rho_1)$ where $\rho_0, \rho_1 \in \mathbb{Z}_q^\ell$. Notice that the schemes in Sec. 3 and Sec 4 have this property. Let $\mathcal{R}$ be a NP relation and $\mathcal{T}$ be a set of allowable transformations for the relation $\mathcal{R}$. Let $\mathcal{NIZK}' = (\mathsf{Init}', \mathsf{P}', \mathsf{V}', \mathsf{ZKEval}')$ be a malleable NIZK argument for $\mathcal{R}'$ with the allowable set of transformations $\mathcal{T}'$ as described below:

$$\mathcal{R}' = \{((\mathsf{pk}, c, x), (w, \rho)) : \ (x, w) \in \mathcal{R} \ \wedge \ c = \mathsf{Enc}(\mathsf{pk}, f(w); \rho)\}$$

$$\mathcal{T}' = \left\{ T' : \ \exists \hat{\rho}, T : \begin{array}{c} T'_x(\mathsf{pk}, c, x) = (\mathsf{pk}, \mathsf{Rand}(\mathsf{pk}, c; \hat{\rho}), T_x(x)), \\ T'_w(w, \rho) = (T_w(w), \rho + \hat{\rho}), \quad T \in \mathcal{T} \end{array} \right\}$$

We also assume that any transformation $T' \in \mathcal{T}'$ can be efficiently parsed as a tuple $(\hat{\rho}, T)$ and viceversa. We define a malleable NIZK argument $\mathcal{NIZK} = (\mathsf{Init}, \mathsf{P}, \mathsf{V}, \mathsf{ZKEval})$ for the relation $\mathcal{R}$ with allowable set of transformations $\mathcal{T}$ in Fig 7. Notice that the co-domain of the function f for which we can prove f-tSE soundness is the message space of the underlying Rand-RCCA PKE scheme. We remark that, although our scheme is presented with a message space $\mathcal{M} = \mathbb{G}_1$, we could easily extend our construction to encrypt vectors in $\mathbb{G}_1^{\ell_0} \times \mathbb{G}_2^{\ell_1}$.

Theorem 3. *For any suitable $(f, \mathcal{R}, \mathcal{T})$ the proof system $\mathcal{NIZK}$ is a malleable NIZK for $\mathcal{R}$ with allowable transformations $\mathcal{T}$, and if $\mathcal{NIZK}'$ is perfectly (resp. statistically) derivation private (Def. 6) and $\mathcal{PKE}$ is perfectly re-randomizable then $\mathcal{NIZK}$ is perfectly (resp. statistically) derivation private.*

Theorem 4. *For any suitable $(f, \mathcal{R}, \mathcal{T})$ the proof system $\mathcal{NIZK}$ described above is true-simulation controlled-malleable f-extractable.*

$\underline{\mathsf{Init}(\mathtt{prm})}$:
$(\mathsf{crs}', tp'_s) \leftarrow \mathsf{Init}'(\mathtt{prm})$;
$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(\mathtt{prm})$;
$\mathsf{crs} \leftarrow (\mathsf{crs}', \mathsf{pk})$, $tp_e \leftarrow \mathsf{sk}, tp_s \leftarrow (\mathsf{pk}, tp'_s)$
Output $(\mathsf{crs}, tp_e, tp_s)$.

$\underline{\mathsf{P}(\mathsf{crs}, x, w)}$:
$\mathtt{C} \leftarrow \mathsf{Enc}(\mathsf{pk}, f(w); r)$;
$\pi' \leftarrow \mathsf{P}'(\mathsf{crs}', (\mathsf{pk}, \mathtt{C}, x), (w, r))$;
Output $\pi = (\mathtt{C}, \pi')$.

$\underline{\mathsf{V}(\mathsf{crs}, x, \pi)}$:
Output $\mathsf{V}'(\mathsf{crs}', (\mathsf{pk}, \mathtt{C}, x), \pi')$

$\underline{\mathsf{ZKEval}(\mathsf{crs}, T, (x, \pi))}$:
Let $\pi = (\mathtt{C}, \pi')$, $\rho \leftarrow_{\$} \mathbb{Z}_q^{\ell}$;
Let $T' = (\rho, T)$;
$\hat{\mathtt{C}} \leftarrow \mathsf{Rand}(\mathsf{pk}, \mathtt{C}; \rho)$;
$\hat{\pi}' \leftarrow_{\$} \mathsf{ZKEval}'(\mathsf{crs}', T', (x, \pi'))$;
Output $(\hat{\mathtt{C}}, \hat{\pi}')$.

Fig. 7: Our f-tSE-cm $\mathcal{NIZK}$ compiler.

The proofs of Theorems 3 and 4 are in the full version of this paper [17]. We give an intuition for the proof of Theorem 4, which proceeds with a two-steps hybrid argument. We start with the true-simulation extractability experiment, we can switch to an experiment where each simulated proof for $\mathcal{NIZK}$ contains an encryption of the $f(w)$. This step can be easily argued using the RCCA security of the scheme. In particular, the guarded decryption oracle and the suitability of $(f, \mathcal{R}, \mathcal{T})$ are necessary to check the winning condition of the tSE experiment. In the second step, we switch to valid proofs for $\mathcal{NIZK}'$, instead of simulated proofs, the indistinguishability follows trivially by the zero-knowledge of $\mathcal{NIZK}'$. At this point we are in an experiment where the proofs provided by the $\mathcal{SIM}$ are not simulated, so the standard adaptive soundness of $\mathcal{NIZK}'$ is sufficient to bound the winning probability of the adversary.

Instantiation. For any suitable $(f, \mathcal{R}, \mathcal{T})$ where the co-domain of f is $\mathbb{G}_1$, we can instantiate the tSE-cm NIZK scheme with the pv-Rand-RCCA Scheme $\mathcal{PKE}_2$. The public verifiability enables for a simpler malleable NIZK proof for the associated $\mathcal{R}'$. In fact, we can subdivide the proof in: (1) a malleable GS proof Π_1 for $\mathcal{R}$ with transformations $\mathcal{T}$, in particular Π_1 contains GS commitments $[\mathbf{c}_w]_1$ of the witness; (2) a malleable GS proof Π_2 to prove that commitments $[\mathbf{c}_w]_1$ and $[\mathbf{c}_{w'}]_1$ open to w, w' an $w' = f(w)$; (3) a malleable proof Π_3 to prove $w' = (-\mathbf{a}^T, 1) \cdot [\mathbf{x}]$, in particular, from the linearity of GS commitments the relation for the last proof is a linear subspace relationship. The verification checks the proofs Π_1, Π_2, Π_3 and verifies the validity of the ciphertext $\mathtt{C}$.

For the case where f is the identity function, namely, re-randomizable NIZK, the proof Π_2 is trivial as we can set $[\mathbf{c}_w]_1 = [\mathbf{c}_{w'}]_1$. The overhead in proof size between a adaptive sound re-randomizable GS proof for $\mathcal{R}$ based on SXDH and an tSE-cm NIZK based on SXDH is equal to $13|\mathbb{G}_1| + 11|\mathbb{G}_2| + 4|\mathbb{G}_T|$.

6 An UC-Secure Mix-Net

In this section we propose an application of pv-Rand-PKE schemes with RCCA security to Mix-Net protocols. Our starting point is a recent work of Faonio and Fiore [16] who build an UC-secure Optimistic Mix-Net using a new paradigm that relies on a specific re-randomizable and RCCA-secure PKE scheme. Here

we extend the main idea of [16] and use the power of public verifiability in order to obtain a full fledged Mix-Net protocol (not only optimistic secure).

The Universal Composability model. We review some basic notions of the Universal Composability model and the extension to *auditable protocols* of Faonio and Fiore. In a nutshell, a protocol Π $\mathsf{UC\text{-}realizes}$ an ideal functionality $\mathcal{F}$ with setup assumption $\mathcal{G}$ if there exists a PPT simulator S such that no PPT environment $\mathcal{Z}$ can distinguish an execution of the protocols Π which can interact with the setup assumption $\mathcal{G}$ from a joint execution of the simulator S with the ideal functionality $\mathcal{F}$. The environment $\mathcal{Z}$ provides the inputs to all the parties of the protocols, decides which party to corrupt (we consider static corruption, where the environment decides the corrupted parties before the protocol starts), and schedules the order of the messages in the networks. When specifying an ideal functionality, we use the "delayed outputs" terminology of Canetti [4]. Namely, when a functionality $\mathcal{F}$ *sends a public delayed output* M *to party* $\mathcal{P}_{P_i}$ we mean that M is first sent to the simulator and then forwarded to $\mathcal{P}_{P_i}$ only after acknowledgement by the simulator. Faonio and Fiore consider a variation of the UC model where, roughly speaking, a bulletin board functionality **BB** acts as global setup assumption. More in details, the bulletin board is present in both the ideal world and the real world, so that the simulator does not have any advantage over the real-world adversary and all the parties of the protocol can register their message on the board. An *auditable protocol* is a tuple (Π, Audit) where Π is a protocol and Audit is a PPT algorithm. The model additionally includes an external off-line party, the auditor. The auditor is an incorruptible party which, whenever is called on an input y', runs the audit algorithm Audit on this input and the transcript written in the bulletin boards and forwards its output to the environment. In the ideal world, the auditor always replies according to the output of the ideal functionality, for example, if the ideal functionality has output y and the auditor is called on input y', the auditor replies with `valid` if and only if $y = y'$.

Defining Mix-Net Protocols. Our protocol UC-realizes the ideal functionality $\mathcal{F}_{\mathsf{Mix}}$ described in Fig. 8 with setup assumptions: the ideal functionality $\mathcal{F}_{\mathsf{TDec}}$ for threshold decryption of our PKE scheme and the ideal functionality for a common-reference string $\mathcal{F}_{\mathsf{CRS}}$ (and the bulletin board of the auditable framework of Faonio and Fiore). The functionality $\mathcal{F}_{\mathsf{Mix}}$ (similarly to [16]) is slightly weaker than the one considered by Wikström in [44,45]. The difference is that the corrupted senders can replace their inputs, however, they loose this ability when the first honest mixer sends its message `mix`. On the other hand, in the ideal functionality of Wikström, the senders can cast their messages only during the inputs submission phase.

Building blocks. The main building blocks of our mix-net construction are:

(i) An *linear* pv-Rand-RCCA PKE scheme $\mathcal{PKE}$. We say that a pv-Rand-RCCA PKE scheme is *linear* if there exist a group $\mathbb{G}$ (for example $\mathbb{G} = \mathbb{G}_1$) and parameters $\ell, \ell', \ell'' \in \mathbb{N}$ such that (1) every key pair $(\mathsf{pk}, \mathsf{sk})$ we can parse $\mathsf{pk} = ([\mathbf{P}], \hat{\mathsf{pk}})$ and $\mathsf{sk} = (\mathbf{S}, \hat{\mathsf{sk}})$, where $[\mathbf{P}] \in \mathbb{G}^{\ell \times \ell''}$ and $\mathbf{S} \in \mathbb{Z}_q^{\ell' \times \ell}$, (2) any

Functionality $\mathcal{F}_{\mathsf{Mix}}$:

The functionality has n sender parties $\mathcal{P}_{S_i}$ and m mixer parties $\mathcal{P}_{M_i}$:

Input: On message $(\mathtt{input}, \mathtt{M}_i)$ from $\mathcal{P}_{S_i}$ (or the adversary if $\mathcal{P}_{S_i}$ is corrupted) register the index i in the list of the senders and register the entry $(i, \mathtt{M}_i)$ in the database of the inputs. Notify the adversary that the sender $\mathcal{P}_{S_i}$ has sent its input.

Mix: On message $\mathtt{mix}$ from $\mathcal{P}_{M_i}$ (or the adversary if $\mathcal{P}_{M_i}$ is corrupted), register the index i in the list of the mixers and notify the adversary.

Delivery: If all the senders are in the list of the senders and at least one honest mixer is in the list of the mixers send a public delayed output $\mathcal{O} \leftarrow \mathsf{Sort}(\langle \mathtt{M}_j \rangle_{j \in [n]})$ to all the mixers.

Fig. 8: Ideal Functionality for Mixing.

ciphertext $\mathtt{C} \in \mathcal{C}$ can be parsed as $([\mathbf{y}], \hat{\mathtt{C}})$ where $[\mathbf{y}] \in \mathbb{G}^{\ell}$, (3) for any ciphertext $\mathtt{C}$ such that $\mathsf{Ver}(\mathsf{pk}, \mathtt{C}) = 1$ the decryption procedure is linear, i.e., we have $\mathsf{Dec}(\mathsf{sk}, \mathtt{C}) = \mathbf{S} \cdot [\mathbf{y}]$ (4) let $\mathtt{C}' = \mathsf{Rand}(\mathsf{pk}, \mathtt{C}; \mathbf{r}, r)$ where $\mathtt{C}' = ([\mathbf{y}'], \hat{\mathtt{C}}')$ be a re-randomization of $\mathtt{C} = ([\mathbf{y}], \hat{\mathtt{C}})$ and $\mathbf{r} \in \mathbb{Z}_q^{\ell''}$ then $([\mathbf{y}] - [\mathbf{y}']) = [\mathbf{P}]\mathbf{r}$. We notice that both the scheme $\mathcal{PKE}_2$ in Sec. 4 and the pv-Rand-RCCA PKE scheme of [34,8] are linear. Indeed, our abstraction is made to include the three schemes under the same template.

(ii) An All-but-One label-based NIZK. An ABO label-based $\mathcal{NIZK}_{\mathtt{sd}} = (\mathsf{Init}_{\mathtt{sd}}, \mathsf{P}_{\mathtt{sd}}, \mathsf{V}_{\mathtt{sd}})$ for knowledge of the plaintext of the linear PKE. More in details a ABO label-based $\mathcal{NIZK}$ is a NIZK system with labels where there exists an algorithm $\mathsf{ABOInit}(\mathtt{prm}, \tau)$ which creates a common reference string crs together with a trapdoor tp_s such that for any label $\tau' \neq \tau$ the trapdoor allows for zero-knowledge while for τ the proof system is adaptive sound. A ABO label-based $\mathcal{NIZK}$ in the random-string model can be easily obtained from GS NIZK proof system.

(iii) An adaptive sound NIZK. $\mathcal{NIZK}_{\mathtt{mx}} = (\mathsf{Init}_{\mathtt{mx}}, \mathsf{P}_{\mathtt{mx}}, \mathsf{V}_{\mathtt{mx}})$ for proving membership in the relation $\mathcal{R}_{\mathtt{mx}} = \{([\mathbf{P}], [\mathbf{y}]) : [\mathbf{y}] \in span([\mathbf{P}])\}$. We recall that GS proof system is in the random-string model.

(iv) An ideal functionality $\mathcal{F}_{\mathsf{TDec}}$ for threshold decryption of the pv-Rand-RCCA PKE $\mathcal{PKE}$ scheme. More in details, $\mathcal{F}_{\mathsf{TDec}}$ takes as parameters the definition of the PKE scheme and group parameters $\mathtt{prm}$ for the key generation. The functionality initializes a fresh key pair and accepts input of the form $(\mathtt{dec}, \mathtt{C})$ from the mixers: when a mixer sends a message of this kind, we say that the mixer *asks for the decryption of* $\mathtt{C}$. When all all the mixers have sent a message of the form $(\mathtt{dec}, \mathtt{C})$ the functionality sends a public delayed output $\mathsf{Dec}(\mathsf{sk}, \mathtt{C})$: in this case we say that the mixers *agreed on the decryption of* $\mathtt{C}$. In the full version of this paper [17] we show a protocol for the functionality $\mathcal{F}_{\mathsf{TDec}}$ in the $\mathcal{F}_{\mathsf{CRS}}$-hybrid world.

(v) An ideal functionality for the common reference string of the above NIZKs. The functionality initializes m different CRS $\{\mathsf{crs}^i_{\mathtt{mx}}\}_{i=1,\ldots,m}$, one for each mixer,[10] for $\mathcal{NIZK}_{\mathtt{mx}}$ and a CRS $\mathsf{crs}_{\mathtt{sd}}$ for $\mathcal{NIZK}_{\mathtt{sd}}$. We stress that all the CRSs can be sampled as uniformly random strings in the real protocol.

Also we recall that our auditable protocol uses a Bulletin Board functionality. We do not mention it as a "building block" because every auditable protocol, as defined by [16], necessarily needs a bulletin board as setup assumption.

Our Mix-Net Protocol. Following the design rationale of Faonio and Fiore, given two lists of ciphertexts $\mathcal{L} = \langle \mathtt{C}_1, \ldots, \mathtt{C}_n \rangle$ and $\mathcal{L}' = \langle \mathtt{C}'_1, \ldots, \mathtt{C}'_n \rangle$, we define the *checksum* of these lists as the output of the following procedure:

Procedure $\mathbf{CkSum}(\mathcal{L}, \mathcal{L}')$:

1. For all $j \in [n]$ parse $\mathtt{C}_j = ([\mathbf{y}_j], \hat{\mathtt{C}}_j)$ and $\mathtt{C}'_j = ([\mathbf{y}'_j], \hat{\mathtt{C}}'_j)$;
2. Output $\sum_j [\mathbf{y}_j] - [\mathbf{y}'_j]$.

We describe our mix-net protocol Π between n sender parties $\mathcal{P}_{S_i}$ and m mixer parties $\mathcal{P}_{M_i}$ and with resources the ideal functionalities $\mathcal{F}_{\mathsf{TDec}}$ and $\mathcal{F}_{\mathsf{CRS}}$:

Inputs Submission. Every sender $\mathcal{P}_{S_j}$, with $j \in [n]$, encrypts its message $\mathtt{M}_j$ by computing $\mathtt{C}_j \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathtt{M}_j; r)$, and creates a NIZK proof of knowledge $\pi^{\mathtt{sd}}_j \leftarrow \mathsf{P}_{\mathtt{sd}}(\mathsf{crs}_{\mathtt{sd}}, j, (\mathsf{pk}, \mathtt{C}), (\mathtt{M}_j, r))$ (the label for the proof is j). The party $\mathcal{P}_{S_j}$ posts $(\mathtt{C}_j, \pi^{\mathtt{sd}}_j)$ on the bulletin board.

Mix. Once all the senders are done with the previous phase, let $\mathcal{L}_0 = \langle \mathtt{C}_{0,j} \rangle_{j \in [n]}$ be the list of ciphertexts they posted on the bulletin board. To simplify the exposition of the result, we assume that all the NIZK proofs $\{\pi^{\mathtt{sd}}_j\}_{j \in [n]}$ and all the ciphertexts in $\mathcal{L}_0$ verify.
For $i = 1$ to m, the mixer $\mathcal{P}_{M_i}$ waits for the $\mathcal{P}_{M_{i-1}}$ to complete and does:

1. Sample a permutation $\tau_i \leftarrow_{\$} \mathcal{S}_n$;
2. Read from the BB the message $(\mathcal{L}_{i-1}, \pi^{\mathtt{mx}}_{i-1})$ posted by $\mathcal{P}_{M_{i-1}}$ (or read $\mathcal{L}_0$ if this is the first mixer), and parse $\mathcal{L}_{i-1} = \langle \mathtt{C}_{i-1,j} \rangle_{j \in [n]}$;
3. Build the list $\mathcal{L}_i \leftarrow \langle \mathtt{C}_{i,j} \rangle_{j \in [n]}$ of shuffled and re-randomized ciphertexts by sampling randomness $\mathbf{r}_j, r_j$ and computing $\mathtt{C}_{i,\tau_i(j)} \leftarrow \mathsf{Rand}(\mathsf{pk}, \mathtt{C}_{i-1,j}; \mathbf{r}_j, r_j)$.
4. Compute a NIZK proof $\pi^{\mathtt{mx}}_i \leftarrow_{\$} \mathsf{P}_{\mathtt{mx}}(\mathsf{crs}^i_{\mathtt{mx}}, ([\mathbf{P}], \mathbf{CkSum}(\mathcal{L}_{i-1}, \mathcal{L}_i)), \sum_j \mathbf{r}_j)$,
5. Post in the BB the tuple $(\mathcal{L}_i, \pi^{\mathtt{mx}}_i)$

Verification. Once all mixers are done, every mixer $\mathcal{P}_{M_i}$ executes:

1. Read the messages $(\mathcal{L}_i, \pi^{\mathtt{mx}}_i)$ posted by every mixer on the BB, as well as the messages $(\mathtt{C}_{0,j}, \pi^{\mathtt{sd}}_j)$ posted by the senders;
2. For all $i \in [m]$ and for all $j \in [n]$ check that $\mathsf{Ver}(\mathsf{pk}, \mathtt{C}_{i,j}) = 1$;
3. For all $i \in [m]$, check $\mathsf{V}_{\mathtt{mx}}(\mathsf{crs}^i_{\mathtt{mx}}, ([\mathbf{P}], \mathbf{CkSum}(\mathcal{L}_{i-1}, \mathcal{L}_i)), \pi^{\mathtt{mx}}_i) = 1$;
4. If one of the checks does not verify abort and write `invalid` in the BB.

[10] We could modify our protocol to let the mixers share the same CRS, at the price of requiring $\mathcal{NIZK}_{\mathtt{mx}}$ be simulation sound. Since in most applications the number of mixers is small, we go for the simpler option of one crs per mixer.

Decrypt. All the mixers $\mathcal{P}_{M_i}$ execute the following in parallel (using the ideal functionality $\mathcal{F}_{\mathsf{TDec}}$ to compute decryptions):

1. let $\mathcal{L}_m = \langle \mathtt{C}^*_j \rangle_{j\in[n]}$ be the list of ciphertexts returned by the last mixer. For $j = 1$ to n, ask $\mathcal{F}_{\mathsf{TDec}}$ for the decryption of $\mathtt{C}^*_j$. Once all the mixers agreed on the decryption, receive $\mathtt{M}_j \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathtt{C}^*_j)$ from the functionality;
2. Post $\mathsf{Sort}(\langle \mathtt{M}_j \rangle_{j\in[n]})$ on the BB.

Audit Message. The mixers $\mathcal{P}_{M_i}$ post the message `valid` on the BB.

Algorithm Audit: the algorithm reads from the BB and computes the verification step of the protocol above (notice that this only relies on public information).

Theorem 5. *The auditable protocol* (Π, Audit) *described above* $\mathsf{UC\text{-}realizes}$ $\mathcal{F}_{\mathsf{Mix}}$ *with setup assumptions* $\mathcal{F}_{\mathsf{TDec}}$ *and* $\mathcal{F}_{\mathsf{CRS}}$.

Proof (Sketch.). We prove the theorem via a sequence of hybrid experiments. In the last experiment we define a simulator and highlight its interaction with the ideal functionality.

In the proof, we let h^* be the index of the first honest mixer. Also, we consider two sets Ψ_{in} and Ψ_{hide}, both consisting of tuples $(X, Y) \in \mathbb{G}_1^2$. For Ψ_{in} (resp. Ψ_{hide}) we define a corresponding *map* $\psi_{\mathsf{in}} : \mathbb{G}_1 \to \mathbb{G}_1$ (resp. ψ_{hide}) such that $\psi_{\mathsf{in}}(X)$ (resp. $\psi_{\mathsf{hide}}(X)$) is equal to Y if $(X, Y) \in \Psi_{\mathsf{in}}$ (resp. $(X, Y) \in \Psi_{\mathsf{hide}}$), otherwise X. We assume that all the NIZK proofs verify and that all the ciphertexts verify (as otherwise the protocol would abort without producing any output).

For space reason, in this proof sketch, we group together the hybrid experiments according to their function in the overall strategy.

Hybrids $\mathbf{H}_1$ to $\mathbf{H}_3$: In the first step we program the CRSs of both the NIZKs so that we can simulate the proof of the h^*-th mixer and of all the senders but one corrupted sender (whose index is hidden to the adversary by the CRS indistinguishability). For this step we can use the zero-knowledge property of the NIZKs. In the second and third step we use perfect-rerandomizability and RCCA security to introduce a change in the output of the h^*-th mixer. Specifically, the mixer $\mathcal{P}_{M_{h^*}}$ outputs ciphertexts which are fresh encryptions of random and independent messages $\mathtt{H}_1, \ldots, \mathtt{H}_n$. Moreover, we populate the set Ψ_{hide} with the pairs $(\mathtt{M}_{h^*-1,j}, \mathtt{H}_j)_{j\in[n]}$ to associate $\mathtt{H}_j$ with $\mathtt{M}_{h^*-1,j} \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathtt{C}_{h^*-1,j})$, and then we simulate the ideal functionality $\mathcal{F}_{\mathsf{TDec}}$ to output $\Psi_{\mathsf{hide}}(\mathtt{M})$ instead of $\mathtt{M}$. This way the modification is not visible by looking at the decrypted ciphertexts.

Hybrid $\mathbf{H}_4$: Let $\mathcal{V}_m$ (resp. $\mathcal{V}_{h^*}$) be the decryption of the list of ciphertexts output by the last mixer $\mathcal{P}_{M_m}$ (resp. by the first honest mixer $\mathcal{P}_{M_{h^*}}$). The hybrid $\mathbf{H}_4$ aborts if $\mathcal{V}_m \neq \mathcal{V}_{h^*}$. Using the perfect adaptive soundness of $\mathcal{NIZK}_{\mathtt{mx}}$ and the RCCA security and the public-verifiability of our PKE, we can show that this abort can happen only with negligible probability. We adapt the security argument of Faonio and Fiore [16] to our pv-Rand-PKE and our NIZK proof of "checksum". The idea is that the proofs of checksum $\pi^{\mathtt{mx}}_{h^*+1}, \ldots, \pi^{\mathtt{mx}}_n$ establish a linear relationship between the plaintexts encrypted in the list of ciphertexts output by $\mathcal{P}_{M_{h^*}}$ and the plaintexts in the list of ciphertext output by $\mathcal{P}_{M_m}$. The

reduction to RCCA security can install a challenge ciphertext in the first list and then learn information about the underlying plaintext by decrypting the second list. The idea is that the condition $\mathcal{V}_m \neq \mathcal{V}_{h^*}$ guarantees that the RCCA decryption oracle would not answer $\diamond$ on ciphertexts from the second list, and the linear relationship guaranteed by the proofs allows to extract the information on the challenge ciphertext.

Hybrid $\mathbf{H}_5$: Simulate the ideal functionality $\mathcal{F}_{\mathsf{TDec}}$ in different way. Whenever the mixers agree on the decryption of a ciphertext $\mathtt{C} \in \mathcal{L}_m$, simulate the functionality $\mathcal{F}_{\mathsf{TDec}}$ by outputting a message chosen uniformly at random (without re-introduction) from the list $\mathcal{V}_{h^*-1}$. Notice, we don't need to compile the list Ψ_{hide} anymore as the mixers would only agree to decrypt ciphertexts from the last list $\mathcal{L}_m$ and $\mathcal{V}_m = \mathcal{V}_{h^*} = \Psi_{\mathsf{hide}}(\mathcal{V}_{h^*-1})$.

We can prove that $\mathbf{H}_5$ and $\mathbf{H}_4$ are identically distributed. In fact in $\mathbf{H}_4$, after the first honest mixer outputs $\mathcal{L}_{h^*}$, an unbounded environment $\mathcal{Z}$ knows that in Ψ_{hide} the element $\mathtt{H}_j$ for $j \in [n]$ is mapped to some other value in $\mathcal{V}_{h^*-1}$ but, from its view, it cannot know to which value. Such information is revealed only during decryption time. In other words, we could sample the permutation τ_{h^*} (uniformly at random) at decryption time.

It is easy to check that, at this point of the hybrid argument, the list of ciphertexts received by the first honest mixers is (a permutation of) the output of the protocol. Moreover, the ordering of the ciphertexts in the former list and in the latter list are uncorrelated. With the next hybrids we make sure that the inputs of the honest senders are not discarded along the way from the first mixer to first honest mixer.

Hybrids $\mathbf{H}_6$ to $\mathbf{H}_7$: Notice that at this point the output of the mix-net is already distributed uniformly over the set of all the possible permutations of the inputs. However, the input messages of the honest senders are still (at least information theoretically) in the view of the adversary, as the honest senders still encrypt their inputs. In the next hybrids we switch into a hybrid experiment where all the honest senders encrypt dummy messages from a set $\mathcal{M}_H$, that we call the set of honest simulated messages. To do so we first program the map ψ_{in} to map the simulated messages to the (real) honest ones, and we simulate the functionality $\mathcal{F}_{\mathsf{TDec}}$ to pick messages $\mathtt{M}$ chosen uniformly at random (without re-introduction) from the list $\mathcal{V}_{h^*-1}$ and return $\psi_{\mathsf{in}}(\mathtt{M})$ instead of $\mathtt{M}$. Then in the second step we switch and encrypt the simulated messages, relying on RCCA security.

Hybrid $\mathbf{H}_8$ to $\mathbf{H}_9$: In the last two hybrids we make sure that (1) the malicious senders do not copy the ciphertexts of the honest senders, for this step we rely on the ABO soundness of the $\mathcal{NIZK}_{\mathtt{sd}}$ proof system, and (2) the malicious mixers do not duplicate or remove the messages of the honest senders, this argument is almost the same as in the step $\mathbf{H}_4$.

We can proceed to present the simulator S. For space reason, here we describe the most important parts.

Extraction of the Inputs: Let $\mathcal{L}_{h^*-1}$ be the list produced by the malicious mixer $\mathcal{P}_{M_{h^*-1}}$. For any j, the simulator S decrypts $\hat{\mathtt{M}}_j \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathtt{C}_{h^*-1,j})$ and if $\hat{\mathtt{M}}_j \notin \mathcal{M}_H$ then it submits it as input to the ideal functionality $\mathcal{F}_{\mathsf{Mix}}$.

Decryption Phase: The simulator S receives from the ideal functionality $\mathcal{F}_{\mathsf{Mix}}$ the sorted output $\langle \mathtt{M}_1^o, \ldots, \mathtt{M}_n^o \rangle$. Whenever the mixers agree on the decryption of a ciphertext, it simulates the ideal functionality $\mathcal{F}_{\mathsf{TDec}}$ by outputting a message from the sorted output randomly chosen (without reinsertion).

We notice that the hybrid compiles the map ψ_{in} by setting a correspondence between the inputs of the honest senders and the simulated ones, and, during the decryption phase, uses the map ψ_{in} to revert this correspondence. On the other hand, the simulator does not explicitly set the map, as it does not know the inputs of the honest senders (which are sent directly to the functionality). However, at inputs submission phase the simulator picks a simulated input for any honest sender, and at decryption phase it picks a message from the ordered list in output, which contains the inputs of the honest senders. By doing so, the simulator is implicitly defining the map ψ_{in}. The second difference is that the simulator picks the outputs from the list $\langle \mathtt{M}_1^o, \ldots, \mathtt{M}_n^o \rangle$ while the hybrid $\mathbf{H}_9$ uses the list $\psi_{\mathsf{in}}(\mathcal{V}_{h^*-1})$. However, recall that the simulator extracts the corrupted inputs from the same list $\mathcal{V}_{h^*-1}$, and that, by the change introduced in $\mathbf{H}_9$, we are assured that all the inputs of the honest senders will be in the list $\psi_{\mathsf{in}}(\mathcal{V}_{h^*-1})$.

References

1. S. Bayer and J. Groth. Efficient zero-knowledge argument for correctness of a shuffle. In *EUROCRYPT 2012*, volume 7237, pages 263–280. Springer, Heidelberg, 2012.
2. M. Bellare, K. G. Paterson, and P. Rogaway. Security of symmetric encryption against mass surveillance. In *CRYPTO 2014, Part I*, volume 8616, pages 1–19. Springer, Heidelberg, 2014.
3. J. Camenisch, K. Haralambiev, M. Kohlweiss, J. Lapon, and V. Naessens. Structure preserving CCA secure encryption and applications. In *ASIACRYPT 2011*, volume 7073, pages 89–106. Springer, Heidelberg, 2011.
4. R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *42nd FOCS*, pages 136–145. IEEE Computer Society Press, 2001.
5. R. Canetti and S. Goldwasser. An efficient threshold public key cryptosystem secure against adaptive chosen ciphertext attack. In *EUROCRYPT'99*, volume 1592, pages 90–106. Springer, Heidelberg, 1999.
6. R. Canetti, S. Halevi, and J. Katz. Chosen-ciphertext security from identity-based encryption. In *EUROCRYPT 2004*, volume 3027, pages 207–222. Springer, Heidelberg, 2004.
7. R. Canetti, H. Krawczyk, and J. B. Nielsen. Relaxing chosen-ciphertext security. In *CRYPTO 2003*, volume 2729, pages 565–582. Springer, Heidelberg, 2003.
8. M. Chase, M. Kohlweiss, A. Lysyanskaya, and S. Meiklejohn. Malleable proof systems and applications. In *EUROCRYPT 2012*, volume 7237, pages 281–300. Springer, Heidelberg, 2012.
9. R. Chen, Y. Mu, G. Yang, W. Susilo, F. Guo, and M. Zhang. Cryptographic reverse firewall via malleable smooth projective hash functions. In *ASIACRYPT 2016, Part I*, volume 10031, pages 844–876. Springer, Heidelberg, 2016.

10. R. Cramer and V. Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In *EUROCRYPT 2002*, volume 2332, pages 45–64. Springer, Heidelberg, 2002.
11. I. Damgård, S. Faust, P. Mukherjee, and D. Venturi. Bounded tamper resilience: How to go beyond the algebraic barrier. In *ASIACRYPT 2013, Part II*, volume 8270, pages 140–160. Springer, Heidelberg, 2013.
12. Y. Dodis, K. Haralambiev, A. López-Alt, and D. Wichs. Efficient public-key cryptography in the presence of key leakage. In *ASIACRYPT 2010*, volume 6477, pages 613–631. Springer, Heidelberg, 2010.
13. Y. Dodis, I. Mironov, and N. Stephens-Davidowitz. Message transmission with reverse firewalls—secure communication on corrupted machines. In *CRYPTO 2016, Part I*, volume 9814, pages 341–372. Springer, Heidelberg, 2016.
14. T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *CRYPTO'84*, volume 196, pages 10–18. Springer, Heidelberg, 1984.
15. A. Escala, G. Herold, E. Kiltz, C. Ràfols, and J. Villar. An algebraic framework for Diffie-Hellman assumptions. In *CRYPTO 2013, Part II*, volume 8043, pages 129–147. Springer, Heidelberg, 2013.
16. A. Faonio and D. Fiore. Optimistic mixing, revisited. Cryptology ePrint Archive, Report 2018/864, 2018. https://eprint.iacr.org/2018/864.
17. A. Faonio, D. Fiore, J. Herranz, and C. Ràfols. Structure-preserving and re-randomizable rcca-secure public key encryption and its applications. Cryptology ePrint Archive, Report 2019/955, 2019. https://eprint.iacr.org/2019/955.
18. A. Faonio and D. Venturi. Efficient public-key cryptography with bounded leakage and tamper resilience. In *ASIACRYPT 2016, Part I*, volume 10031, pages 877–907. Springer, Heidelberg, 2016.
19. P. Fauzi, H. Lipmaa, J. Siim, and M. Zajac. An efficient pairing-based shuffle argument. In *ASIACRYPT 2017, Part II*, volume 10625, pages 97–127. Springer, Heidelberg, 2017.
20. S. Garg, A. Jain, and A. Sahai. Leakage-resilient zero knowledge. In *CRYPTO 2011*, volume 6841, pages 297–315. Springer, Heidelberg, 2011.
21. R. Gennaro and Y. Lindell. A framework for password-based authenticated key exchange. In *EUROCRYPT 2003*, volume 2656, pages 524–543. Springer, Heidelberg, 2003. http://eprint.iacr.org/2003/032.ps.gz.
22. P. Golle, M. Jakobsson, A. Juels, and P. F. Syverson. Universal re-encryption for mixnets. In *CT-RSA 2004*, volume 2964, pages 163–178. Springer, Heidelberg, 2004.
23. P. Golle, S. Zhong, D. Boneh, M. Jakobsson, and A. Juels. Optimistic mixing for exit-polls. In *ASIACRYPT 2002*, volume 2501, pages 451–465. Springer, Heidelberg, 2002.
24. J. Groth. Rerandomizable and replayable adaptive chosen ciphertext attack secure cryptosystems. In *TCC 2004*, volume 2951, pages 152–170. Springer, Heidelberg, 2004.
25. J. Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In *ASIACRYPT 2006*, volume 4284, pages 444–459. Springer, Heidelberg, 2006.
26. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *EUROCRYPT 2008*, volume 4965, pages 415–432. Springer, Heidelberg, 2008.
27. G. Herold, J. Hesse, D. Hofheinz, C. Ràfols, and A. Rupp. Polynomial spaces: A new framework for composite-to-prime-order transformations. In *CRYPTO 2014, Part I*, volume 8616, pages 261–279. Springer, Heidelberg, 2014.

28. G. Herold, M. Hoffmann, M. Klooß, C. Ràfols, and A. Rupp. New techniques for structural batch verification in bilinear groups with applications to groth-sahai proofs. In *ACM CCS 17*, pages 1547–1564. ACM Press, 2017.
29. C. S. Jutla and A. Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In *ASIACRYPT 2013, Part I*, volume 8269, pages 1–20. Springer, Heidelberg, 2013.
30. C. S. Jutla and A. Roy. Switching lemma for bilinear tests and constant-size NIZK proofs for linear subspaces. In *CRYPTO 2014, Part II*, volume 8617, pages 295–312. Springer, Heidelberg, 2014.
31. J. Katz and V. Vaikuntanathan. Round-optimal password-based authenticated key exchange. In *TCC 2011*, volume 6597, pages 293–310. Springer, Heidelberg, 2011.
32. E. Kiltz. Chosen-ciphertext security from tag-based encryption. In *TCC 2006*, volume 3876, pages 581–600. Springer, Heidelberg, 2006.
33. E. Kiltz and H. Wee. Quasi-adaptive NIZK for linear subspaces revisited. In *EUROCRYPT 2015, Part II*, volume 9057, pages 101–128. Springer, Heidelberg, 2015.
34. B. Libert, T. Peters, and C. Qian. Structure-preserving chosen-ciphertext security with shorter verifiable ciphertexts. In *PKC 2017, Part I*, volume 10174, pages 247–276. Springer, Heidelberg, 2017.
35. S. Micali, C. Rackoff, and B. Sloan. The notion of security for probabilistic cryptosystems. In *CRYPTO'86*, volume 263, pages 381–392. Springer, Heidelberg, 1987.
36. I. Mironov and N. Stephens-Davidowitz. Cryptographic reverse firewalls. In *EUROCRYPT 2015, Part II*, volume 9057, pages 657–686. Springer, Heidelberg, 2015.
37. C. Namprempre, P. Rogaway, and T. Shrimpton. Reconsidering generic composition. In *EUROCRYPT 2014*, volume 8441, pages 257–274. Springer, Heidelberg, 2014.
38. M. Naveed, S. Agrawal, M. Prabhakaran, X. Wang, E. Ayday, J.-P. Hubaux, and C. A. Gunter. Controlled functional encryption. In *ACM CCS 14*, pages 1280–1291. ACM Press, 2014.
39. O. Pereira and R. L. Rivest. Marked mix-nets. In *FC 2017 Workshops*, volume 10323, pages 353–369. Springer, Heidelberg, 2017.
40. D. H. Phan and D. Pointcheval. OAEP 3-round: A generic and secure asymmetric encryption padding. In *ASIACRYPT 2004*, volume 3329, pages 63–77. Springer, Heidelberg, 2004.
41. M. Prabhakaran and M. Rosulek. Rerandomizable RCCA encryption. In *CRYPTO 2007*, volume 4622, pages 517–534. Springer, Heidelberg, 2007.
42. A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *40th FOCS*, pages 543–553. IEEE Computer Society Press, 1999.
43. V. Shoup and R. Gennaro. Securing threshold cryptosystems against chosen ciphertext attack. In *EUROCRYPT'98*, volume 1403, pages 1–16. Springer, Heidelberg, 1998.
44. D. Wikström. A universally composable mix-net. In *TCC 2004*, volume 2951, pages 317–335. Springer, Heidelberg, 2004.
45. D. Wikström. A sender verifiable mix-net and a new proof of a shuffle. In *ASIACRYPT 2005*, volume 3788, pages 273–292. Springer, Heidelberg, 2005.
46. D. Wikström. Verificatum, 2010. https://www.verificatum.com.