A preliminary version of this paper is published in the proceedings of the 10th Conference on Security and Cryptography for Networks [BMV16]—SCN 2016. This is the full version.

Naor-Yung Paradigm with Shared Randomness and Applications[*]

Silvio Biagioni[1], Daniel Masny[2], and Daniele Venturi[3]

[1] *University of Trento, Italy*
[2] *University of California, Berkeley*
[3] *Sapienza University of Rome, Italy*

June 25, 2017

Abstract

The Naor-Yung paradigm (Naor and Yung, STOC '90) allows to generically boost security under chosen-plaintext attacks (CPA) to security against chosen-ciphertext attacks (CCA) for public-key encryption (PKE) schemes. The main idea is to encrypt the plaintext twice (under independent public keys), and to append a non-interactive zero-knowledge (NIZK) proof that the two ciphertexts indeed encrypt the same message. Later work by Camenisch, Chandran, and Shoup (Eurocrypt '09) and Naor and Segev (Crypto '09 and SIAM J. Comput. '12) established that the very same techniques can also be used in the settings of key-dependent message (KDM) and key-leakage attacks (respectively).

In this paper we study the conditions under which the two ciphertexts in the Naor-Yung construction can share the same random coins. We find that this is possible, provided that the underlying PKE scheme meets an additional simple property. The motivation for reusing the same random coins is that this allows to design much more efficient NIZK proofs. We showcase such an improvement in the random oracle model, under standard complexity assumptions including Decisional Diffie-Hellman, Quadratic Residuosity, and Subset Sum. The length of the resulting ciphertexts is reduced by 50%, yielding truly efficient PKE schemes achieving CCA security under KDM and key-leakage attacks.

As an additional contribution, we design the first PKE scheme whose CPA security under KDM attacks can be directly reduced to (low-density instances of) the Subset Sum assumption. The scheme supports key-dependent messages computed via any affine function of the secret key.

[*] Work partially done while the authors were, respectively, with Sapienza University of Rome, Ruhr-Universität Bochum, and University of Trento.

Contents

1 Introduction 1
1.1 Our Contributions 3
1.2 Related Work 4

2 Preliminaries 5
2.1 Notation 5
2.2 Public-Key Encryption 5
2.3 Non-Interactive Zero-Knowledge Argument Systems 7

3 Naor-Yung Paradigm with Shared Randomness 8
3.1 A Twist of Naor-Yung 8
3.2 Randomness Fusion 8
3.3 Main Theorem: KDM Security . 9
3.4 Main Theorem: Key-Leakage Security 14

4 KDM Security from Subset Sum 19
4.1 The Subset Sum Problem 19
4.2 Scheme Description 21
4.3 Proof of Correctness 21
4.4 Proof of Security 22

5 Concrete Instantiations and Comparisons 25
5.1 Instantiantion from Decisional Diffie-Hellman 26
5.2 Instantiation from Quadratic Residuosity 27
5.3 Instantiation from Subset Sum . 28

6 Conclusion and Open Problems 31

1 Introduction

Forty years ago, in their seminal paper [DH76], Diffie and Hellman put forward the concept of public-key cryptography. Since then, the field has experienced huge advances, making public-key encryption (PKE) one of the most fundamental and deployed cryptographic applications. Intuitively, PKE allows a sender to encrypt a message under a receiver's public key; the receiver holding the corresponding secret key is the only one able to decrypt the resulting ciphertext, and thus recover the transmitted message.

Since PKE is ubiquitously employed, there are many different applications demanding different notions of security. An obvious requirement is that it should be unfeasible to recover the plaintext behind a given ciphertext. This is, however, far from being sufficient, as it does not exclude, e.g., the possibility that one is able to recover parts of the encrypted message or to recover the message of a different ciphertext. Seminal work on the subject [Yao82, GM84, MRS88] established the equivalence of different formulations leading to the following minimal requirement: No efficient adversary, given a target public key, should be able to distinguish the encryption of two messages of his choice, which are called the challenge ciphertexts. This notion is often known under the name of indistinguishability under *chosen-plaintext* attacks (CPA), and it is by far the most basic security requirement a PKE scheme needs to meet.

Unfortunately, CPA security does not consider adversaries able to alter an encrypted message, making this notion unsuitable for applications like, e.g., online auctions, where an adversary should be unable to slightly increase the offer of a previous bidder by altering his encrypted offer. In such settings ciphertexts must be *non-malleable* [DDN91, BDPR98, PSV07, CDTV16], meaning that it be hard, given a ciphertext encrypting some message, to create a valid ciphertext encrypting a related message.

The de-facto standard notion of security for PKE, which implies both CPA security and non-malleability, is called indistinguishability under *chosen-ciphertext* attacks (CCA). CCA security requires that CPA security should hold even in the presence of decryption queries, i.e., the adversary is allowed to ask for the decryption of arbitrary (possibly altered) ciphertexts (except the challenge ciphertexts itself). The famous attack on PKCS #1 by Bleichenbacher [Ble98] has shown that CCA security is not only a theoretical matter, and this is particularly true for a fundamental and widely employed primitive such as PKE.

KDM security. In fact, even CCA security is not enough for all possible applications of PKE. In a key-dependent message (KDM) attack, an adversary might see (or generate) ciphertexts encrypting messages related to the secret key. This is the case, e.g., in disk encryption software (including Windows Vista's BitLocker utility) and in certain anonymous credential systems [CL01], or could be due to careless key management. Further, to apply techniques like bootstrapping [Gen09] or key-switching [BGV12], in order to reduce the noise and ciphertext growth in fully-homomorphic encryption, it is necessary to publish a ciphertext which encrypts its own secret key. The notions of CPA/CCA security can be extended to the KDM setting by requiring that encryptions of messages depending on the secret key (via adversarial functions) are indistinguishable from encryptions of a fixed string [BRS02, CL01].

Several PKE schemes with CPA/CCA-KDM security exist, under different complexity assumptions including Learning with Errors [ACPS09, BGK11], Decisional Diffie-Hellman [BHHO08, BGK11, Hof13, Wee16], Quadratic Residuosity [BG10, Hof13, Wee16], Learning Parity with Noise [Döt15], and indistinguishability obfuscation and one-way functions [MPS16]. A weaker security guarantee, where there is an a-priori upper bound on the total number of adversarial KDM queries, was put forward by Galindo, Herranz and Villar [GHV12].

Key-leakage security. Key-leakage attacks are another example of a setting in which standard CPA/CCA security is not enough. Such attacks model an adversary having access to a piece of hardware implementing a cryptographic primitive, and trying to learn learn partial information on the secret key by means of so-called side-channel attacks, exploiting physical phenomena such as timing [Koc96], power consumption [KJJ99], and electronic emission [QS01]. Such (bounded) key-leakage attacks can be captured within the notions of CPA/CCA security by empowering the adversary with access to a so-called Λ-leakage oracle: Upon input an efficiently computable function, the oracle returns the result of the function applied to the secret key, for a total of at most Λ bits.

Several PKE schemes with CPA/CCA security under bounded key-leakage attacks exist, under different complexity assumptions including Learning with Errors [AGV09], Decisional Diffie-Hellman [NS09, DHLW10, NS12], and Quadratic Residuosity [BG10].

Naor-Yung paradigm. The Naor-Yung paradigm is a method to generically transform a CPA-secure PKE scheme into a CCA-secure one, in a non-black-box way. Specifically, to encrypt a given message m, one samples two independent public keys pk and pk' for the underlying CPA-secure PKE, encrypts the message m twice yielding ciphertexts c and c' (the first one under pk and the second one under pk'), and finally gives a non-interactive proof π that the ciphertexts indeed encrypt the same message. One can show that if the non-interactive proof satisfies zero-knowledge, and moreover it is simulation-sound [Sah99], the resulting PKE scheme meets CCA security.

Later work by Camenisch, Chandran, and Shoup [CCS09], and by Naor and Segev [NS09, NS12], showed that the original Naor-Yung paradigm also works in the more generic settings of KDM attacks and key-leakage attacks. However, in the standard model, the resulting PKE scheme is not very efficient due to the cost of simulation-sound non-interactive zero-knowledge (NIZK) proof systems. Since efficiency is crucial in many applications, and also one of the main obstacles for employing cryptographic primitives in the real world, a common approach to improve efficiency is to consider the random oracle model (ROM) due to Bellare and Rogaway [BR93]. A cryptographic scheme in the ROM employs a hash function that is modelled as an external random function accessible by all parties (including the adversary). While random oracles offer only heuristic security, in the sense that in some cases it might be even impossible to implement them in practice [CGH04], they allow for dramatic efficiency improvements, and in fact, many cryptographic primitives implemented in nowadays standards are analyzed in the ROM.

NIZKs are not an exception in this respect, and indeed, as shown by Faust *et al.* [FKMV12], the powerful Fiat-Shamir heuristic [FS86] allows to instantiate the NIZK in the Naor-Yung construction in the ROM, yielding the most efficient realizations of PKE schemes with CCA security under KDM and key-leakage attacks known today.

1.1 Our Contributions

In this paper, we contribute to the line of research on secure PKE in the following ways.

A twist of Naor-Yung. We analyze a slight modification of the original Naor-Yung paradigm. The main idea is to have the two ciphertexts c and c' share the same random coins. As we will see, this allows for a substantial efficiency improvement in the design of the NIZK, yielding beyond state-of-the-art PKE schemes with CCA security under KDM and key-leakage attacks (in the ROM).

Our analysis (see Section 3) shows that the above idea indeed works, provided that the underlying CPA-secure PKE scheme meets an additional property that we dub “randomness fusion”: Given two ciphertexts c and c' of messages m and m' respectively (computed under independent public keys pk and pk') it is possible to re-randomize (c, c') into a new pair $(\tilde{c}, \tilde{c}')$ such that the distribution of $(\tilde{c}, \tilde{c}')$ is statistically close to the distribution of $(\hat{c}, \hat{c}')$, where $(\hat{c}, \hat{c}')$ are computed using the normal encryption with the same (uniform) randomness r^*.

A similar requirement has been put forward by Bellare *et al.* [BBS03] in their study of randomness re-use in multi-recipient PKE. As we will argue later when defining our requirement in more detail, randomness fusion is weaker than the property defined in [BBS03], but still suffices for our application.

KDM security from Subset Sum. As a contribution of independent interest, in Section 4, we design the first PKE scheme whose KDM-CPA security can be based directly on low-density instances of the Subset Sum problem. Such an assumption is particularly interesting given its robustness to quantum attacks [BJLM13]. The set of supported KDM functions consists of all possible (efficiently computable) affine modifications of the secret key; a result of Applebaum [App11, App14] allows to generically boost this form of KDM security to security against all functions that can be computed in some fixed polynomial time.

Our construction borrows ideas from Applebaum *et al.* [ACPS09], that we needed to carefully adapt to the case of Subset Sum. The PKE scheme we design can be effectively used in our framework (as we argue below), yielding a truly efficient PKE scheme with KDM-CCA security from the Subset Sum assumption (in the ROM).

Comparison. Finally, we instantiate our twist of the Naor-Yung construction under three complexity assumptions: Decisional Diffie-Hellman, Quadratic Residuosity, and Subset Sum. As our analysis shows (see Section 5, Table 1), ciphertexts computed via our approach are shorter by a factor of roughly 50% compared to those one would obtain via the original Naor-Yung paradigm.

The reason behind such an efficiency improvement is best understood using an example. Consider the ElGamal PKE scheme [ElG85], whose CPA-security can be based on the Decisional

Diffie-Hellman assumption. A public key consists of a single element h, within a cyclic group $\mathbb{G}$ of prime order q (with generator g); an encryption of $m \in \mathbb{G}$ under h equals $c := (c_1, c_2) = (g^r, h^r \cdot m)$, for uniform randomness $r \in \mathbb{Z}_q$. The PKE scheme is easily seen to meet the randomness fusion property.[1]

When using the above PKE scheme in the original Naor-Yung construction one samples two independent public keys $h, h' \in \mathbb{G}$, and computes a "double encryption" of message m by defining $c := (c_1, c_2) = (g^r, h^r \cdot m)$ and $c' := (c'_1, c'_2) = (g^{r'}, (h')^{r'} \cdot m)$, for independent randomness $r, r' \in \mathbb{Z}_q$. Finally, one needs to compute a (simulation-sound) NIZK proof π for the fact that c and c' are well-distributed ciphertexts encrypting the same messages; this is equivalent to showing knowledge of r, r' such that $c_1 = g^r$, $c'_1 = g^{r'}$, and $c_2/c'_2 = h^r/(h')^{r'}$. We refer to the pair $x := (r, r')$ as the witness, and to $y := (h, (c_1, c_2), h', (c'_1, c'_2))$ as the statement to be proven.

The standard way to compute π (in the ROM) is by applying the Fiat-Shamir heuristic [FS86] to a so-called Sigma-protocol for the above considered language.[2] In the case of ElGamal (see [FKMV12, Section 5]) $\pi := (\alpha, \gamma)$, where $\alpha := (\alpha_1, \alpha_2, \alpha_3) = (g^s, g^{s'}, h^s \cdot (h')^{s'})$ and $\gamma := (\gamma_1, \gamma_2) = (s - \beta r, s' + \beta r')$, with random $s, s' \in \mathbb{Z}_q$ and β implicitly defined as $\beta := H(y||\alpha)$ through the application of the random oracle H. This way, a ciphertext consists of 9 group elements. Using our twist of the Naor-Yung construction one can completely drop α_2 and γ_2, thus saving 3 group elements (note that $c_1 = c'_1$). Hence, a ciphertext consists of 6 group elements, yielding a 33% gain in ciphertext size.

While the above instantiation is not interesting on its own right (as one can obtain CCA security in the standard model under the same complexity assumption, with even shorter ciphertexts [CS98]) it contains the crux of our method, and moreover it constitutes the base for understanding our concrete instantiations for KDM and key-leakage security in Section 5.

1.2 Related Work

The first PKE scheme with CPA security directly based on Subset Sum has been constructed by Lyubashevsky, Palacio, and Segev [LPS10]; their work has been extended to the setting of CCA security by Faust, Masny, and Venturi [FMV16]. Subset Sum also found application in the context of outsourced pattern matching [FHV13].

While we focused on public-key encryption, KDM security can also be defined in the secret-key setting. See, among others, [BRS02, ACPS09]. Sometimes KDM security is defined in a multi-key variant, where there are polynomially many public/secret key pairs, and the key-dependent message is chosen as a function of all the keys. Although our twist of the Naor-Yung paradigm works even in the multi-user setting, our Subset Sum based PKE scheme is only proven secure in the single-key setting.

Many definitions for security under key-leakage attacks exist in the literature, beyond the setting of bounded leakage considered in this paper. We refer the reader directly to the literature (e.g., [ADW09b, SPY$^+$10]) for a more in-depth discussion on the relevance of each definition. We also dispose of many leakage-resilient primitives beyond public-key encryption, see, among many others, [DP08, ADW09a, KV09, DDV10, BSW13, NVZ14, FNV15].

Rackoff and Simon [RS91] considered a variation of the Naor-Yung paradigm in which the sender encrypts the message only once, and then it proves in zero-knowledge that it knows the plaintext corresponding to the transmitted ciphertext. In order for this to work, the NIZK

[1] In fact, it satisfies the reproducibility test of Bellare *et al.* [BBS03] which implies the randomness fusion property.

[2] A Sigma-protocol is a public-coin interactive protocol consisting of three messages (α, β, γ), satisfying certain properties; see Section 5 for a more precise definition.

proof system needs to satisfy a stronger version of soundness known as simulation extractability. Unfortunately, this paradigm does not lead to very efficient instantiations in the ROM due to the fact that Fiat-Shamir NIZK are not known to be simulation extractable. (See [BFW15, BFW16] for negative indications on this matter.) An alternative (always in the ROM) is to use Fischlin's transformation [Fis05], but the price to instantiate the NIZK might be higher [DV14].

An alternative construction to generically boost CPA security to CCA security for PKE in the random oracle model is due to Fujisaki and Okamoto [FO99]. The security of this construction under KDM attacks has been analyzed by Kitagawa *et al.* [KMHT16].

2 Preliminaries

2.1 Notation

We write $\lambda \in \mathbb{N}$ for the security parameter. We say that a function ν is negligible in λ, if it is asymptotically smaller than the inverse of any polynomial in λ, i.e. $\nu(\lambda) = \lambda^{-\omega(1)}$. An algorithm A is probabilistic polynomial-time (PPT) if A is randomized, and for any input $x, r \in \{0,1\}^*$ the computation of $\mathsf{A}(x;r)$ (i.e., A with input x and random coins r) terminates in at most $\mathrm{poly}(|x|)$ steps. When the coins are left implicit, we write $y \leftarrow\!\!\$\ \mathsf{A}(x)$ to denote the output of $\mathsf{A}(x;r)$ with uniform randomness. If $\mathcal{X}$ is a set, then $x \leftarrow\!\!\$\ \mathcal{X}$ denotes that x is sampled uniformly at random from $\mathcal{X}$.

For a distribution $\mathbf{D}$, we denote with $x \leftarrow\!\!\$\ \mathbf{D}$ that x is sampled according to the distribution $\mathbf{D}$. For two distributions $\mathbf{D}$ and $\mathbf{D}'$ over a shared domain $\mathcal{D}$ we write $\mathbf{D}(x)$ for the probability assigned to $x \in \mathcal{D}$ and $\Delta(\mathbf{D}, \mathbf{D}') := \frac{1}{2}\sum_{x \in \mathcal{D}} |\mathbf{D}(x) - \mathbf{D}'(x)|$ for the statistical distance between $\mathbf{D}$ and $\mathbf{D}'$. Whenever the statistical distance is negligible, we write $\mathbf{D} \approx_s \mathbf{D}'$. Similarly, given two ensembles $\mathbf{X} = \{X_\lambda\}_{\lambda \in \mathbb{N}}$ and $\mathbf{Y} = \{Y_\lambda\}_{\lambda \in \mathbb{N}}$, we write $\mathbf{X} \approx_c \mathbf{Y}$ to denote that the two ensembles are computationally indistinguishable.

Vectors and matrices are denoted in boldface. For two vectors $\mathbf{u}, \mathbf{v}$, with $\mathbf{u} = (u_1, \ldots, u_n)$ and $\mathbf{v} = (v_1, \ldots, v_n)$, the inner product between $\mathbf{u}$ and $\mathbf{v}$ is defined as $\langle \mathbf{u}, \mathbf{v} \rangle := \sum_{i=1}^{n} u_i \cdot v_i$. The Hadamard product between $\mathbf{u}$ and $\mathbf{v}$ is defined as $\mathbf{u} \circ \mathbf{v} := (u_1 \cdot v_1, \ldots, u_n \cdot v_n)$.

We represent elements in $\mathbb{Z}_p$ as values in the range $[-(p-1)/2, (p-1)/2]$, where $p > 2$ is a prime number. The absolute value of $v \in \mathbb{Z}_p$, denoted $|v|$, is the absolute value of the corresponding value in $[-(p-1)/2, (p-1)/2]$, and the infinity norm of a vector $\mathbf{v} := (v_1, \ldots, v_n) \in \mathbb{Z}_p^n$ is $\|\mathbf{v}\|_\infty := \max_{i \in [n]} |v_i|$. We will also use the following rounding functions: $\lfloor \cdot \rceil : \mathbb{R} \to \mathbb{Z}$, which maps a real number to its closest integer; $\lfloor \cdot \rfloor : \mathbb{R} \to \mathbb{Z}$, which maps a real number to its closest smaller integer; and $\lceil \cdot \rceil : \mathbb{R} \to \mathbb{Z}$, which maps a real number to its closest larger integer. For any $q, p \in \mathbb{N}$, we denote by $\lfloor x \rceil_p : \mathbb{Z}_q \to \mathbb{Z}_p$ the rounding function $\lfloor x \rceil_p := \lfloor \frac{p}{q} \cdot x \rceil$; in case $\mathbf{v}$ is a vector, we write $\lfloor \mathbf{v} \rceil_p$ for the application of $\lfloor \cdot \rceil_p$ component wise.

2.2 Public-Key Encryption

A Public-Key Encryption (PKE) scheme is a tuple of algorithms $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ defined as follows. (1) Algorithm Gen takes as input the security parameter, and outputs a public/secret key pair (pk, sk); for a given value of the security parameter $\lambda \in \mathbb{N}$, the set of all secret keys is denoted by $\mathcal{SK}_\lambda$ and the set of all public keys by $\mathcal{PK}_\lambda$. (2) The randomized algorithm Enc takes as input the public key pk, a message $m \in \mathcal{M}$, and implicit randomness $r \in \mathcal{R}$, and outputs a ciphertext $c = \mathsf{Enc}(pk, m; r)$; the set of all ciphertexts is denoted by $\mathcal{C}$, and we sometimes write $\mu \in \mathbb{N}$ for the bit-length of a plaintext $m \in \mathcal{M}$. (3) The deterministic algorithm Dec takes as input the secret key sk and a ciphertext $c \in \mathcal{C}$, and outputs $m = \mathsf{Dec}(sk, c)$ which is either equal to some message $m \in \mathcal{M}$ or to an error symbol $\perp$.

```
Experiment $\mathbf{Exp}^{\text{kdm-cca}}_{\Pi,\mathsf{A}}(\lambda, \mathcal{F})$:
    $(pk, sk) \leftarrow\$ \mathsf{Gen}(1^\lambda)$; $b \leftarrow\$ \{0,1\}$
    $\mathcal{Q}_{\text{dec}}, \mathcal{Q}_{\text{kdm}} \leftarrow \emptyset$
    $b' \leftarrow \mathsf{A}^{\mathcal{O}^{\text{dec}}_{sk}(\cdot), \mathcal{O}^{\text{kdm}}_{sk,b}(\cdot)}(pk)$
    Return $(b' = b) \wedge (\mathcal{Q}_{\text{dec}} \cap \mathcal{Q}_{\text{kdm}} = \emptyset)$

Oracle $\mathcal{O}^{\text{kdm}}_{sk,b}(f)$:
    If $b = 0$
        Return $c \leftarrow\$ \mathsf{Enc}(pk, 0^\mu)$
    Else
        Return $c \leftarrow\$ \mathsf{Enc}(pk, f(sk))$
    $\mathcal{Q}_{\text{kdm}} \leftarrow \mathcal{Q}_{\text{kdm}} \cup \{c\}$

Oracle $\mathcal{O}^{\text{dec}}_{sk}(c)$:
    Return $\mathsf{Dec}(sk, c)$
    $\mathcal{Q}_{\text{dec}} \leftarrow \mathcal{Q}_{\text{dec}} \cup \{c\}$
```

Figure 1: Experiment defining KDM security of a PKE scheme.

Correctness. We say that Π satisfies *correctness* if, for all $(pk, sk) \leftarrow\$ \mathsf{Gen}(1^\lambda)$, there exists a negligible function $\nu : \mathbb{N} \to [0,1]$ such that that $\mathbb{P}[\mathsf{Dec}(sk, \mathsf{Enc}(pk, m)) = m] \geq 1 - \nu(\lambda)$ (where the randomness is taken over the internal coin tosses of algorithm Enc).

KDM security. We now turn to defining key-dependent message (KDM) security for PKE, both in the case of chosen-plaintext attacks (CPA) and chosen-ciphertext attacks (CCA).

Definition 1 (KDM security)**.** Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme with message space $\mathcal{M}$ and secret-key space $\mathcal{SK}_\lambda$ (for security parameter $\lambda \in \mathbb{N}$), and let $\mathcal{F} : \mathcal{SK}_\lambda \to \mathcal{M}$ be a set of efficiently computable functions. We say that Π has $\mathcal{F}$-key-dependent message security under chosen-ciphertext attacks ($\mathcal{F}$-KDM-CCA for short), if for all PPT adversaries A there exists a negligible function $\nu : \mathbb{N} \to [0,1]$ such that

$$\left| \mathbb{P}\left[\mathbf{Exp}^{\text{kdm-cca}}_{\Pi,\mathsf{A}}(\lambda, \mathcal{F}) = 1 \right] - \frac{1}{2} \right| \leq \nu(\lambda),$$

where the experiment $\mathbf{Exp}^{\text{kdm-cca}}_{\Pi,\mathsf{A}}(\lambda, \mathcal{F})$ is defined in Figure 1.

Moreover, we say that Π has $\mathcal{F}$-KDM-CPA security if the above holds for all PPT adversaries that are not allowed any query to oracle $\mathcal{O}^{\text{dec}}_{sk}(\cdot)$; in this case we denote by $\mathbf{Exp}^{\text{kdm-cpa}}_{\Pi,\mathsf{A}}(\lambda, \mathcal{F})$ the corresponding experiment.

We remark that $\mathcal{F}$-KDM-CPA security implies standard CPA security by considering the set $\mathcal{F}$ of all constant functions that output a given (hard-coded) plaintext in the message space, i.e. $\mathcal{F}_{\text{msg}} := \{f_m : f_m(\cdot) = m\}_{m \in \mathcal{M}}$.

Key-leakage security. Informally a PKE scheme is CPA-secure under Λ-key-leakage attacks if it remains CPA-secure even given Λ bits of (adaptive) leakage on the secret key [NS09, DHLW10]. CCA security under Λ-key-leakage attacks is defined similarly, but now the adversary can additionally ask for decryption queries.

Definition 2 (Key-leakage security)**.** Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme with message space $\mathcal{M}$ and secret-key space $\mathcal{SK}_\lambda$ (for security parameter $\lambda \in \mathbb{N}$), and let $\mathcal{O}^{\Lambda}_{sk}(\cdot)$ be an oracle depending on a secret key $sk \in \mathcal{SK}_\lambda$, which takes as input (the description of) functions $f_i : \mathcal{SK}_\lambda \to \{0,1\}^{\Lambda_i}$ and returns a total of at most $\sum_i \Lambda_i \leq \Lambda$ bits. We say that Π has Λ-key-leakage security under chosen-ciphertext attacks (Λ-LKG-CCA for short), if for all PPT adversaries A there exists a negligible function $\nu : \mathbb{N} \to [0,1]$ such that

$$\left| \mathbb{P}\left[\mathbf{Exp}^{\text{lkg-cca}}_{\Pi,\mathsf{A}}(\lambda, \Lambda) = 1 \right] - \frac{1}{2} \right| \leq \nu(\lambda),$$

where the experiment $\mathbf{Exp}^{\text{lkg-cca}}_{\Pi,\mathsf{A}}(\lambda, \Lambda)$ is defined in Figure 2. Π has Λ-LKG-CPA security if the above holds for all PPT adversaries that are not allowed any query to oracle $\mathcal{O}^{\text{dec}}_{sk}(\cdot)$.

```
Experiment $\mathbf{Exp}^{\text{lkg-cca}}_{\Pi,\mathsf{A}}(\lambda, \Lambda)$:
    $(pk, sk) \leftarrow\$ \mathsf{Gen}(1^\lambda)$; $b \leftarrow\$ \{0,1\}$
    $\mathcal{Q}_{\text{dec}} \leftarrow \emptyset$
    $(m_0, m_1) \leftarrow \mathsf{A}^{\mathcal{O}^{\text{dec}}_{sk}(\cdot), \mathcal{O}^{\Lambda}_{sk}(\cdot)}(pk)$
    $c_b \leftarrow \mathsf{Enc}(pk, m_b)$
    $b' \leftarrow \mathsf{A}^{\mathcal{O}^{\text{dec}}_{sk}(\cdot)}(c_b)$
    Return $(b' = b) \wedge (c_b \notin \mathcal{Q}_{\text{dec}})$
```

```
Oracle $\mathcal{O}^{\text{dec}}_{sk}(c)$:
    Return $\mathsf{Dec}(sk, c)$
    $\mathcal{Q}_{\text{dec}} \leftarrow \mathcal{Q}_{\text{dec}} \cup \{c\}$
```

```
Oracle $\mathcal{O}^{\Lambda}_{sk}(f)$:
    Return $f(sk)$
```

Figure 2: Experiment defining key-leakage security of a PKE scheme.

2.3 Non-Interactive Zero-Knowledge Argument Systems

A *decision problem* related to a language $L \subseteq \{0,1\}^*$ requires to determine if a given string y is in L or not. We can associate to any NP-language L a polynomial-time recognizable relation $R \subseteq \{0,1\}^* \times \{0,1\}^*$ defining L itself, i.e. $L = \{y : \exists x \text{ s.t. } (y, x) \in R\}$ for $|x| \leq \text{poly}(|y|)$. The string x is called a *witness* for membership of $y \in L$.

Let L be an NP-language. We now recall the definition of a non-interactive zero-knowledge (NIZK) argument system for L, in the random oracle model (ROM). Let H be a hash function (modeled as a random oracle). A non-interactive argument system for L is a pair of PPT algorithms $(\mathsf{P}^H, \mathsf{V}^H)$ specified as follows. (1) Algorithm P^H takes as input a pair (y, x) such that $(y, x) \in R$, and returns a proof π. (2) Algorithm V^H takes as input a pair (y, π) and returns a decision bit. We write P^H, V^H, to specify that both algorithms are allowed random oracle queries.

By correctness, we mean that $\mathsf{V}^H(y, \pi) = 1$ whenever $\pi \leftarrow\$ \mathsf{P}^H(y, x)$ and $(y, x) \in R$. Below, we define two further properties of non-interactive arguments, namely zero-knowledge and simulation soundness. The definitions are taken from [FKMV12].

Zero-knowledge. The zero-knowledge property captures the intuition that a non-interactive proof π for a given statement y does not reveal anything beyond the fact that $y \in L$. This intuition is formalized by the existence of an efficient simulator S that is able to simulate π without knowing a witness. Note that, as we make explicit in the definition below, the simulator is allowed to fully control the random oracle, a flavor that is sometimes known as zero-knowledge in the explicitly programmable random oracle model [Wee09].

Definition 3 (NIZK)**.** Let L be an NP-language, and let H be a hash function (modeled as a random oracle). Denote by $\mathsf{S}_1, \mathsf{S}_2$ the oracles such that $\mathsf{S}_1(\cdot)$ returns the first output of $(h, \tau) \leftarrow\$ \mathsf{S}(1, \tau, \cdot)$, and $\mathsf{S}_2(y, x)$ returns the first output of $(\pi, \tau) \leftarrow\$ \mathsf{S}(2, \tau, y)$ if $(y, x) \in R$, where $\tau \in \{0,1\}^*$ denotes some arbitrary state information shared between S_1 and S_2. We say that $(\mathsf{P}^H, \mathsf{V}^H)$ is a NIZK for L in the random oracle model if there exists a PPT simulator S such that, for all PPT distinguishers D, there is a negligible function $\nu : \mathbb{N} \to [0,1]$ for which

$$\left| \mathbb{P}\left[\mathsf{D}^{H(\cdot), \mathsf{P}^H(\cdot,\cdot)}(1^\lambda) = 1\right] - \mathbb{P}\left[\mathsf{D}^{\mathsf{S}_1(\cdot), \mathsf{S}_2(\cdot,\cdot)}(1^\lambda) = 1\right] \right| \leq \nu(\lambda),$$

where both oracles P and S_2 return $\perp$ in case $(y, x) \notin R$.

Simulation soundness. The simulation soundness property captures the intuition that it should be hard to find an accepting proof π for a false statement $y \notin L$, even after seeing polynomially many simulated proofs of possibly false statements.

Definition 4 (Simulation soundness)**.** Let L be an NP-language, and let H be a hash function (modeled as a random oracle). Consider a NIZK $(\mathsf{P}^H, \mathsf{V}^H)$ for L, with zero-knowledge simulator S. Denote by $\mathsf{S}_1, \mathsf{S}'_2$ the oracles such that $\mathsf{S}_1(\cdot)$ returns the first output of $(h, \tau) \leftarrow\!\!\$\ \mathsf{S}(1, \tau, \cdot)$, and $\mathsf{S}'_2(y)$ returns the first output of $(\pi, \tau) \leftarrow\!\!\$\ \mathsf{S}(2, \tau, y)$. We say that $(\mathsf{P}^H, \mathsf{V}^H)$ is simulation sound in the random oracle model if, for all PPT adversaries A, there is a negligible function $\nu : \mathbb{N} \to [0, 1]$ such that

$$\mathbb{P}\left[\mathsf{V}^{\mathsf{S}_1}(y^*, \pi^*) = 1 \wedge y^* \notin L \wedge (y^*, \pi^*) \notin \mathcal{Q} : \ (y^*, \pi^*) \leftarrow\!\!\$\ \mathsf{A}^{\mathsf{S}_1(\cdot), \mathsf{S}'_2(\cdot)}(1^\lambda)\right] \leq \nu(\lambda),$$

where $\mathcal{Q}$ contains the list of pairs (y_i, π_i) such that y_i was asked to S'_2 yielding answer π_i.

3 Naor-Yung Paradigm with Shared Randomness

We start by describing a twist of the Naor-Yung paradigm, in Section 3.1, where the same random string is used to generate both ciphertexts in the Naor-Yung construction. Then, in Section 3.2, we put forward a simple property of a PKE scheme which will be useful for proving security of the modified Naor-Yung paradigm. Our main theorem, and its proof, can be found in Section 3.3 (for the case of KDM security), and in Section 3.4 (for the case of key-leakage).

3.1 A Twist of Naor-Yung

The original Naor-Yung paradigm combines two CPA-secure PKE schemes Π and Π' into a new PKE scheme Π^* that achieves CCA security [NY90]. A ciphertext in Π^* consists of two independent encryptions of the same message (using fresh randomness), together with a non-interactive proof that the two ciphertexts indeed encrypt the same message. This paradigm was later extended to the setting of KDM security by Camenisch, Chandran and Shoup [CCS09], and to the setting of key-leakage by Naor and Segev [NS09, NS12].

Below, we present a twist of the Naor-Yung construction in which the two encryptions share the same random coins. As we will see in the sequel (cf. Section 5), this allows for significant efficiency improvements in the size of the resulting non-interactive proofs. Although our construction works for any pair of PKE schemes with shared message and randomness space, for simplicity we consider the special case in which $\Pi' = \Pi$.

Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme with message space $\mathcal{M}$ and randomness space $\mathcal{R}$, and let $(\mathsf{P}^H, \mathsf{V}^H)$ be a NIZK in the ROM for the following NP-language

$$L^{\Pi}_{\mathrm{NY}} := \left\{(pk, pk', c, c') : \exists m, r^* \text{ s.t. } c = \mathsf{Enc}(pk, m; r^*), c' = \mathsf{Enc}(pk', m; r^*)\right\}. \qquad (1)$$

The modified PKE scheme $\Pi^* = (\mathsf{Gen}^*, \mathsf{Enc}^*, \mathsf{Dec}^*)$ is described in Fig. 3.

3.2 Randomness Fusion

We now put forward a simple property of a PKE scheme Π which will be useful for proving security of the modified Naor-Yung construction. Informally, the property says that given two ciphertexts c and c' of messages m and m' respectively (computed under independent public keys pk and pk') it is possible to re-randomize (c, c') into a new pair $(\tilde{c}, \tilde{c}')$ such that the distribution of $(\tilde{c}, \tilde{c}')$ is statistically close to the distribution of $(\hat{c}, \hat{c}')$, where $(\hat{c}, \hat{c}')$ are computed using Enc with the same (uniform) random coins r^*.

Definition 5 (Randomness fusion)**.** Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme. We say that Π satisfies the randomness fusion property if there exists a PPT algorithm Rand such that, for

Naor-Yung Paradigm with Shared Randomness

Consider the following PKE scheme $\Pi^* = (\mathsf{Gen}^*, \mathsf{Enc}^*, \mathsf{Dec}^*)$ based on an auxiliary PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ and on a non-interactive argument system $(\mathsf{P}^H, \mathsf{V}^H)$ for the language L_{NY}^{Π} of Eq. (1).

Key generation: Given as input the security parameter λ, algorithm Gen^* runs Gen twice obtaining $(pk, sk) \leftarrow\$\ \mathsf{Gen}(1^\lambda)$ and $(pk', sk') \leftarrow\$\ \mathsf{Gen}(1^\lambda)$. Hence, it outputs $pk^* = (pk, pk')$ and $sk^* = sk$ (the key sk' is erased).

Encryption: Given as input a message $m \in \mathcal{M}$, algorithm Enc^* samples random coins $r^* \leftarrow\$\ \mathcal{R}$, computes $c = \mathsf{Enc}(pk, m; r^*)$ and $c' = \mathsf{Enc}(pk', m; r^*)$, and obtains a proof $\pi \leftarrow\$\ \mathsf{P}^H((pk, pk', c, c'), (m, r^*))$ for membership of $(pk, pk', c, c') \in L_{\text{NY}}^{\Pi}$. Hence, it outputs the ciphertext $c^* = (c, c', \pi)$.

Decryption: Given as input a ciphertext $c^* = (c, c', \pi)$, algorithm Dec^* first runs $\mathsf{V}^H((pk, pk', c, c'), \pi)$; if the output is zero Dec^* outputs $\perp$ and stops. Otherwise, it outputs the same as $\mathsf{Dec}(sk, c)$.

Figure 3: Modified Naor-Yung construction

all $m, m' \in \mathcal{M}$, it holds that $\mathbf{D}_{m,m'} \approx_s \tilde{\mathbf{D}}_{m,m'}$, where the distributions $\mathbf{D}_{m,m'}$ and $\tilde{\mathbf{D}}_{m,m'}$ are defined as follows:

$$\mathbf{D}_{m,m'} := \left\{ (\hat{c}, \hat{c}') : \begin{array}{c} (pk, sk) \leftarrow\$\ \mathsf{Gen}(1^\lambda); (pk', sk') \leftarrow\$\ \mathsf{Gen}(1^\lambda); r^* \leftarrow\$\ \mathcal{R} \\ \hat{c} = \mathsf{Enc}(pk, m; r^*); \hat{c}' = \mathsf{Enc}(pk', m'; r^*) \end{array} \right\} \tag{2}$$

$$\tilde{\mathbf{D}}_{m,m'} := \left\{ (\tilde{c}, \tilde{c}') : \begin{array}{c} (pk, sk) \leftarrow\$\ \mathsf{Gen}(1^\lambda); (pk', sk') \leftarrow\$\ \mathsf{Gen}(1^\lambda); r, r' \leftarrow\$\ \mathcal{R} \\ c = \mathsf{Enc}(pk, m; r); c' = \mathsf{Enc}(pk', m'; r') \\ \mathsf{aux} := (pk, pk', sk', r', m'); (\tilde{c}, \tilde{c}') \leftarrow\$\ \mathsf{Rand}((c, c'), \mathsf{aux}) \end{array} \right\}. \tag{3}$$

Although the above definition appears to be quite technical, the intuition behind it is pretty simple, and moreover many natural schemes, including the ones we consider in Section 5, directly meet this property, without requiring any modification. Looking ahead, randomness fusion is exactly the property one needs in order to analyse our twist of the Naor-Yung paradigm. In fact, readers familiar with the security proof of the original Naor-Yung construction might already observe that the particular choice of the auxiliary information is compliant with the information known by the reduction from CPA to CCA security.

Alternative formulations. A particular case is the one where the distribution of ciphertexts using independent randomness or shared randomness are directly statistically close. Such a requirement is more stringent, and can be cast in Definition 5 by requiring that Rand simply outputs the pair (c, c').

Yet another variation of the above property has been considered by Bellare *et al.* [BBS03] in their study of randomness re-use in multi-recipient PKE. The reproducibility test of [BBS03] can be cast in Definition 5 by requiring that $\mathbf{D}_{m,m'}$ and $\tilde{\mathbf{D}}_{m,m'}$ are identically distributed, and moreover Rand can produce the pair $(\tilde{c}, \tilde{c}')$ without knowing the randomness r' (corresponding to ciphertext c').

Our choice to go for the formulation above is due to the fact that Definition 5 is a weaker requirement, yet it is sufficient to prove security of our twist of the Naor-Yung paradigm.

3.3 Main Theorem: KDM Security

We now turn to state our main theorem, which quantifies the security of our twist of the Naor-Yung paradigm.

```
Game $\mathbf{G}_{0\text{-}9}$:
$(pk, sk) \leftarrow_\$ \mathsf{Gen}(1^\lambda)$
$(pk', sk') \leftarrow_\$ \mathsf{Gen}(1^\lambda)$
$sk^* = sk$
$b \leftarrow_\$ \{0,1\}$
$\mathcal{Q}_{\mathrm{dec}}, \mathcal{Q}_{\mathrm{kdm}}, \mathcal{Q}_H \leftarrow \emptyset$
$b' \leftarrow \mathsf{A}^{\mathcal{O}^{\mathrm{dec}}_{sk^*}(\cdot), \mathcal{O}^{\mathrm{kdm}}_{sk^*,b}(\cdot)}(pk^*)$
Return $(b' = b) \wedge (\mathcal{Q}_{\mathrm{dec}} \cap \mathcal{Q}_{\mathrm{kdm}} = \emptyset)$

Oracle $\mathcal{O}^H(x)$:
If $(x, \cdot) \notin \mathcal{Q}_H$
    $h \leftarrow_\$ \{0,1\}^*$                               $\backslash\mathbf{G}_{0,8,9}$
    $(h, \tau) \leftarrow_\$ \mathsf{S}(1, \tau, x)$          $\backslash\mathbf{G}_{1\text{-}7}$
    $\mathcal{Q}_H \leftarrow \mathcal{Q}_H \cup \{(x, h)\}$
    Return $h$
Else
    Return $h$

Oracle $\mathcal{O}^{\mathrm{dec}}_{sk^*}(c^*)$:
If $\mathsf{V}^H((pk, pk', c, c'), \pi) = 0$
    Return $\bot$
Else
    Return $\mathsf{Dec}(sk, c)$                               $\backslash\mathbf{G}_{0\text{-}3,9}$
    Return $\mathsf{Dec}(sk', c')$                             $\backslash\mathbf{G}_{4\text{-}8}$
$\mathcal{Q}_{\mathrm{dec}} \leftarrow \mathcal{Q}_{\mathrm{dec}} \cup \{c^*\}$

Oracle $\mathcal{O}^{\mathrm{kdm}}_{sk^*,b}(f)$:
$r, r' \leftarrow_\$ \mathcal{R}$
$c = \mathsf{Enc}(pk, f(sk); r)$                               $\backslash\mathbf{G}_{0\text{-}5}$
$c = \mathsf{Enc}(pk, 0^\mu; r)$                               $\backslash\mathbf{G}_{6\text{-}9}$
$c' = \mathsf{Enc}(pk', f(sk); r)$                             $\backslash\mathbf{G}_{0,1}$
$c' = \mathsf{Enc}(pk', f(sk); r')$                            $\backslash\mathbf{G}_{2}$
$c' = \mathsf{Enc}(pk', 0^\mu; r')$                            $\backslash\mathbf{G}_{3\text{-}6}$
$c' = \mathsf{Enc}(pk', 0^\mu; r)$                             $\backslash\mathbf{G}_{7\text{-}9}$
$\mathsf{aux} = (pk, pk', sk, r, f(sk))$                       $\backslash\mathbf{G}_{2\text{-}4}$
$\mathsf{aux} = (pk, pk', sk, r, 0^\mu)$                       $\backslash\mathbf{G}_{5\text{-}6}$
$(c, c') \leftarrow_\$ \mathsf{Rand}((c, c'), \mathsf{aux})$   $\backslash\mathbf{G}_{2\text{-}6}$
$\pi \leftarrow_\$ \mathsf{P}^H((pk, pk', c, c'), r)$          $\backslash\mathbf{G}_{0,8,9}$
$(\pi, \tau) \leftarrow_\$ \mathsf{S}(2, \tau, (pk, pk', c, c'))$   $\backslash\mathbf{G}_{1\text{-}7}$
$c^* = (c, c', \pi)$
Return $c^*$
$\mathcal{Q}_{\mathrm{kdm}} \leftarrow \mathcal{Q}_{\mathrm{kdm}} \cup \{c^*\}$
```

Figure 4: Games in the proof of Theorem 1.

Theorem 1 (Main theorem, KDM security)**.** *Let Π be a PKE scheme satisfying $\mathcal{F}$-KDM-CPA security and with the randomness fusion property (cf. Definition 5), and let $(\mathsf{P}^H, \mathsf{V}^H)$ be a simulation-sound NIZK for the language L^{Π}_{NY} of Eq. (1). Then, the PKE scheme Π^* described in Fig. 3 satisfies $\mathcal{F}$-KDM-CCA security in the random oracle model.*

Proof. We consider a series of games, depicted in Fig. 4, where the initial game is identical to the KDM-CCA experiment with hidden bit $b = 1$ and the last game is identical to the KDM-CCA experiment with $b = 0$. Hence, we show that the games are computationally indistinguishable unless one of the assumptions in the theorem statement is violated. This implies the theorem.

Game $\mathbf{G}_0$: This game is identical to the KDM-CCA security experiment for Π^*, with hidden bit $b = 1$. In particular, adversary A has access to the oracle $\mathcal{O}^{\mathrm{kdm}}_{sk^*,1}(\cdot)$, that upon input a query $f \in \mathcal{F}$ returns $\mathsf{Enc}^*(pk^*, f(sk))$. The oracle $\mathcal{O}^{\mathrm{dec}}_{sk^*}(\cdot)$, on input a ciphertext $c^* := (c, c', \pi)$, first checks that $\mathsf{V}^H((pk, pk', c, c'), \pi) = 1$; hence, it returns $\mathsf{Dec}(sk, c)$.

Game $\mathbf{G}_1$: We change the way ciphertexts returned by the KDM oracle are computed. In particular, the proof π attached to each ciphertext is now obtained by running the zero-knowledge simulator, as in $(\pi, \tau) \leftarrow_\$ \mathsf{S}(2, \tau, (pk, pk', c, c'))$. Furthermore, algorithm $\mathsf{S}(1, \tau, \cdot)$ is used to answer random oracle queries.

Game $\mathbf{G}_2$: We change the way ciphertexts returned by the KDM oracle are computed. In particular, instead of computing c and c' using the same randomness $r^* \leftarrow_\$ \mathcal{R}$, we sample $r, r' \leftarrow_\$ \mathcal{R}$, let $c = \mathsf{Enc}(pk, f(sk); r)$ and $c' = \mathsf{Enc}(pk', f(sk); r')$, and re-randomize (c, c')

by running $\mathsf{Rand}((c, c'), \mathsf{aux})$ with $\mathsf{aux} = (pk, pk', sk, r, f(sk))$. Note that, since the two ciphertexts use independent randomness, the attached simulated proof π is a proof of a false statement.

Game $\mathbf{G}_3$: We change the way ciphertexts returned by the KDM oracle are computed. In particular, we now let $c' = \mathsf{Enc}(pk', 0^\mu; r')$. Note that algorithm Rand always takes a single secret key as input, and this will be important while arguing indistinguishability with the previous game.

Game $\mathbf{G}_4$: We change the way decryption queries are answered. In particular, upon input a decryption query $c^* = (c, c', \pi)$, we verify the proof π (as before), and return $\mathsf{Dec}(sk', c')$ whenever π is accepting.

Game $\mathbf{G}_5$: We change the way ciphertexts returned by the KDM oracle are computed. In particular, the auxiliary information used by algorithm Rand is changed to $\mathsf{aux} = (pk, pk', sk', r', 0^\mu)$.

Game $\mathbf{G}_6$: We change the way ciphertexts returned by the KDM oracle are computed. In particular, the ciphertext c is obtained as an encryption of 0^μ, i.e. $c = \mathsf{Enc}(pk, 0^\mu; r)$ for $r \leftarrow\$ \mathcal{R}$.

Game $\mathbf{G}_7$: We change the way ciphertexts returned by the KDM oracle are computed. In particular, we do not run algorithm Rand anymore. This means that the ciphertexts c and c' are now obtained again using the same random coins $r^* \leftarrow\$ \mathcal{R}$, and thus the proof π is for a true statement.

Game $\mathbf{G}_8$: We change the way ciphertexts returned by the KDM oracle are computed. In particular, the proof π is now obtained using the real prover algorithm, as in $\pi \leftarrow\$ \mathsf{P}^H((pk, pk', c, c'), (0^\mu, r^*))$. Furthermore, random oracle queries are answered using the random oracle H.

Game $\mathbf{G}_9$: We change the way decryption queries are answered. In particular, we now return $\mathsf{Dec}(sk, c)$ (as long as the proof π is accepting). This yields a distribution identical to the one in the KDM-CCA security experiment for Π^*, with hidden bit $b = 0$.

Next, we proceed to show indistinguishability of the above defined games.

Claim 1. *For all PPT adversaries A there exists a negligible function $\nu_{0,1} : \mathbb{N} \to [0, 1]$ such that $|\mathbb{P}[\mathbf{G}_0 = 1] - \mathbb{P}[\mathbf{G}_1 = 1]| \leq \nu_{0,1}(\lambda)$.*

Proof of claim. The only difference between $\mathbf{G}_0$ and $\mathbf{G}_1$ is that the ciphertexts returned by the KDM oracle contain proofs π that are computed by running the real prover (with witness $(f(sk), r^*)$) in the former game, while the zero-knowledge simulator is used in the latter game. Moreover, in game $\mathbf{G}_1$ the queries to the random oracle are simulated by running $\mathsf{S}(1, \tau, \cdot)$. Thus, the claim follows readily from the non-interactive zero-knowledge property of $(\mathsf{P}^H, \mathsf{V}^H)$ (cf. Definition 3). □

Claim 2. *For all PPT adversaries A there exists a negligible function $\nu_{1,2} : \mathbb{N} \to [0, 1]$ such that $|\mathbb{P}[\mathbf{G}_1 = 1] - \mathbb{P}[\mathbf{G}_2 = 1]| \leq \nu_{1,2}(\lambda)$.*

Proof of claim. Game $\mathbf{G}_2$ is identical to $\mathbf{G}_1$, except that in the former algorithm Rand is run to re-randomize the pair of ciphertexts (c, c'), for each query to the KDM oracle. Put differently, the pair (c, c') corresponding to each ciphertext c^* computed inside the KDM oracle is sampled from the distribution $\mathbf{D}_{f(sk),f(sk)}$ in game $\mathbf{G}_1$, while the distribution $\tilde{\mathbf{D}}_{f(sk),f(sk)}$ is used in $\mathbf{G}_2$, where the distributions $\mathbf{D}$ and $\tilde{\mathbf{D}}$ are defined in Eq. (2) and Eq. (3).

We use a hybrid argument, where we switch the answer to KDM queries one by one. By the randomness fusion property of the PKE scheme (cf. Definition 5), we know that $\mathbf{D}_{f(sk),f(sk)}$ and

$\tilde{\mathbf{D}}_{f(sk),f(sk)}$ are statistically close, and so must be each pair of adjecent hybrids, and therefore also $\mathbf{G}_1$ and $\mathbf{G}_2$. $\square$

Claim 3. *For all PPT adversaries* A *there exists a negligible function* $\nu_{2,3} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_2 = 1] - \mathbb{P}[\mathbf{G}_3 = 1]| \le \nu_{2,3}(\lambda)$.

Proof of claim. Assume there exists a PPT adversary A and a polynomial $p_{2,3}(\cdot)$ such that, for infinitely many values of $\lambda \in \mathbb{N}$, we have $|\mathbb{P}[\mathbf{G}_2 = 1] - \mathbb{P}[\mathbf{G}_3 = 1]| \ge 1/p_{2,3}(\lambda)$. We construct a PPT adversary A' breaking CPA security (and thus $\mathcal{F}_{\mathrm{msg}}$-KDM-CPA security,[3] see Section 2.2) of Π, as follows.

- Receive pk' from the challenger, sample $(pk, sk) \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$, and return $pk^* = (pk, pk')$ to A.
- Upon input a query f to the KDM oracle, let $\bar{m} := f(sk)$ and forward $f_{\bar{m}} \in \mathcal{F}_{\mathrm{msg}}$ to the target KDM oracle receiving back a ciphertext c' (computed using fresh coins $r' \leftarrow\!\!\$\ \mathcal{R}$); compute $c = \mathsf{Enc}(pk, f(sk); r)$ using fresh coins $r \leftarrow\!\!\$\ \mathcal{R}$, run $(\tilde{c}, \tilde{c}') \leftarrow\!\!\$\ \mathsf{Rand}((c, c'), (pk, pk', sk, r, f(sk)))$, simulate the proof $\pi \leftarrow\!\!\$\ \mathsf{S}(2, \tau, (pk, pk', \tilde{c}, \tilde{c}'))$, and return $\tilde{c}^* = (\tilde{c}, \tilde{c}', \pi)$ to A.
- Upon input a query $c^* = (c, c', \pi)$ to the decryption oracle, answer this query as in both $\mathbf{G}_2$ and $\mathbf{G}_3$ (i.e., by decrypting c using sk, after verifying the proof π).
- Return the same guess as that of A.

Depending on A''s target oracle being initialized either with hidden bit $b = 1$ or $b = 0$, we obtain exactly the same distribution as in game $\mathbf{G}_2$ or $\mathbf{G}_3$. As a consequence, the above simulation is perfect. and A' retains the same advantage as A. The claim follows. $\square$

Claim 4. *For all PPT adversaries* A *there exists a negligible function* $\nu_{3,4} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_3 = 1] - \mathbb{P}[\mathbf{G}_4 = 1]| \le \nu_{3,4}(\lambda)$.

Proof of claim. Notice that $\mathbf{G}_3$ and $\mathbf{G}_4$ use a different decryption procedure; in particular, decryption queries are answered using the secret key sk in the former game, while sk' is used in the latter game. The proof is down to the simulation soundness property of the NIZK. Consider the following event, defined over the probability space of game $\mathbf{G}_4$: The event E becomes true whenever there exists a decryption query $c_i^* = (c_i, c_i', \pi_i')$ such that π_i' is accepting but $\mathsf{Dec}(sk, c_i) \ne \mathsf{Dec}(sk', c_i')$. Notice that the distributions of $\mathbf{G}_3$ and $\mathbf{G}_4$ are identical conditioned on event E not happening, hence, by a standard argument, it suffices to bound the probability that event E happens.

Assume there exists a PPT adversary A and a polynomial $p_{3,4}(\cdot)$, such that for infinitely many values of $\lambda \in \mathbb{N}$ adversary A provokes event E (in game $\mathbf{G}_4$) with probability at least $1/p_{3,4}(\lambda)$. We construct a PPT adversary A' attacking simulation soundness of $(\mathsf{P}^H, \mathsf{V}^H)$, as follows.

- Run $(pk, sk) \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$, $(pk', sk') \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$, and return $pk^* = (pk, pk')$ to A.
- Upon input a query $f \in \mathcal{F}$ to the KDM oracle, compute the pair of ciphertexts $(\tilde{c}, \tilde{c}')$ as in game $\mathbf{G}_4$, i.e. let $c = \mathsf{Enc}(pk, f(sk); r)$, $c' = \mathsf{Enc}(pk', 0^\mu; r')$, and $(\tilde{c}, \tilde{c}') \leftarrow\!\!\$\ \mathsf{Rand}((c, c'), \mathsf{aux})$. Hence, forward the statement $(pk, pk', \tilde{c}, \tilde{c}')$ to the target simulation oracle obtaining a proof π, and return $c^* = (\tilde{c}, \tilde{c}', \pi)$ to A.
- Answer A's queries to the decryption oracle as in game $\mathbf{G}_4$; this can be done because the reduction knows the secret key sk'.
- Let $(c_1, c_1', \pi_1), \ldots, (c_q, c_q', \pi_q)$ be the list of A's decryption queries. Find an index $i \in [q]$ such that $\mathsf{V}^{\mathsf{S}(1,\tau,\cdot)}((pk, pk', c_i, c_i'), \pi_i) = 1$ and $\mathsf{Dec}(sk, c_i) \ne \mathsf{Dec}(sk', c_i')$; return $((pk, pk', c_i, c_i'), \pi_i)$.

[3] Strictly speaking, we need to assume that $\mathcal{F}_{\mathrm{msg}} \subseteq \mathcal{F}$, but this is the case for all interesting classes $\mathcal{F}$.

Note that the simulation done by A' is perfect. Thus, A will provoke event E with probability $1/p_{3,4}(\lambda)$, and A' breaks simulation soundness of the NIZK with the same probability. This concludes the proof of the claim. □

Claim 5. *For all PPT adversaries* A *there exists a negligible function* $\nu_{4,5} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_4 = 1] - \mathbb{P}[\mathbf{G}_5 = 1]| \le \nu_{4,5}(\lambda)$.

Proof of claim. Game $\mathbf{G}_4$ and $\mathbf{G}_5$ differ on the auxiliary information provided to the algorithm Rand. Namely, the ciphertexts (c, c') corresponding to each query to the KDM oracle are sampled from the distribution $\tilde{\mathbf{D}}_{f(sk),f(sk)}$ in the former game, while the distribution $\tilde{\mathbf{D}}_{f(sk),0^\mu}$ is used in the latter game.

Define the hybrid game $\mathbf{G}_{4.5}$, where instead of running algorithm Rand the ciphertexts (c, c') corresponding to each query to the KDM oracle are sampled from the distribution $\mathbf{D}_{f(sk),0^\mu}$ of Eq. (2). A straightforward reduction to the CPA security (or the $\mathcal{F}_{\mathrm{msg}}$-KDM-CPA security) of Π shows that $\tilde{\mathbf{D}}_{f(sk),f(sk)} \approx_c \mathbf{D}_{f(sk),0^\mu}$. By a standard hybrid argument, where we switch the answer to KDM queries one by one, the above implies that $\mathbf{G}_4 \approx_c \mathbf{G}_{4.5}$.

Now, using the randomness fusion property of the PKE scheme (cf. Definition 5), we know that $\mathbf{D}_{f(sk),0^\mu} \approx_s \tilde{\mathbf{D}}_{f(sk),0^\mu}$, and thus, using again a hybrid argument on the number of KDM queries, we obtain that $\mathbf{G}_{4.5} \approx_s \mathbf{G}_5$. This concludes the proof. □

Claim 6. *For all PPT adversaries* A *there exists a negligible function* $\nu_{5,6} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_5 = 1] - \mathbb{P}[\mathbf{G}_6 = 1]| \le \nu_{5,6}(\lambda)$.

Proof of claim. The indistinguishability of the two games is due to the $\mathcal{F}$-KDM-CPA security of Π. In particular, assume there exists a PPT adversary A and a polynomial $p_{5,6}(\cdot)$ such that, for infinitely many values of $\lambda \in \mathbb{N}$, we have $|\mathbb{P}[\mathbf{G}_5 = 1] - \mathbb{P}[\mathbf{G}_6 = 1]| \ge 1/p_{5,6}(\lambda)$. Consider the following PPT adversary A' attacking $\mathcal{F}$-KDM-CPA security of Π.

- Receive pk from the challenger, sample $(pk', sk') \leftarrow\$ \mathsf{Gen}(1^\lambda)$, and return $pk^* = (pk, pk')$ to A.
- Upon input a query f to the KDM oracle, forward f to the target KDM oracle receiving back a ciphertext c, that is computed using fresh coins $r \leftarrow\$ \mathcal{R}$; compute $c' = \mathsf{Enc}(pk, 0^\mu; r')$ using fresh coins $r' \leftarrow\$ \mathcal{R}$, run $(\tilde{c}, \tilde{c}') \leftarrow\$ \mathsf{Rand}((c, c'), (pk, pk', sk', r', 0^\mu))$, simulate the proof $\pi \leftarrow\$ \mathsf{S}(2, \tau, (pk, pk', \tilde{c}, \tilde{c}'))$, and return $\tilde{c}^* = (\tilde{c}, \tilde{c}', \pi)$ to A.
- Upon input a query $c^* = (c, c', \pi)$ to the decryption oracle, answer this query as in both $\mathbf{G}_5$ and $\mathbf{G}_6$ (i.e., decrypt c' using sk' after verifying the proof π).
- Return the same guess as that of A.

Depending on A''s target KDM oracle being initialized with hidden bit $b = 1$ or $b = 0$, we obtain exactly the same distribution as in game $\mathbf{G}_5$ or $\mathbf{G}_6$. Thus, the above simulation is perfect and A' retains the same advantage as A. The claim follows. □

Claim 7. *For all PPT adversaries* A *there exists a negligible function* $\nu_{6,7} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_6 = 1] - \mathbb{P}[\mathbf{G}_7 = 1]| \le \nu_{6,7}(\lambda)$.

Proof of claim. The only difference between game $\mathbf{G}_6$ and $\mathbf{G}_7$ is that the latter game does not run algorithm Rand in order to re-randomize the pair of ciphertexts (c, c') within each answer to KDM queries. Put differently, the pair (c, c') corresponding to each ciphertext c^* computed inside the KDM oracle is sampled from the distribution $\mathbf{D}_{0^\mu,0^\mu}$ in game $\mathbf{G}_7$, while the distribution $\tilde{\mathbf{D}}_{0^\mu,0^\mu}$ is used in $\mathbf{G}_6$, where the distributions $\mathbf{D}$ and $\tilde{\mathbf{D}}$ are defined in Eq. (2) and Eq. (3). We use a hybrid argument, where we switch the answer to KDM queries one by one. By the randomness fusion property of the PKE scheme (cf. Definition 5), we known that $\mathbf{D}_{0^\mu,0^\mu}$

and $\tilde{\mathbf{D}}_{0^\mu,0^\mu}$ are statistically close, and so must be each pair of adjecent hybrids, and therefore also $\mathbf{G}_6$ and $\mathbf{G}_7$. □

Claim 8. *For all PPT adversaries* A *there exists a negligible function* $\nu_{7,8} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_7 = 1] - \mathbb{P}[\mathbf{G}_8 = 1]| \leq \nu_{7,8}(\lambda)$.

Proof of claim. The only difference between game $\mathbf{G}_7$ and $\mathbf{G}_8$ is that the ciphertexts returned by the KDM oracle contain proofs π that are computed by running the real prover (with witness $(0^\mu, r^*)$) in the latter game, while the zero-knowledge simulator is used in the former game. Moreover, in game $\mathbf{G}_8$ the queries to the random oracle are evaluated by running $H(\cdot)$. Thus, the claim follows readily from the non-interactive zero-knowledge property of $(\mathsf{P}^H, \mathsf{V}^H)$ (cf. Definition 3). □

Claim 9. *For all PPT adversaries* A *there exists a negligible function* $\nu_{8,9} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_8 = 1] - \mathbb{P}[\mathbf{G}_9 = 1]| \leq \nu_{8,9}(\lambda)$.

Proof of claim. The proof is down to the simulation soundness property of the NIZK.[4] Define the following event E in the probability space of game $\mathbf{G}_9$: The event occurs whenever there exists a decryption query $c_i^* = (c_i, c_i', \pi_i')$ such that π_i' is accepting but $\mathsf{Dec}(sk, c_i) \neq \mathsf{Dec}(sk', c_i')$. The distributions of $\mathbf{G}_8$ and $\mathbf{G}_9$ are identical conditioned on event E not happening. Hence, by a standard argument, it suffices to bound the probability that event E occurs.

Assume there exists a PPT adversary A and a polynomial $p_{8,9}(\cdot)$, such that, for infinitely many values of $\lambda \in \mathbb{N}$, adversary A provokes event E (in game $\mathbf{G}_9$) with probability at least $1/p_{8,9}(\lambda)$. A PPT adversary A' is constructed, attacking simulation soundness of $(\mathsf{P}^H, \mathsf{V}^H)$, as follows.

- Run $(pk, sk) \leftarrow\$ \mathsf{Gen}(1^\lambda)$, $(pk', sk') \leftarrow\$ \mathsf{Gen}(1^\lambda)$, and return $pk^* = (pk, pk')$ to A.
- Answer A's queries to the KDM and decryption oracles as in game $\mathbf{G}_9$. This can be done because the secret key sk is known to the reduction. Furthermore, the reduction can run the real prover to obtain the proof π, which is always for a true statement.
- Let $(c_1, c_1', \pi_1), \ldots, (c_q, c_q', \pi_q)$ be the list of A's decryption queries. Find an index $i \in [q]$ such that $\mathsf{V}^H((pk, pk', c_i, c_i'), \pi_i) = 1$ and $\mathsf{Dec}(sk, c_i) \neq \mathsf{Dec}(sk', c_i')$; return $((pk, pk', c_i, c_i'), \pi_i)$.

Note that the simulation done by A' is perfect. Thus, A will provoke event E with probability $1/p_{8,9}(\lambda)$, and A' breaks simulation soundness with the same probability. This concludes the proof of the claim. □

□

Remark 1. *While we did not write this explicitly, we stress that the KDM functions* $f \in \mathcal{F}$ *can depend on the random oracle* H*. As originally argued in [FKMV12], this dependency does not affect the above proof.*

3.4 Main Theorem: Key-Leakage Security

As proven in [NS09, FKMV12] the classical Naor-Yung paradigm allows to boost CPA security under Λ-key-leakage attacks to CCA security under Λ-key-leakage attacks. A similar result holds for our twist of the Naor-Yung construction, assuming the underlying PKE scheme meets the randomness fusion property.

[4] Actually soundness is already sufficient for this step of the proof, i.e. the reduction below does not need to make any oracle query to the zero-knowledge simulator.

```
Game $\mathbf{G}_{0\text{-}9}$:
$(pk, sk) \leftarrow_\$ \mathsf{Gen}(1^\lambda)$
$(pk', sk') \leftarrow_\$ \mathsf{Gen}(1^\lambda)$
$sk^* = sk$
$b \leftarrow_\$ \{0,1\}$
$\mathcal{Q}_{\mathrm{dec}}\ \mathcal{Q}_H \leftarrow \emptyset$
$(m_0, m_1) \leftarrow \mathsf{A}^{\mathcal{O}^{\mathrm{dec}}_{sk^*}(\cdot), \mathcal{O}^{\Lambda}_{sk^*}(\cdot)}(pk^*)$
$r, r' \leftarrow_\$ \mathcal{R}$
$c = \mathsf{Enc}(pk, m_1; r)$                          $\backslash\mathbf{G}_{0\text{-}5}$
$c = \mathsf{Enc}(pk, m_0; r)$                          $\backslash\mathbf{G}_{6\text{-}9}$
$c' = \mathsf{Enc}(pk', m_1; r)$                        $\backslash\mathbf{G}_{0,1}$
$c' = \mathsf{Enc}(pk', m_1; r')$                       $\backslash\mathbf{G}_{2,3}$
$c' = \mathsf{Enc}(pk', m_0; r')$                       $\backslash\mathbf{G}_{3\text{-}6}$
$c' = \mathsf{Enc}(pk', m_0; r)$                        $\backslash\mathbf{G}_{7\text{-}9}$
$\mathsf{aux} = (pk, pk', sk, r, m_1)$                  $\backslash\mathbf{G}_{2\text{-}4}$
$\mathsf{aux} = (pk, pk', sk, r, m_0)$                  $\backslash\mathbf{G}_{5\text{-}6}$
$(c, c') \leftarrow_\$ \mathsf{Rand}((c, c'), \mathsf{aux})$     $\backslash\mathbf{G}_{2\text{-}6}$
$\pi \leftarrow_\$ \mathsf{P}^H((pk, pk', c, c'), r)$          $\backslash\mathbf{G}_{0,8,9}$
$(\pi, \tau) \leftarrow_\$ \mathsf{S}(2, \tau, (pk, pk', c, c'))$   $\backslash\mathbf{G}_{1\text{-}7}$
$c_b^* = (c, c', \pi)$
$b' \leftarrow \mathsf{A}^{\mathcal{O}^{\mathrm{dec}}_{sk^*}(\cdot)}(c_b^*)$
Return $(b' = b) \wedge (c_b^* \notin \mathcal{Q}_{\mathrm{dec}})$

Oracle $\mathcal{O}^{\mathrm{dec}}_{sk^*}(c^*)$:
If $\mathsf{V}^H((pk, pk', c, c'), \pi) = 0$
    Return $\bot$
Else
    Return $\mathsf{Dec}(sk, c)$                        $\backslash\mathbf{G}_{0\text{-}3,9}$
    Return $\mathsf{Dec}(sk', c')$                      $\backslash\mathbf{G}_{4\text{-}8}$
$\mathcal{Q}_{\mathrm{dec}} \leftarrow \mathcal{Q}_{\mathrm{dec}} \cup \{c^*\}$

Oracle $\mathcal{O}^{\Lambda}_{sk^*}(f)$:
Return $f(sk^*)$

Oracle $\mathcal{O}^H(x)$:
If $(x, \cdot) \notin \mathcal{Q}_H$
    $h \leftarrow_\$ \mathcal{R}$                       $\backslash\mathbf{G}_{0,8,9}$
    $(h, \tau) \leftarrow_\$ \mathsf{S}(1, \tau, x)$        $\backslash\mathbf{G}_{1\text{-}7}$
    $\mathcal{Q}_H \leftarrow \mathcal{Q}_H \cup \{(x, h)\}$
    Return $h$
Else
    Return $h$
```

Figure 5: Games in the proof of Theorem 2.

Theorem 2 (Main theorem, key-leakage security)**.** *Let Π be a PKE scheme satisfying Λ-LKG-CPA security and with the randomness fusion property (cf. Definition 5), and let $(\mathsf{P}^H, \mathsf{V}^H)$ be a simulation-sound NIZK for the language L^{Π}_{NY} of Eq. (1). Then, the PKE scheme Π^* described in Fig. 3 satisfies Λ-LKG-CCA security in the random oracle model.*

Proof. We consider a series of games, depicted in Fig. 5, where the initial game is identical to the LKG-CCA experiment with hidden bit $b = 1$ and the last game is identical to the LKG-CCA experiment with $b = 0$. As we show, their outcomes cannot be mutually distinguished, as long as the theorem's hypothesis are not violated.

Game $\mathbf{G}_0$: This game is identical to the LKG-CCA security experiment for Π^*, with hidden bit $b = 1$. In particular, the challenge ciphertext $c_1 := (c, c', \pi)$ contains encryptions c and c' of the plaintext m_1. The oracle $\mathcal{O}^{\mathrm{dec}}_{sk^*}(\cdot)$, upon input a ciphertext $c^* := (c, c', \pi)$, first checks that $\mathsf{V}^H((pk, pk', c, c'), \pi) = 1$; hence, it returns the same as $\mathsf{Dec}(sk, c)$.

Game $\mathbf{G}_1$: We change the way the challenge ciphertext is computed. In particular, the proof π is now obtained by running the zero-knowledge simulator, as in $(\pi, \tau) \leftarrow_\$ \mathsf{S}(2, \tau, (pk, pk', c, c'))$. Furthermore, random oracle queries are answered thrugh the algorithm $\mathsf{S}(1, \tau, \cdot)$.

Game $\mathbf{G}_2$: We change the way the challenge ciphertext is computed. In particular, instead of computing c and c' using the same randomness $r^* \leftarrow_\$ \mathcal{R}$, we now sample $r, r' \leftarrow_\$ \mathcal{R}$, let $c = \mathsf{Enc}(pk, m_1; r)$, $c' = \mathsf{Enc}(pk', m_1; r')$, and finally re-randomize (c, c') by running $\mathsf{Rand}((c, c'), \mathsf{aux})$, with $\mathsf{aux} = (pk, pk', sk, r, m_1)$. Note that, since the two ciphertexts use independent randomness, the attached simulated proof π is a proof of a false statement.

Game $\mathbf{G}_3$: We change the way the challenge ciphertext is computed. In particular, the ciphertext c' is now computed as $c' = \mathsf{Enc}(pk', m_0; r')$. Note that algorithm Rand always takes a single secret key as input, and this will be important while arguing indistinguishability with the previous game.

Game $\mathbf{G}_4$: We change the way decryption queries are answered. In particular, upon input a decryption query $c^* := (c, c', \pi)$, we verify the proof π (as before) but return $\mathsf{Dec}(sk', c')$ in case the proof is accepting.

Game $\mathbf{G}_5$: We change the way the challenge ciphertext is computed. In particular, the auxiliary information used by algorithm Rand is changed to $\mathsf{aux} = (pk, pk', sk', r', m_0)$.

Game $\mathbf{G}_6$: We change the way the challenge ciphertext is computed. In particular, the ciphertex c is obtained as an encryption of m_0, i.e., $c = \mathsf{Enc}(pk, m_0; r)$ for $r \leftarrow\!\!\$\ \mathcal{R}$.

Game $\mathbf{G}_7$: We change the way the challenge ciphertext is computed. In particular, we do not run algorithm Rand anymore. This means that ciphertexts c and c' are now computed again using the same random coins $r^* \leftarrow\!\!\$\ \mathcal{R}$, and thus the proof π is for a true statement.

Game $\mathbf{G}_8$: We change the way the challenge ciphertext is computed. In particular, the proof π is now obtained by running the real prover algorithm, as in $\pi \leftarrow\!\!\$\ \mathsf{P}^H((pk, pk', c', c'), (m_0, r^*))$. Furthermore, random oracle queries are answered using the random oracle H.

Game $\mathbf{G}_9$: We change the way decryption queries are answered. In particular, we now return $\mathsf{Dec}(sk, c)$ (as long as the proof π is accepting). This yields a distribution identical to the one of the LKG-CCA security experiment for Π^*, with hidden bit $b = 0$.

Next, we proceed to show indistinguishability of the above defined games.

Claim 10. *For all PPT adversaries* A *there exists a negligible function* $\nu_{0,1} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_0 = 1] - \mathbb{P}[\mathbf{G}_1 = 1]| \leq \nu_{0,1}(\lambda)$.

Proof of claim. The only difference between $\mathbf{G}_0$ and $\mathbf{G}_1$ is that the challenge ciphertext contains a proof π that is computed by running the real prover (with witness (m_1, r^*)) in the former game, while the zero-knowledge simulator is used in the latter game. Moreover, in game $\mathbf{G}_1$ the queries to the random oracle are simulated by running $\mathsf{S}(1, \tau, \cdot)$. Thus, the claim follows readily from the non-interactive zero-knowledge property of $(\mathsf{P}^H, \mathsf{V}^H)$ (cf. Definition 3). □

Claim 11. *For all PPT adversaries* A *there exists a negligible function* $\nu_{1,2} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_1 = 1] - \mathbb{P}[\mathbf{G}_2 = 1]| \leq \nu_{1,2}(\lambda)$.

Proof of claim. Game $\mathbf{G}_2$ is identical to $\mathbf{G}_1$, except that in the former algorithm Rand is run to re-randomize the pair of ciphertexts (c, c'). Put differently, the pair (c, c') is sampled from the distribution $\mathbf{D}_{m_1,m_1}$ in game $\mathbf{G}_1$, while the distribution $\tilde{\mathbf{D}}_{m_1,m_1}$ is used in $\mathbf{G}_2$, where the distributions $\mathbf{D}$ and $\tilde{\mathbf{D}}$ are defined in Eq. (2) and Eq. (3). By the randomness fusion property of the PKE scheme (cf. Definition 5), we known that $\mathbf{D}_{m_1,m_1}$ and $\tilde{\mathbf{D}}_{m_1,m_1}$ are statistically close, and so must be $\mathbf{G}_1$ and $\mathbf{G}_2$. □

Claim 12. *For all PPT adversaries* A *there exists a negligible function* $\nu_{2,3} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_2 = 1] - \mathbb{P}[\mathbf{G}_3 = 1]| \leq \nu_{2,3}(\lambda)$.

Proof of claim. Assume there exists a PPT adversary A and a polynomial $p_{2,3}(\cdot)$ such that, for infinitely many values of $\lambda \in \mathbb{N}$, we have $|\mathbb{P}[\mathbf{G}_2 = 1] - \mathbb{P}[\mathbf{G}_3 = 1]| \geq 1/p_{2,3}(\lambda)$. We construct a PPT adversary A' breaking CPA security (and thus 0-LKG-CPA security) of Π, as follows.

- Receive pk' from the challenger, sample $(pk, sk) \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$, and return $pk^* = (pk, pk')$ to A.

- Upon input a query f to the leakage oracle, answer with $f(sk)$; this can be done as A' knows the secret key sk.
- When A outputs m_0 and m_1, forward (m_0, m_1) to the challenger obtaining a ciphertext c' (which is either an encryption of m_1 or an encryption of m_0 under pk', using fresh coins $r' \leftarrow\!\!\$\ \mathcal{R}$). Hence, compute $c = \mathsf{Enc}(pk, m_1; r)$ using fresh random coins $r \leftarrow\!\!\$\ \mathcal{R}$, run $(\tilde{c}, \tilde{c}') \leftarrow\!\!\$\ \mathsf{Rand}((c, c'), (pk, pk', sk, r, m_1))$, simulate the proof $\pi \leftarrow\!\!\$\ \mathsf{S}(2, \tau, (pk, pk', \tilde{c}, \tilde{c}'))$, and return $\tilde{c}^* = (\tilde{c}, \tilde{c}', \pi)$ to A.
- Upon input a query $c^* = (c, c', \pi)$ to the decryption oracle, answer this query as in both $\mathbf{G}_2$ and $\mathbf{G}_3$ (i.e., decrypt c using sk after verifying the proof π).
- Return the same guess as that of A.

Depending on A''s challenger using hidden bit $b = 1$ or $b = 0$, we obtain exactly the same distribution as in game $\mathbf{G}_2$ or $\mathbf{G}_3$. As a consequence, the above simulation is perfect and A' retains the same advantage as A. The claim follows. □

Claim 13. *For all PPT adversaries* A *there exists a negligible function* $\nu_{3,4} : \mathbb{N} \to [0, 1]$ *such that* $|\mathbb{P}[\mathbf{G}_3 = 1] - \mathbb{P}[\mathbf{G}_4 = 1]| \leq \nu_{3,4}(\lambda)$.

Proof of claim. Notice that $\mathbf{G}_3$ and $\mathbf{G}_4$ use a different decryption procedure; in particular, decryption queries are answered using the secret key sk in the former game, while sk' is used in the latter game. The proof is down to the simulation soundness property of the NIZK. Consider the following event, defined over the probability space of game $\mathbf{G}_4$: The event E becomes true whenever there exists a decryption query $c_i^* = (c_i, c_i', \pi_i')$ such that π_i' is accepting but $\mathsf{Dec}(sk, c_i) \neq \mathsf{Dec}(sk', c_i')$. Notice that the distributions of $\mathbf{G}_3$ and $\mathbf{G}_4$ are identical conditioned on event E not happening, hence, by a standard argument, it suffices to bound the probability that event E happens.

Assume there exists a PPT adversary A and a polynomial $p_{3,4}(\cdot)$ such that, for infinitely many values of $\lambda \in \mathbb{N}$, adversary A provokes event E (in game $\mathbf{G}_4$) with probability at least $1/p_{3,4}(\lambda)$. We construct a PPT adversary A' attacking simulation soundness of $(\mathsf{P}^H, \mathsf{V}^H)$, as follows.

- Run $(pk, sk) \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$, $(pk', sk') \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$, and return $pk^* = (pk, pk')$ to A.
- Upon input a leakage query f, answer with $f(sk)$; this can be done as A' knows the secret key sk.
- When A ouputs (m_0, m_1) compute the pair of ciphertexts $(\tilde{c}, \tilde{c}')$ as in game $\mathbf{G}_4$, i.e. let $c = \mathsf{Enc}(pk, m_1; r)$, $c' = \mathsf{Enc}(pk', m_0; r')$, and $(\tilde{c}, \tilde{c}') \leftarrow\!\!\$\ \mathsf{Rand}((c, c'), \mathsf{aux})$. Hence, forward the statement $(pk, pk', \tilde{c}, \tilde{c}')$ to the target simulation oracle obtaining a proof π, and return $c^* = (\tilde{c}, \tilde{c}', \pi)$ to A.
- Answer A's queries to the decryption oracle as in game $\mathbf{G}_4$; this can be done because the secret key sk' is known to the reduction.
- Let $(c_1, c_1', \pi_1), \ldots, (c_q, c_q', \pi_q)$ be the list of A's decryption queries. Find an index $i \in [q]$ such that $\mathsf{V}^{\mathsf{S}(1,\tau,\cdot)}((pk, pk', c_i, c_i'), \pi_i) = 1$ and $\mathsf{Dec}(sk, c_i) \neq \mathsf{Dec}(sk', c_i')$; return $((pk, pk', c_i, c_i'), \pi_i)$.

Notice that the simulation done by A' is perfect. Thus, A will provoke event E with probability $1/p_{3,4}(\lambda)$, and A' breaks simulation soundness with the same probability. This concludes the proof of the claim. □

Claim 14. *For all PPT adversaries* A *there exists a negligible function* $\nu_{4,5} : \mathbb{N} \to [0, 1]$ *such that* $|\mathbb{P}[\mathbf{G}_4 = 1] - \mathbb{P}[\mathbf{G}_5 = 1]| \leq \nu_{4,5}(\lambda)$.

Proof of claim. Game $\mathbf{G}_4$ and $\mathbf{G}_5$ differ on the auxiliary information provided to the algorithm Rand. Define the hybrid game $\mathbf{G}_{4.5}$, where instead of running algorithm Rand the ciphertexts (c, c') corresponding to the challenge ciphertext are sampled from the distribution $\tilde{\mathbf{D}}_{m_1,m_0}$ of Eq. (3). By the randomness fusion property is obtained $\mathbf{G}_4 \approx_s \mathbf{G}_{4.5} \approx_s \mathbf{G}_5$, which implies the claim. □

Claim 15. *For all PPT adversaries* A *there exists a negligible function* $\nu_{5,6} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_5 = 1] - \mathbb{P}[\mathbf{G}_6 = 1]| \le \nu_{5,6}(\lambda)$.

Proof of claim. The indistinguishability of the two games is down to the Λ-LKG-CPA security of Π. In particular, assume there exists a PPT adversary A and a polynomial $p_{5,6}(\cdot)$ such that, for infinitely many values of $\lambda \in \mathbb{N}$, we have $|\mathbb{P}[\mathbf{G}_5 = 1] - \mathbb{P}[\mathbf{G}_6 = 1]| \ge 1/p_{5,6}(\lambda)$. Consider the following PPT adversary A' attacking Λ-KDM-CPA security of Π.

- Receive pk from the challenger, sample $(pk', sk') \leftarrow\!\$\ \mathsf{Gen}(1^\lambda)$, and return $pk^* = (pk, pk')$ to A.
- Upon input a query f to the leakage oracle, forward f to the target leakage oracle and return the corresponding output to A.
- Whenever A outputs (m_0, m_1), forward (m_0, m_1) to the challenger obtaining a ciphertext c (which is either an encryption of m_0 or an encryption of m_1 under public key pk, using fresh random coins $r \leftarrow\!\$\ \mathcal{R}$). Hence, compute $c' = \mathsf{Enc}(pk, m_0; r')$ using fresh coins $r' \leftarrow\!\$\ \mathcal{R}$, run $(\tilde{c}, \tilde{c}') \leftarrow\!\$\ \mathsf{Rand}((c, c'), (pk, pk', sk', r', m_0))$, simulate the proof $\pi \leftarrow\!\$\ \mathsf{S}(2, \tau, (pk, pk', \tilde{c}, \tilde{c}'))$, and return $\tilde{c}^* = (\tilde{c}, \tilde{c}', \pi)$ to A.
- Upon input a query $c^* = (c, c', \pi)$ to the decryption oracle, answer this query as in both $\mathbf{G}_5$ and $\mathbf{G}_6$ (i.e., decrypt c' using sk' after verifying the proof π).
- Return the same guess as that of A.

Depending on A''s challenger using hidden bit $b = 1$ or $b = 0$, we obtain exactly the same distribution as in game $\mathbf{G}_5$ or $\mathbf{G}_6$. Thus, the above simulation is perfect and A' retains the same advantage as A. The claim follows. □

Claim 16. *For all PPT adversaries* A *there exists a negligible function* $\nu_{6,7} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_6 = 1] - \mathbb{P}[\mathbf{G}_7 = 1]| \le \nu_{6,7}(\lambda)$.

Proof of claim. The only difference between $\mathbf{G}_6$ and $\mathbf{G}_7$ is that the latter game does not run algorithm Rand in order to re-randomize the pair of ciphertexts (c, c') within the challenge ciphertext. Put differently, the pair (c, c') is sampled from the distribution $\mathbf{D}_{m_0,m_0}$ in game $\mathbf{G}_7$ and from the distribution $\tilde{\mathbf{D}}_{m_0,m_0}$ in $\mathbf{G}_6$, where the distributions $\mathbf{D}$ and $\tilde{\mathbf{D}}$ are defined in Eq. (2) and Eq. (3). By the randomness fusion property of the PKE scheme (cf. Definition 5), we known that $\mathbf{D}_{m_0,m_0}$ and $\tilde{\mathbf{D}}_{m_0,m_0}$ are statistically close, and so must be $\mathbf{G}_6$ and $\mathbf{G}_7$. □

Claim 17. *For all PPT adversaries* A *there exists a negligible function* $\nu_{7,8} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_7 = 1] - \mathbb{P}[\mathbf{G}_8 = 1]| \le \nu_{7,8}(\lambda)$.

Proof of claim. The only difference between game $\mathbf{G}_7$ and $\mathbf{G}_8$ is that the proof π corresponding to the challenge ciphertext is computed by running the real prover (with witness (m_0, r^*)) in the former game, while the zero-knowledge simulator is used in $\mathbf{G}_7$. Moreover, in game $\mathbf{G}_8$ the queries to the random oracle are evaluated by running $H(\cdot)$. Thus, the claim follows readily from the non-interactive zero-knowledge property of $(\mathsf{P}^H, \mathsf{V}^H)$ (cf. Definition 3). □

Claim 18. *For all PPT adversaries* A *there exists a negligible function* $\nu_{8,9} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathbf{G}_8 = 1] - \mathbb{P}[\mathbf{G}_9 = 1]| \le \nu_{8,9}(\lambda)$.

Proof of claim. The proof is down to the simulation soundness property of the NIZK. Define the following event E in the probability space of game $\mathbf{G}_9$: The event occours whenever there exists a decryption query $c_i^* = (c_i, c_i', \pi_i')$ such that π_i' is accepting but $\mathsf{Dec}(sk, c_i) \neq \mathsf{Dec}(sk', c_i')$. The distributions of $\mathbf{G}_8$ and $\mathbf{G}_9$ are identical conditioned on event E not happening. Hence, by a standard argument, it suffices to bound the probability that event E happens.

Assume there exists a PPT adversary A and a polynomial $p_{8,9}(\cdot)$ such that, for infinitely many values of $\lambda \in \mathbb{N}$, adversary A provokes event E (in game $\mathbf{G}_9$) with probability at least $1/p_{8,9}(\lambda)$. We construct a PPT adversary A' attacking simulation soundness of $(\mathsf{P}^H, \mathsf{V}^H)$, as follows.

- Run $(pk, sk) \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$, $(pk', sk') \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$, and return $pk^* = (pk, pk')$ to A.
- Upon input a leakage query f from A, answer with $f(sk)$; this can be done as A' knows the secret key sk.
- Whenever A outputs (m_0, m_1), compute the challenge ciphertext $c^* = (c, c', \pi)$ as in $\mathbf{G}_9$; this can be done as A' can run the real prover to obtain the proof π, which is always for a true statement.
- Let $(c_1, c_1', \pi_1), \ldots, (c_q, c_q', \pi_q)$ be the list of A's decryption queries. Find an index $i \in [q]$ such that $\mathsf{V}^H((pk, pk', c_i, c_i'), \pi_i) = 1$ and $\mathsf{Dec}(sk, c_i) \neq \mathsf{Dec}(sk', c_i')$; return $((pk, pk', c_i, c_i'), \pi_i)$.

Note that the simulation done by A' is perfect. Thus, A will provoke event E with probability $1/p_{8,9}(\lambda)$ and A' breaks simulation soundness with the same probability. This concludes the proof of the claim. □

□

Remark 2. *While we did not write this explicitly, we stress that in the above proof the leakage functions can depend on the random oracle H. See also Remark 1.*

4 KDM Security from Subset Sum

We start by recalling the Subset Sum assumption in Section 4.1. Our new Subset-Sum based PKE scheme is described in Section 4.2, and its correctness and security are showed in Section 4.3 and Section 4.4, respectively.

4.1 The Subset Sum Problem

In its simplest form, the *search* version of the Subset Sum problem—denoted $\mathrm{SS}(n, q)$ and parameterized by values $n(\lambda), q(\lambda) \in \mathbb{N}$—asks to compute a secret vector $\mathbf{s}$ given $(\mathbf{a}, t)$ such that $t := \langle \mathbf{a}, \mathbf{s} \rangle \bmod q$, where both $\mathbf{a} \in \mathbb{Z}_q^n$ and $\mathbf{s} \in \{0,1\}^n$ are randomly chosen. The decisional version of the problem, instead, asks to distinguish $(\mathbf{a}, t)$ from $(\mathbf{a}, u)$ where u is uniform in $\mathbb{Z}_q$. The equivalence between the search and the decisional version of the Subset Sum problem has been established in a seminal paper by Impagliazzo and Naor [IN96].

Below, we recall a variant of the Subset Sum problem which was considered for the first time by Lyubashevsky, Palacio and Segev [LPS10]. Here the modulus q is a power of an odd number; in our case we will set $q := p^m$, for some $m \in \mathbb{N}$. Such a variant of the problem helps interpreting the Subset Sum problem as an instance of the Learning with Errors [Reg05, Reg09] (LWE) problem with "deterministic noise", as we recall below.

Definition 6 (Subset Sum assumption)**.** For security parameter $\lambda \in \mathbb{N}$, and parameters $n(\lambda)$, $p(\lambda)$, $m(\lambda) \in \mathbb{N}$, consider the following distribution $\mathbf{D}_{\mathrm{SS}}(\lambda, n, p, m)$:

- Sample $\mathbf{A} \leftarrow\!\!\$\, \mathbb{Z}_p^{m\times n}$ and $\mathbf{s} \leftarrow\!\!\$\, \{0,1\}^n$.
- Parse $\mathbf{A} := (a_{1,1}, \dots, a_{m,n})$, $\mathbf{s} := (s_1, \dots, s_n)$, compute $\mathbf{A}\cdot\mathbf{s} \in \mathbb{Z}_p^n$, and let $e_1(\mathbf{A},\mathbf{s}) := 0$. For all $j \in [m]$, $j \neq 1$, compute

$$e_j(\mathbf{A},\mathbf{s}) := \left\lfloor \frac{e_{j-1}(\mathbf{A},\mathbf{s}) + \sum_{i=1}^n s_i \cdot a_{j-1,i}}{p} \right\rfloor \bmod p.$$

- Set $\mathbf{e}(\mathbf{A},\mathbf{s}) := (e_m(\mathbf{A},\mathbf{s}), \dots, e_1(\mathbf{A},\mathbf{s}))^\intercal$ and $\mathbf{t} := \mathbf{A}\cdot\mathbf{s} + \mathbf{e}(\mathbf{A},\mathbf{s})$. Output $(\mathbf{A},\mathbf{t},\mathbf{s})$.

We say that the decisional Subset Sum assumption $\mathrm{SS}(n, p^m)$ holds, if for all PPT distinguishers D there exists a negligible function $\nu : \mathbb{N} \to [0,1]$ such that

$$\Big|\mathbb{P}\left[\mathsf{D}(\mathbf{A},\mathbf{t}) = 1 :\ (\mathbf{A},\mathbf{t},\mathbf{s}) \leftarrow\!\!\$\, \mathbf{D}_{\mathrm{SS}}(\lambda,n,p,m)\right] \\ - \mathbb{P}\left[\mathsf{D}(\mathbf{A},\mathbf{u}) = 1 :\ (\mathbf{A},\mathbf{u}) \leftarrow\!\!\$\, \mathbb{Z}_p^{m\times n} \times \mathbb{Z}_p^m\right]\Big| \leq \nu(\lambda).$$

It can be shown that the above decisional version of Subset Sum is equivalent to the search version (i.e., to finding $\mathbf{s}$). In fact, [LPS10] showed that the representation $(\mathbf{A},\mathbf{t}) \in \mathbb{Z}_p^{m\times n} \times \mathbb{Z}_p^m$ of Subset Sum is equivalent to the original representation $(\mathbf{a}, t) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, whenever $q = p^m$ and $p \geq 2\sqrt{n}\log n + 3$. In particular, given $\mathbf{a} := (a_1, \dots, a_n)$ and $\mathbf{s} := (s_1, \dots, s_n)$, the matrix $\mathbf{A} := (a_{1,1}, \dots, a_{m,n})$ can be defined as follows. For $i \in [n]$ and $j \in [m]$, let $a_{j,i} := \lfloor \frac{a_i}{p^{j-1}} \rfloor \bmod p$, and interpret the vector $\mathbf{e}(\mathbf{A},\mathbf{s})$ as the vector of carries in the computation of $t := \sum_{i=1}^n s_i \cdot a_i \bmod p^m$. This way, the value t directly corresponds to

$$\left(\sum_{i=1}^n s_i \begin{pmatrix} a_{m,i} \\ \vdots \\ a_{1,i} \end{pmatrix}\right) + \begin{pmatrix} e_m(\mathbf{A},\mathbf{s}) \\ \vdots \\ e_0(\mathbf{A},\mathbf{s}) \end{pmatrix} = \mathbf{t},$$

as desired.

Therefore, Subset Sum can be seen as LWE with deterministic noise $\mathbf{e}(\mathbf{A},\mathbf{s})$ which only depends on $\mathbf{A}$ and $\mathbf{s}$. An important difference between Subset Sum and LWE is that for LWE the value m can be arbitrarily large as long as it remains polynomial. Instead, for Subset Sum the density $\delta := n/\log q = n/(m\log p)$ decreases with the size of m; this implies that Subset Sum can be solved efficiently for $m \approx n^2$. However, the problem is considered to be hard whenever $\delta \in O(1/\log n)$.

The following lemma, which can be easily derived from [LPS10, Lemma 3.4], states that the deterministic noise $\mathbf{e}(\mathbf{A},\mathbf{s})$ is small, and additionally it remains small when multiplied by a matrix $\mathbf{R}$ with components of bounded size.

Lemma 1 ([LPS10])**.** *For security parameter $\lambda \in \mathbb{N}$, and parameters $n(\lambda)$, $p(\lambda)$, $m(\lambda)$, $\ell(\lambda) \in \mathbb{N}$, let $\ell, m \in \mathrm{poly}(\lambda)$ and p be a prime such that $p \geq 2\sqrt{n}\log n + 3$. Let $(\mathbf{A},\mathbf{t},\mathbf{s}) \leftarrow\!\!\$\, \mathbf{D}_{SS}(\lambda,n,p,m)$ and $\mathbf{R} \leftarrow\!\!\$\, [-\lfloor\sqrt{p}/2\rfloor, \lfloor\sqrt{p}/2\rfloor]^{\ell\times m}$. There exist negligible functions $\nu, \nu' : \mathbb{N} \to [0,1]$ such that*

$$\mathbb{P}\left[\|\mathbf{e}(\mathbf{A},\mathbf{s})\|_\infty < \sqrt{n}\log n + 1\right] \geq 1 - \nu(\lambda)$$
$$\mathbb{P}\left[\|\mathbf{R}\cdot\mathbf{e}(\mathbf{A},\mathbf{s})\|_\infty < \sqrt{pmn}\log^2 n + n\sqrt{p}\right] \geq 1 - \nu'(\lambda). \qquad (4)$$

Leftover hash lemma. Let $\mathcal{H} := \{h : \mathcal{D} \to \mathcal{I}\}$ be a family of hash functions with domain $\mathcal{D}$ and image $\mathcal{I}$. Recall that $\mathcal{H}$ is called *universal* if for any $x \in \mathcal{D}$ and $x' \in \mathcal{D}$ the following holds:

$$\mathop{\mathbb{P}}_{h \leftarrow\!\$\, \mathcal{H}}\left[h(x) = h(x')\right] = \frac{1}{|\mathcal{I}|}.$$

The celebrated leftover hash lemma [HILL99, AP11] states that, over a random choice of $h \leftarrow_\$ \mathcal{H}$, $x \leftarrow_\$ \mathcal{D}$, and $u \leftarrow_\$ \mathcal{I}$, the statistical distance between $(h, h(x))$ and (h, u) is smaller than $1/2\sqrt{|\mathcal{I}|/|\mathcal{D}|}$.

It is easy to show that matrices in $\mathbb{Z}_p^{m\times n}$ are a family of universal hash functions for prime p and any domain $\mathcal{D} \subseteq \mathbb{Z}_p^m$. As a consequence, we obtain the following lemma which will be important for showing security of our PKE scheme.

Lemma 2. *For prime p and values $n, m, \ell \in \mathbb{N}$, let $\mathbf{A} \leftarrow_\$ \mathbb{Z}_p^{m\times n}$, $\mathbf{u}_1, \mathbf{u}_2 \leftarrow_\$ \mathbb{Z}_p^m$, $\mathbf{R} \leftarrow_\$ [-\lfloor\sqrt{p}/2\rfloor, \lfloor\sqrt{p}/2\rfloor]^{\ell\times m}$, and $\mathbf{B} \leftarrow_\$ \mathbb{Z}_p^{\ell\times(n+2)}$. Then,*

$$\Delta\left((\mathbf{A}, \mathbf{u}_1, \mathbf{u}_2, \mathbf{RA}, \mathbf{Ru}_1, \mathbf{Ru}_2); (\mathbf{A}, \mathbf{u}_1, \mathbf{u}_2, \mathbf{B})\right) \leq \frac{\ell}{2}\sqrt[4]{2^{2(n+2)\log p - m\log(p-2)}}.$$

Proof. Since $\mathcal{H} := \mathbb{Z}_p^{m\times(n+2)}$ is a family of universal hash functions with domain $\mathcal{D} := [-\lfloor\sqrt{p}/2\rfloor, \lfloor\sqrt{p}/2\rfloor]^m$ and image $\mathcal{I} := \mathbb{Z}_p^{n+2}$, the statement follows directly by the leftover hash lemma and the triangle inequality (via a standard hybrid argument). □

4.2 Scheme Description

We now describe a PKE scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, with message space $\mathcal{M} = \{0,1\}^\ell$ for an arbitrary polynomial $\ell(\cdot)$. The scheme depends on the Subset Sum distribution of Definition 6, with parameters n, p, $m \in \mathbb{N}$.

Key Generation: Upon input the security parameter $\lambda \in \mathbb{N}$, the randomized key generation algorithm Gen samples $(\mathbf{A}, \mathbf{t}, \mathbf{s}) \leftarrow_\$ \mathbf{D}_{\mathrm{SS}}(\lambda, n, p, m)$ and defines $pk := (\mathbf{A}, \mathbf{t})$ and $sk := \mathbf{s}$.

Encryption: Upon input a plaintext $M \in \{0,1\}^\ell$ and the public key $pk := (\mathbf{A}, \mathbf{t})$, the randomized encryption algorithm Enc picks a random matrix $\mathbf{R} \leftarrow_\$ [-\lfloor\sqrt{p}/2\rfloor, \lfloor\sqrt{p}/2\rfloor]^{\ell\times m}$ and returns $C := (\mathbf{A}', \mathbf{t}' + \mathbf{m}\cdot\lfloor\frac{p}{2}\rfloor)$ such that $\mathbf{A}' := \mathbf{R}\cdot\mathbf{A}$, $\mathbf{t}' := \mathbf{R}\cdot\mathbf{t}$, and $\mathbf{m} \in \mathbb{Z}_2^\ell$ is the vector representation of the plaintext $M \in \{0,1\}^\ell$.

Decryption: Upon input the secret key $sk := \mathbf{s}$ and a ciphertext $C := (\mathbf{C}_1, \mathbf{c}_2)$, the deterministic decryption algorithm Dec returns $\lfloor \mathbf{c}_2 - \mathbf{C}_1\cdot\mathbf{s}\rceil_2 \in \{0,1\}^\ell$.

4.3 Proof of Correctness

The theorem below states that the above defined PKE scheme meets the correctness requirement, i.e. decryption of honestly computed ciphertexts yields the corresponding plaintext.

Theorem 3 (Correctness of PKE scheme)**.** *Let $n, p, q \in \mathbb{N}$ be parameters such that p is a prime, $p \geq 25mn\log^4 n$, $n \geq 10$, $m \in \Theta(n)$, and $\ell \in O(n^k)$ for some constant $k \in \mathbb{N}$. Then, the PKE scheme of Section 4.2 satisfies correctness.*

Further, correctness holds for ciphertexts of the form $C := (\mathbf{A}', \mathbf{t}' + \mathbf{m}\circ\lfloor\boldsymbol{\xi}/2\rfloor)$, for any vector $\boldsymbol{\xi} \in [p-n-1, p]^\ell$, and where $\circ$ denotes the Hadamard product.

Proof. We prove directly the second part of the statement, as it implies the first part. The decryption algorithm computes

$$\begin{aligned}\lfloor \mathbf{c}_2 - \mathbf{C}_1\cdot\mathbf{s}\rceil_2 &= \lfloor \mathbf{t}' + \mathbf{m}\circ\lfloor\boldsymbol{\xi}/2\rfloor - \mathbf{A}'\cdot\mathbf{s}\rceil_2 = \lfloor \mathbf{R}\cdot\mathbf{t} + \mathbf{m}\circ\lfloor\boldsymbol{\xi}/2\rfloor - \mathbf{R}\cdot\mathbf{A}\cdot\mathbf{s}\rceil_2 \\ &= \lfloor \mathbf{R}(\mathbf{A}\cdot\mathbf{s} + \mathbf{e}(\mathbf{A},\mathbf{s})) + \mathbf{m}\circ\lfloor\boldsymbol{\xi}/2\rfloor - \mathbf{R}\cdot\mathbf{A}\cdot\mathbf{s}\rceil_2 \\ &= \lfloor \mathbf{R}\cdot\mathbf{e}(\mathbf{A},\mathbf{s}) + \mathbf{m}\circ\lfloor\boldsymbol{\xi}/2\rfloor\rceil_2,\end{aligned}$$

where the third equality comes from the definition of the Subset Sum distribution. Finally, for parameters m, n, p as in the theorem statement, by using the bound of Eq. (4) from Lemma 1, with overwhelming probability over the choice of $pk, sk, \mathbf{R}$, we obtain that $\|\mathbf{R} \cdot \mathbf{e}(\mathbf{A}, \mathbf{s})\|_\infty$ is smaller than $\sqrt{pmn} \log^2 n + n\sqrt{p}$. By choosing $p \geq 25mn \log^4 n$ and $n \geq 10$,

$$\|\mathbf{R} \cdot \mathbf{e}(\mathbf{A}, \mathbf{s})\|_\infty < \lfloor p/4 \rfloor - \lfloor (n+1)/2 + 1 \rfloor \leq \lfloor p/4 \rfloor - \| \lfloor \boldsymbol{\xi}'/2 + \mathbf{1} \rfloor \|_\infty$$

holds for $\boldsymbol{\xi}' \in [0, n+1]^\ell$, and thus

$$\begin{aligned} \lfloor \mathbf{R} \cdot \mathbf{e}(\mathbf{A}, \mathbf{s}) + \mathbf{m} \circ \lfloor \boldsymbol{\xi}/2 \rfloor \rceil_2 &= \lfloor (\mathbf{R} \cdot \mathbf{e}(\mathbf{A}, \mathbf{s}) - \mathbf{m} \circ \lfloor \boldsymbol{\xi}'/2 + \mathbf{1} \rfloor + \mathbf{m} \lfloor p/2 \rfloor) \lfloor p/2 \rfloor^{-1} \rceil \\ &= \lfloor (\mathbf{R} \cdot \mathbf{e}(\mathbf{A}, \mathbf{s}) - \mathbf{m} \circ \lfloor \boldsymbol{\xi}'/2 + \mathbf{1} \rfloor) \cdot \lfloor p/2 \rfloor^{-1} \rceil + \lfloor \mathbf{m} \rceil = \mathbf{m}. \end{aligned}$$

□

4.4 Proof of Security

We now prove that our PKE scheme satisfies a form of KDM security, as formalized in the theorem below. The set of manipulations tolerated by the scheme consists of the set of all affine functions of the form

$$\mathcal{F}_{\text{aff}} := \{f : \; f(\mathbf{s}) := \mathbf{F} \cdot \mathbf{s} + \mathbf{f}\}_{\mathbf{F} \in \mathbb{Z}_2^{\ell \times n}, \mathbf{f} \in \mathbb{Z}_2^\ell}.$$

We remark that a generic amplification theorem by Applebaum [App11, App14] allows to boost $\mathcal{F}_{\text{aff}}$-KDM-CPA security to $\mathcal{G}$-KDM-CPA security, where $\mathcal{G}$ consists of the family of functions that can computed in some fixed polynomial time (or the set of all polynomial-size circuits whose size grows with their input and output lengths via a fixed polynomial rate).

For technical reasons, we need that when encrypting a function of the secret key, the ciphertext has a slightly different form. Namely, $\mathbf{c}_2' := \mathbf{t}' + \left(\mathbf{F} \cdot \lfloor \frac{p}{2} \rfloor\right) \cdot \mathbf{s} + \lfloor \frac{p}{2} \rfloor \cdot \mathbf{f}$ instead of $\mathbf{c}_2 := \mathbf{t}' + (\mathbf{F} \cdot \mathbf{s} + \mathbf{f}) \lfloor \frac{p}{2} \rfloor$. This can be easily done by the encryption algorithm whenever $\mathbf{F}$, $\mathbf{f}$ and $\mathbf{s}$ are known. Furthermore, $\mathbf{c}_2'$ and $\mathbf{c}_2$ decrypt to the same value. This can be seen by noticing that $\lfloor \frac{p}{2} \rfloor = \frac{p-1}{2}$, the multiplication with $\mathbf{s}$ and addition with $\mathbf{f}$ is for each component the sum of at most $n+1$ values $\frac{p-1}{2}$ modulo p, and hence $\mathbf{c}_2'$ is a ciphertext of the form $\mathbf{c}_2' = \mathbf{t}' + \lfloor \frac{\boldsymbol{\xi}}{2} \rfloor \circ \mathbf{m}$ for some $\boldsymbol{\xi} \in [p-n-1, p]^\ell$ (cf. Theorem 3).

The reason for this obstacle is that we need to map the function f, which lives in $\mathbb{Z}_2$, into $\mathbb{Z}_p$. Since p is prime, it does not have a subgroup of size 2 to which we could map the components of $\mathbf{F}$ and $\mathbf{f}$. Therefore we need to map them to either $\frac{p-1}{2}$ (when 1) or to 0 (when 0). Since we do not map them to a subgroup, the output of f will also not be in a subgroup, but within range $[p-n-1, p]$ (when 1) or $[-n-1, 0]$ (when 0). One could resolve this obstacle by choosing p even, but then the leftover-hash lemma does only apply for a matrix $\mathbf{R}$ with components in $\{0, 1\}$, such that m needs to be larger. This would decrease the density of the underlying Subset Sum instance to $1/\log^2(n)$. Therefore, we prefer our approach.

Theorem 4 (KDM security of PKE scheme)**.** *Let $n, p, q \in \mathbb{N}$ be parameters such that p is a prime, $p \geq 25mn \log^4 n$, $m \in \Theta(n)$, and $\ell \in O(n^k)$ for some constant $k \in \mathbb{N}$. If the* SS(n, p^m) *assumption holds (achieved with density $\delta \in \Theta(1/\log n)$), then the PKE scheme Π from Section 4.2 satisfies $\mathcal{F}_{\text{aff}}$-KDM-CPA security.*

Proof. We consider a series of games, where the initial game is identical to the KDM-CPA experiment with hidden bit $b = 1$ and the last game is identical to the KDM-CPA experiment with $b = 0$. Hence, we show that the games are computationally indistinguishable unless one of the assumptions in the theorem statement is violated. This implies the theorem.

Game $\mathbf{G}_0$: This game is identical to the KDM-CPA experiment for the PKE scheme Π of Section 4.2, with hidden bit $b = 1$. In particular, this means that the adversary has access to oracle $\mathcal{O}^{\mathrm{kdm}}_{\mathbf{s},1}(\cdot)$ which, upon input a query $(\mathbf{F}, \mathbf{f}) \in \mathcal{F}_{\mathrm{aff}}$, returns a ciphertext $C = (\mathbf{C}_1, \mathbf{c}_2)$ such that

$$\mathbf{C}_1 := \mathbf{R} \cdot \mathbf{A} \qquad \mathbf{c}_2 := \mathbf{R} \cdot \mathbf{t} + \left(\mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor\right) \cdot \mathbf{s} + \left\lfloor \frac{p}{2} \right\rfloor \cdot \mathbf{f},$$

where $\mathbf{R} \leftarrow\!\!\$\, [-\lfloor\sqrt{p}/2\rfloor, \lfloor\sqrt{p}/2\rfloor]^{\ell \times m}$, and $(\mathbf{A}, \mathbf{t}, \mathbf{s}) \leftarrow\!\!\$\, \mathbf{D}_{\mathrm{SS}}(\lambda, n, p, m)$.

Game $\mathbf{G}_1$: We change the way queries to the KDM oracle are answered. Namely, upon input a query $(\mathbf{F}, \mathbf{f}) \in \mathcal{F}_{\mathrm{aff}}$, we now return a ciphertext $C := (\mathbf{C}_1, \mathbf{c}_2)$ such that

$$\mathbf{C}_1 := \mathbf{R} \cdot \mathbf{A} - \mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor \qquad \mathbf{c}_2 := \mathbf{R} \cdot \mathbf{t} + \mathbf{f} \cdot \left\lfloor \frac{p}{2} \right\rfloor.$$

Game $\mathbf{G}_2$: We change the distribution of the public key. Namely, instead of having $sk = \mathbf{s}$ and $pk = (\mathbf{A}, \mathbf{t})$ where $(\mathbf{A}, \mathbf{t}, \mathbf{s}) \leftarrow\!\!\$\, \mathbf{D}_{\mathrm{SS}}(\lambda, n, p, m)$, we now have $sk = \mathbf{s}$ and $pk = (\mathbf{A}, \mathbf{u})$ where $(\mathbf{A}, \mathbf{u}) \leftarrow\!\!\$\, \mathbb{Z}_p^{m \times n} \times \mathbb{Z}_p^m$ and (as before) $\mathbf{s} \leftarrow\!\!\$\, \{0,1\}^n$. Queries to the KDM oracle are answered as in the previous game. Namely, upon input a query $(\mathbf{F}, \mathbf{f}) \in \mathcal{F}_{\mathrm{aff}}$, we now return a ciphertext $C := (\mathbf{C}_1, \mathbf{c}_2)$ such that

$$\mathbf{C}_1 := \mathbf{R} \cdot \mathbf{A} - \mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor \qquad \mathbf{c}_2 := \mathbf{R} \cdot \mathbf{u} + \mathbf{f} \cdot \left\lfloor \frac{p}{2} \right\rfloor.$$

Game $\mathbf{G}_3$: This game is identical to the KDM-CPA experiment for the PKE scheme Π of Section 4.2, with hidden bit $b = 0$. In particular, this means that the adversary has access to oracle $\mathcal{O}^{\mathrm{kdm}}_{\mathbf{s},0}(\cdot)$ which, upon input a query $(\mathbf{F}, \mathbf{f}) \in \mathcal{F}_{\mathrm{aff}}$, returns a ciphertext $C = (\mathbf{C}_1, \mathbf{c}_2)$ such that

$$\mathbf{C}_1 := \mathbf{R} \cdot \mathbf{A} \qquad \mathbf{c}_2 := \mathbf{R} \cdot \mathbf{t},$$

where $\mathbf{R} \leftarrow\!\!\$\, [-\lfloor\sqrt{p}/2\rfloor, \lfloor\sqrt{p}/2\rfloor]^{\ell \times m}$, and $(\mathbf{A}, \mathbf{t}, \mathbf{s}) \leftarrow\!\!\$\, \mathbf{D}_{\mathrm{SS}}(\lambda, n, p, m)$.

Next, we proceed to show indistinguishability of the above defined games.

Claim 19. $\mathbf{G}_0 \approx_s \mathbf{G}_1$.

Proof of claim. The proof is a consequence of leftover-hash lemma (cf. Lemma 2). In fact, basing on that lemma, for $C = (\mathbf{C}_1, \mathbf{c}_2)$ computed by $\mathbf{G}_0$ with

$$\mathbf{C}_1 := \mathbf{R} \cdot \mathbf{A} \qquad \mathbf{c}_2 := \mathbf{R} \cdot \mathbf{t} + \left(\mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor\right) \cdot \mathbf{s} + \mathbf{f} \cdot \left\lfloor \frac{p}{2} \right\rfloor,$$

and $C' = (\mathbf{C}'_1, \mathbf{c}'_2)$ computed by $\mathbf{G}_1$ where

$$\mathbf{C}'_1 := \mathbf{R} \cdot \mathbf{A} - \mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor \qquad \mathbf{c}'_2 := \mathbf{R} \cdot \mathbf{t} + \mathbf{f} \cdot \left\lfloor \frac{p}{2} \right\rfloor,$$

we have that there exists a negligible function $\nu : \mathbb{N} \to [0,1]$ such that

$$\Delta\left((\mathbf{C}_1, \mathbf{Re}(\mathbf{A}, \mathbf{s})); (\mathbf{U}, \mathbf{Re}(\mathbf{A}, \mathbf{s}))\right) \leq \nu(\lambda) \quad \text{and} \quad \Delta\left((\mathbf{C}'_1, \mathbf{Re}(\mathbf{A}, \mathbf{s})); (\mathbf{U}, \mathbf{Re}(\mathbf{A}, \mathbf{s}))\right) \leq \nu(\lambda),$$

where $\mathbf{U} \leftarrow\!\!\$\, \mathbb{Z}_p^{\ell \times n}$. Conditioned on $\mathbf{C}_1$, $\mathbf{C}'_1$, $\mathbf{Re}(\mathbf{A}, \mathbf{s})$, both $\mathbf{c}_2$ and $\mathbf{c}'_2$ are uniquely determined:

$$\mathbf{c}_2 = \mathbf{C}_1\mathbf{s} + \mathbf{Re}(\mathbf{A}, \mathbf{s}) + \left(\mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor\right) \cdot \mathbf{s} + \mathbf{f} \cdot \left\lfloor \frac{p}{2} \right\rfloor$$
$$\mathbf{c}'_2 = \mathbf{C}'_1\mathbf{s} + \mathbf{Re}(\mathbf{A}, \mathbf{s}) + \left(\mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor\right) \cdot \mathbf{s} + \mathbf{f} \cdot \left\lfloor \frac{p}{2} \right\rfloor.$$

Therefore $\mathbf{c}_2$ and $\mathbf{c}'_2$ have the same distribution and we obtain that

$$\Delta((\mathbf{C}_1, \mathbf{c}_2), (\mathbf{C}'_1, \mathbf{c}'_2)) \leq 2\nu(\lambda),$$

so $\mathbf{G}_0$ and $\mathbf{G}_1$ are indistinguishable. □

Claim 20. *For all PPT distinguishers* D*, there exists a negligible function* $\nu_{1,2} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathsf{D}(\mathbf{G}_1(\lambda)) = 1] - \mathbb{P}[\mathsf{D}(\mathbf{G}_2(\lambda)) = 1]| \leq \nu_{1,2}(\lambda)$*.*

Proof of claim. Assume there exists a PPT distinguisher D and a polynomial $p_{1,2}(\cdot)$, such that, for infinitely many values of $\lambda \in \mathbb{N}$, distinguisher D tells apart $\mathbf{G}_1$ and $\mathbf{G}_2$ with probability at least $1/p_{1,2}(\lambda)$. We build a PPT distinguisher D' that breaks the Subset Sum assumption with the same probability, i.e. D' is given a pair $(\mathbf{A}, \mathbf{t})$ as input and is able to distinguish whether this pair was sampled from the Subset Sum distribution $\mathbf{D}_{\mathrm{SS}}(\lambda, n, p, m)$ or uniformly at random. A complete description of $\mathsf{D}'(\mathbf{A}, \mathbf{t})$ follows below.

- Set $pk := (\mathbf{A}, \mathbf{t})$, and forward pk to D.
- Upon input a query $(\mathbf{F}, \mathbf{f})$ to the KDM oracle from D, answer this query as it would be done both in $\mathbf{G}_1$ and $\mathbf{G}_2$. Namely, let $\mathbf{C}_1 := \mathbf{R} \cdot \mathbf{A} + \mathbf{F}\lfloor p/2 \rfloor$, $\mathbf{c_2} := \mathbf{R} \cdot \mathbf{t} + \mathbf{f}\lfloor p/2 \rfloor$, and return $C = (\mathbf{C}_1, \mathbf{c}_2)$ to D. Note that this is possible because in both games the answer to KDM queries can be generated without knowing the secret key.
- Return the guess of D.

For the analysis, note that D' perfectly simulates the view of D. In fact, depending on the public key being Subset Sum distributed or uniformly distributed, the view of D' is identical to either the view in $\mathbf{G}_1$ or the view in $\mathbf{G}_2$. Thus, D' retains the same advantage of D, concluding the proof. □

Claim 21. *For all PPT distinguishers* D*, there exists a negligible function* $\nu_{2,3} : \mathbb{N} \to [0,1]$ *such that* $|\mathbb{P}[\mathsf{D}(\mathbf{G}_2(\lambda)) = 1] - \mathbb{P}[\mathsf{D}(\mathbf{G}_3(\lambda)) = 1]| \leq \nu_{2,3}(\lambda)$*.*

Proof of claim. The proof of indistinguishability between $\mathbf{G}_2$ and $\mathbf{G}_3$ follows from the leftover-hash lemma (cf. Lemma 2) and the Subset Sum assumption. Game $\mathbf{G}_2$ computes ciphertexts $C = (\mathbf{C}_1, \mathbf{c}_2)$ such that

$$\mathbf{C}_1 := \mathbf{R} \cdot \mathbf{A} - \mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor \qquad \mathbf{c}_2 := \mathbf{R} \cdot \mathbf{u} + \mathbf{f} \cdot \left\lfloor \frac{p}{2} \right\rfloor,$$

whereas $\mathbf{G}_3$ computes $C' = (\mathbf{C}_1', \mathbf{c}_2')$ with

$$\mathbf{C}_1' := \mathbf{R} \cdot \mathbf{A} \qquad \mathbf{c}_2' := \mathbf{R} \cdot \mathbf{t}.$$

By Lemma 2 we have that there exists a negligible function $\nu : \mathbb{N} \to [0,1]$ such that

$$\Delta\left((\mathbf{C}_1, \mathbf{Ru}); (\mathbf{U}, \mathbf{Ru})\right) \leq \nu(\lambda) \quad \text{and} \quad \Delta\left((\mathbf{C}_1', \mathbf{Ru}); (\mathbf{U}, \mathbf{Ru})\right) \leq \nu(\lambda),$$

where $\mathbf{U} \leftarrow\!\!\$\, \mathbb{Z}_p^{\ell \times n}$ and $\mathbf{u} \leftarrow\!\!\$\, \mathbb{Z}_p^m$. Furthermore, the components $\mathbf{F} \cdot \lfloor \frac{p}{2} \rfloor$ and $\mathbf{f} \cdot \lfloor \frac{p}{2} \rfloor$—where $\mathbf{F} \in \mathbb{Z}_p^{\ell \times n}$ and $\mathbf{f} \in \mathbb{Z}_p^{\ell}$—represent only a translation for $\mathbf{C}_1$ and $\mathbf{c}_2$, so we have:

$$\mathbf{C}_1 = \mathbf{R} \cdot \mathbf{A} - \mathbf{F} \cdot \left\lfloor \frac{p}{2} \right\rfloor \approx_s \mathbf{R} \cdot \mathbf{A} = \mathbf{C}_1' \qquad \mathbf{c}_2 = \mathbf{R} \cdot \mathbf{u} + \mathbf{f} \cdot \left\lfloor \frac{p}{2} \right\rfloor \approx_s \mathbf{R} \cdot \mathbf{u}.$$

Therefore, the indistinguishability between the distributions $(\mathbf{C}_1, \mathbf{c}_2)$ and $(\mathbf{C}_1', \mathbf{c}_2')$ follows from those approximations and the Subset Sum assumption, whereby $((\mathbf{R} \cdot \mathbf{A}), (\mathbf{R} \cdot \mathbf{u})) \approx_c ((\mathbf{R} \cdot \mathbf{A}), (\mathbf{R} \cdot \mathbf{t}))$, which implies the statement. □

□

5 Concrete Instantiations and Comparisons

In this section we showcase the efficiency improvement due to the adoption of our twist of the Naor-Yung paradigm, when we instantiate the (ROM based) NIZK in the corresponding language using the Fiat-Shamir heuristic [FS86] applied to so-called Sigma-protocols.

Sigma-protocols. Let L be an NP language, with corresponding relation R, and let (P, V) be an interactive argument system for L. We say that $\Sigma = (\mathsf{P}, \mathsf{V})$ is a Sigma-protocol if its transcripts consist of three messages (α, β, γ), with the first message sent by the prover, and with β being the random coin tosses of the verifier. Typically, a Sigma-protocol satisfies the following properties.

Completeness: Transcripts (α, β, γ) generated by a honest prover are accepted by the verifier with overwhelming probability.

Special Soundness: Given two accepting transcripts (α, β, γ) and $(\alpha, \beta', \gamma')$ for a given statement $y \in L$, with $\beta \neq \beta'$, it is possible to extract in polynomial time a value x such that $(y, x) \in R$.

Honest-Verifier Zero-Knowledge: There exists a PPT simulator that, upon input some $y \in L$, outputs transcripts (α, β, γ) that are computationally indistinguishable from honest transcripts resulting from interactions between P and V on common input y.

The Fiat-Shamir heuristic transforms a Sigma-protocol with the above properties into a NIZK argument system, using a hash function H modeled as a random oracle; the main idea is that the prover emulates the verifier by setting $\beta := H(\alpha||y)$. As proven in [FKMV12, Theorem 2], the above transformation yields a simulation-sound NIZK in the random oracle model, provided that Σ satisfies an additional property known as "quasi-unique responses" (a.k.a. strict soundness [Unr12]): No PPT adversary should be able to output a statement $y \in L$ together with two accepting proofs (α, β, γ) and (α, β, γ'), such that $\gamma \neq \gamma'$.

Evaluation. Summarizing the above discussion, to instantiate the NIZK argument in our twist of the Naor-Yung paradigm in the random oracle model, it suffices to give a Sigma-protocol with the properties discussed above, for the language defined in Eq. (1).

Table 1 compares two instantiations of our scheme w.r.t. the original Naor-Yung paradigm, based on two different complexity assumptions: Decisional Diffie-Hellman (DDH) and Quadratic Residuosity (QR). We make the comparison for both cases of CCA security under key-dependent message and key-leakage attacks. The description of the corresponding Sigma-protocols can be found in the following subsections, where we additionally describe an instantiation based on Subset Sum using our PKE scheme from Section 4.

PKE Scheme	Security	Standard NY	Ours	Assumption
BHHO08 [BHHO08]	KDM/LKG	$4\ell + 5$	$2\ell + 4$	DDH
BG10 [BG10]	KDM/LKG	$4\ell + 5$	$2\ell + 4$	QR

Table 1: Comparing two instantiations of our twist of the Naor-Yung paradigm under the DDH and QR assumptions. KDM and LKG stand for CCA security under key-dependent message and key-leakage attacks, respectively. The third and forth columns contain the ciphertext size expressed in group elements or exponents, for the standard Naor-Yung construction and our modified version (respectively). All instantiations are in the random oracle model.

5.1 Instantiantion from Decisional Diffie-Hellman

5.1.1 The PKE Scheme of Boneh, Halevi, Hamburg, and Ostrovsky

We recall the PKE scheme put forward by Boneh, Halevi, Hamburg, and Ostrovsky [BHHO08] (BHHO in what follows). Let $\mathbb{G}$ be a group of prime-order q. For randomly selected generators $g_1, \ldots, g_\ell \leftarrow\$\, \mathbb{G}$, define $\texttt{params} := (\mathbb{G}, g_1, \ldots, g_\ell, q)$. The public key is $pk := h := \prod_{i=i}^{\ell} g_i^{z_i}$ for a secret key $sk := (z_1, ..., z_\ell) \in \mathbb{Z}_q^\ell$. Given the public parameters $\texttt{params}$ and a message $m \in \mathbb{G}$, the encryption algorithm samples a random $r \leftarrow\$\, \mathbb{Z}_q$ and outputs $c = (c_1, ..., c_{\ell+1}) = (g_1^r, \ldots, g_\ell^r, m \cdot h^r)$.

Note that, by setting $\ell = 1$, the BHHO PKE scheme is identical to ElGamal [ElG85], which is CPA-secure under the DDH assumption in $\mathbb{G}$. [BHHO08] showed that, for $\ell = \lceil 3 \log q \rceil$, the scheme is $\mathcal{F}_{\text{aff}}$-KDM-CPA secure under the DDH assumption, where the set $\mathcal{F}_{\text{aff}}$ consists of all affine functions over the secret key space. Naor and Segev [NS09] additionally prove that, for $\ell = 2 + \frac{\Lambda + \omega(\log \lambda)}{\log q}$, the same PKE scheme is CPA-secure under Λ-key-leakage attacks.

On randomness fusion. Consider the following PPT algorithm Rand, taking as input a pair of ciphertexts (c, c') such that $c := (g_1^r, \ldots, g_\ell^r, m \cdot h^r)$ and $c' := (g_1^{r'}, \ldots, g_\ell^{r'}, m' \cdot (h')^{r'})$, and the auxiliary information $\mathsf{aux} := (h, h', (z_1', \ldots, z_\ell'), r', m')$. The algorithm performs the following steps, where all operations are performed in the group $\mathbb{G}$:

(i) For each $i \in [\ell]$, define $\tilde{c}_i := c_i \cdot c_i'$;
(ii) Compute $\tilde{c}_{\ell+1} := c_{\ell+1} \cdot h^{r'}$;
(iii) Compute $\tilde{c}'_{\ell+1} := c'_{\ell+1} \cdot \prod_{i=1}^{\ell} c_i^{z_i'}$;
(iv) Return $\tilde{c} = (\tilde{c}_1, \ldots, \tilde{c}_\ell, \tilde{c}_{\ell+1})$ and $\tilde{c}' = (\tilde{c}_1, \ldots, \tilde{c}_\ell, \tilde{c}'_{\ell+1})$.

One can easily see that the pair of ciphertexts returned by Rand is perfectly distributed to a pair of BHHO encryptions with common (uniform) randomness $r^* := r + r' \bmod q$. For $i \in [\ell]$, $\tilde{c}_i = c_i \cdot c_i' = g_i^{r+r'}$, $\tilde{c}_{\ell+1} = c_{\ell+1} \cdot h^{r'} = m \cdot h^{r+r'}$ and

$$\tilde{c}'_{\ell+1} = c'_{\ell+1} \cdot \prod_{i=1}^{\ell} c_i^{z_i'} = m' \cdot h^{r'} \cdot \left(\prod_{i=1}^{\ell} g_i^{z_i'} \right)^{r} = m' \cdot h^{r+r'}.$$

5.1.2 The Protocol

In order to instantiate our twist of the Naor-Yung paradigm with the BHHO PKE scheme, we need to construct a Sigma-protocol for the following language:

$$L_{\text{NY}}^{\text{BHHO}} = \{(h, h', c, c') : \exists r \in \mathbb{Z}_q, m \in \mathbb{G} \text{ s.t. } c = (g_1^r, \ldots, g_\ell^r, m \cdot h^r), c' = (g_1^r, \ldots, g_\ell^r, m \cdot (h')^r)\},$$

where $c := (c_1, ..., c_\ell, c_{\ell+1})$ and $c' := (c_1, ..., c_\ell, c'_{\ell+1})$ are BHHO encryptions with common randomness r, using independent public keys $pk := h$ and $pk' := h'$, and common public parameters $\texttt{params} = (\mathbb{G}, g_1, \ldots, g_\ell, q)$. The protocol $\Sigma = (\mathsf{P}, \mathsf{V})$ is described below:

- P chooses $s \leftarrow\$\, \mathbb{Z}_q$ and defines the commitment to be $\alpha := (\alpha_1, \ldots, \alpha_\ell, \alpha_{\ell+1}) := (g_1^s, \ldots, g_\ell^s, (h/h')^s)$.
- V replies with a random $\beta \leftarrow\$\, \mathbb{Z}_q$.
- P computes the response $\gamma := s - \beta r$.
- Given a transcript (α, β, γ) and some statement (h, h', c, c') the verifier accepts it if and only if $\alpha_i = g_i^\gamma \cdot c_i^\beta$ (for all $i \in [\ell]$), and also $\alpha_{\ell+1} = (h/h')^\gamma \cdot (c_{\ell+1}/c'_{\ell+1})^\beta$.

Next, we argue that the above protocol meets the required properties.

Completeness. Follows by inspection, as

$$g_i^{\gamma} \cdot c_i^{\beta} = g_i^{s-\beta r} \cdot c_i^{\beta} = g_i^{s-\beta r} \cdot g_i^{\beta r} = g_i^{s} = \alpha_i$$
$$(h/h')^{\gamma} \cdot (c_{\ell+1}/c'_{\ell+1})^{\beta} = (h/h')^{s-\beta r} \cdot (h/h')^{\beta r} = (h/h')^{s} = \alpha_{\ell+1}.$$

Special soundness. Let (α, β, γ) and $(\alpha, \beta', \gamma')$ be two accepting transcripts for some $y \in L_{\mathrm{NY}}^{\mathrm{BHHO}}$, such that $\beta \neq \beta'$. This means that, for all $i \in [\ell]$, $\alpha_i = g_i^{\gamma} \cdot c_i^{\beta}$, and thus $c_i = g_i^{r}$ for $r = (\gamma - \gamma')(\beta' - \beta)^{-1}$. Note that for the same value of r is also holds that $c_{\ell+1}/c'_{\ell+1} = (h/h')^{r}$, and thus r is a valid witness for $y \in L_{\mathrm{NY}}^{\mathrm{BHHO}}$.

HVZK. Consider the simulator that, upon input a statement $y := (h, h', (c_1, \ldots, c_{\ell+1}), (c'_1, \ldots, c'_{\ell+1}))$, first samples $\beta \leftarrow\!\!\$\, \mathbb{Z}_q$, $\gamma \leftarrow\!\!\$\, \mathbb{Z}_q$, and then defines $\alpha = (\alpha_1, \ldots, \alpha_\ell, \alpha_{\ell+1})$ such that $\alpha_i := g_i^{\gamma} \cdot c_i^{\beta}$ (for all $i \in [\ell]$) and $\alpha_{\ell+1} := (h/h')^{\gamma} \cdot (c_{\ell+1}/c'_{\ell+1})^{\beta}$. It is easy to see that the above yields an identical distribution to the one of honest transcripts (α, β, γ).

Quasi-unique responses. Assume that, for $y := (h, h', (c_1, \ldots, c_{\ell+1}), (c'_1, \ldots, c'_{\ell+1})) \in L_{\mathrm{NY}}^{\mathrm{BHHO}}$, there exist two accepting proofs (α, β, γ) and (α, β, γ'). This means in particular that $(h/h')^{\gamma} = (h/h')^{\gamma'}$, and thus $\gamma \equiv \gamma' \pmod q$.

5.2 Instantiation from Quadratic Residuosity

5.2.1 The PKE Scheme of Brakerski and Goldwasser

We recall the PKE encryption scheme put forward by Brakerski and Goldwasser [BG10] (BG in what follows). Let $\mathbb{G}_U = \mathbb{G}_M \times \mathbb{G}_L$ be a group such that $\mathbb{G}_M$ is cyclic, and the orders of $\mathbb{G}_M$ and $\mathbb{G}_L$ are relatively prime and denoted by M and L (respectively). For randomly selected generators $g_1, \ldots, g_\ell \leftarrow\!\!\$\, \mathbb{G}_L$, define $\mathtt{params} := (\mathbb{G}_U, \mathbb{G}_M, \mathbb{G}_L, g_1, \ldots, g_\ell, M, L, T)$ with $T \geq M \cdot L$. The public key is $pk := g_0 := \prod_{i=i}^{\ell} g_i^{-z_i}$ for a secret key $sk := (z_1, ..., z_\ell) \in \{0,1\}^{\ell}$. Given the public parameters $\mathtt{params}$ and a message $m \in \mathbb{G}_M$, the encryption algorithm samples a random $r \leftarrow\!\!\$\, [T^2]$ and outputs $c = (c_1, ..., c_{\ell+1}) = (g_1^{r}, \ldots, g_\ell^{r}, m \cdot g_0^{r})$.

[BG10] showed that, for $\ell = \log L + \omega(\log \lambda)$, the scheme is $\mathcal{F}_{\mathrm{aff}}$-KDM-CPA secure under the Subgroup Indistinguishability assumption,[5] where the set $\mathcal{F}_{\mathrm{aff}}$ consists of all affine functions over the secret key space. For $\ell = \Lambda + \log(ML) + \omega(\log \lambda)$, the same PKE scheme is also CPA-secure under Λ-key-leakage attacks.

On randomness fusion. The proof that the BG PKE scheme meets the randomness fusion property of Definition 5 follows along the same lines of BHHO.

5.2.2 The Protocol

In order to instantiate our twist of the Naor-Yung paradigm with the BG PKE scheme, we need to construct a Sigma-protocol for the following language:

$$L_{\mathrm{NY}}^{\mathrm{BG}} = \{(g_0, g'_0, c, c') : \exists r \in [T^2], m \in \mathbb{G}_M \text{ s.t. } c = (g_1^{r}, \ldots, g_\ell^{r}, m \cdot g_0^{r}), c' = (g_1^{r}, \ldots, g_\ell^{r}, m \cdot (g'_0)^{r})\},$$

where $c := (c_1, ..., c_\ell, c_{\ell+1})$ and $c' := (c_1, ..., c_\ell, c'_{\ell+1})$ are BG encryptions with common randomness r, using independent public keys $pk := g_0$ and $pk' := g'_0$, and common public parameters $\mathtt{params} = (\mathbb{G}_U, \mathbb{G}_M, \mathbb{G}_L, g_1, \ldots, g_\ell, M, L, T)$. The protocol $\Sigma = (\mathsf{P}, \mathsf{V})$ is described below:

[5] This assumption states that random elements from $\mathbb{G}_U$ are computationally indistinguishable from random elements from $\mathbb{G}_L$, and it includes the Quadratic Residuosity and Paillier's Decisional Composite Residuosity assumptions as a special case [BG10].

- P chooses $s \leftarrow_{\$} [T^2]$ and defines the commitment to be $\alpha := (\alpha_1, \ldots, \alpha_\ell, \alpha_{\ell+1}) := (g_1^s, \ldots, g_\ell^s, (g_0/g_0')^s)$.
- V replies with a random $\beta \leftarrow_{\$} [T^2]$.
- P computes the response $\gamma := s - \beta r$.
- Given a transcript (α, β, γ) and some statement (g_0, g_0', c, c'), the verifier accepts it if and only if $\alpha_i = g_i^\gamma \cdot c_i^\beta$ (for all $i \in [\ell]$), and also $\alpha_{\ell+1} = (g_0/g_0')^\gamma \cdot (c_{\ell+1}/c_{\ell+1}')^\beta$.

The proof that the above Sigma-protocol satisfies completeness, special soundness, HVZK, and quasi-unique responses is similar to the case of BHHO and is therefore omitted.

5.3 Instantiation from Subset Sum

Next, we turn to our PKE scheme based on Subset Sum from Section 4.

On randomness fusion. We show that the scheme from Section 4 fulfills a slight variation of the randomness fusion property of Definition 5. First, we need that both public keys use the same component $\mathbf{A}$. Since $\mathbf{A}$ is independent of the secret key, this is not an issue and therefore $\mathbf{A}$ could be seen as a public parameter. Further, we need some leakage on the randomness $\mathbf{R}$ of a ciphertext. Exploiting the leftover-hash lemma, it is easy to see that our PKE scheme from Section 4 is still secure when for a ciphertext $\mathbf{C}_1 = \mathbf{RA}$, $\mathbf{c}_2 = \mathbf{R}(\mathbf{As} + \mathbf{e}(\mathbf{A}, \mathbf{s})) + \mathbf{m}\lfloor p/2 \rfloor$ the value $\mathbf{Re}(\mathbf{A}, \mathbf{s}')$ is leaked, as long as $\mathbf{s}'$ is independent of $\mathbf{s}$. By the leftover-hash lemma (cf. Lemma 2) there exists a negligible function $\nu : \mathbb{N} \to [0,1]$ such that

$$\Delta\big((\mathbf{RA}, \mathbf{Re}(\mathbf{A},\mathbf{s}), \mathbf{Re}(\mathbf{A},\mathbf{s}')); (\mathbf{U}, \mathbf{Re}(\mathbf{A},\mathbf{s}), \mathbf{Re}(\mathbf{A},\mathbf{s}'))\big) \le \nu(\lambda)$$
$$\Delta\big((\mathbf{RA}, \mathbf{Ra}, \mathbf{Re}(\mathbf{A},\mathbf{s}')); (\mathbf{U}, \mathbf{u}, \mathbf{Re}(\mathbf{A},\mathbf{s}'))\big) \le \nu(\lambda),$$

for uniform $\mathbf{U}$, $\mathbf{a}$ and $\mathbf{u}$. Therefore, the leakage $\mathbf{Re}(\mathbf{A}, \mathbf{s}')$ increases the statistical distance of the component $\mathbf{C}_1$ of a normal ciphertext from uniform only by a negligible term, and the same also holds for a uniform "ciphertext" $(\mathbf{C}_1, \mathbf{c}_2)$. This is sufficient for Theorem 4 which guarantees the security of the PKE scheme.

Given leakage $\mathbf{Re}(\mathbf{A}, \mathbf{s}')$, ciphertexts $C = (\mathbf{C}_1, \mathbf{c}_2) = (\mathbf{RA}, \mathbf{Rt} + \mathbf{m}\lfloor p/2 \rfloor)$, $C' = (\mathbf{C}_1', \mathbf{c}_2') = (\mathbf{R}'\mathbf{A}, \mathbf{R}'\mathbf{t}' + \mathbf{m}'\lfloor p/2 \rfloor)$, and auxiliary information $\mathsf{aux} := (\mathbf{t}, \mathbf{t}', \mathbf{s}', \mathbf{R}', \mathbf{m}', \mathbf{Re}(\mathbf{A}, \mathbf{s}))$, we can compute

$$\tilde{C} := (\mathbf{C}_1 + \mathbf{C}_1', \mathbf{c}_2 + \mathbf{R}'\mathbf{t}')$$

which is a ciphertext for randomness $\mathbf{R} + \mathbf{R}'$. For computing $\tilde{C}'$ we need to exploit the knowledge of the leakage and compute

$$\tilde{C}' := (\mathbf{C}_1 + \mathbf{C}_1', \mathbf{c}_2' + \mathbf{C}_1\mathbf{s} + \mathbf{Re}(\mathbf{A}, \mathbf{s}')),$$

which is also a ciphertext for randomness $\mathbf{R} + \mathbf{R}'$. Clearly, $\mathbf{R} + \mathbf{R}'$ does not have the same distribution as $\mathbf{R}$. Hence, a ciphertext using randomness $\mathbf{R}$ is not statistically close to a ciphertext with randomness $\mathbf{R} + \mathbf{R}'$. We modify the encryption algorithm of the PKE scheme from Section 4 such that it uses randomness $\mathbf{R}^* := \mathbf{R} + \mathbf{R}'$ instead of $\mathbf{R}$ and call it $\Pi' = (\mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec})$. The correctness of the decryption will hold, when sampling $\mathbf{R}$, $\mathbf{R}'$ from $[-\lfloor\sqrt{p}/4\rfloor, \lfloor\sqrt{p}/4\rfloor]^{\ell \times m}$. The leftover-hash lemma (cf. Lemma 2) still yields

$$\Delta\left((\mathbf{A}, \mathbf{u}_1, \mathbf{u}_2, \mathbf{R}^*\mathbf{A}, \mathbf{R}^*\mathbf{u}_1, \mathbf{R}^*\mathbf{u}_2); (\mathbf{A}, \mathbf{u}_1, \mathbf{u}_2, \mathbf{B})\right) \le \frac{\ell}{2}\sqrt[4]{2^{2(n+2)\log p + m - m\log(p-1)}}.$$

Thus, one could still choose $m = 3n$ to obtain a negligible statistical distance.

Finally, we obtain a slightly weaker randomness fusion property where re-randomized ciphertexts of PKE Π are statistically close to ciphertexts of Π'. This variant is indeed sufficient to apply the Naor-Yung paradigm with shared randomness (cf. Theorem 1).

The protocol. In order to instantiate our twist of the Naor-Yung paradigm with the Subset Sum based PKE scheme from Section 4.2, we need to construct a Sigma-protocol for the following language:

$$L_{\mathrm{NY}}^{\mathrm{BMV}} = \{(pk, pk', C, C') : \exists \mathbf{R} \in [-\lfloor\sqrt{p}/2\rfloor, \lfloor\sqrt{p}/2\rfloor]^{\ell\times m}, \mathbf{m} \in \mathbb{Z}_2^{\ell} \\ \text{s.t. } C = (\mathbf{RA}, \mathbf{Rt} + \mathbf{m}\lfloor p/2\rfloor), C' = (\mathbf{RA}, \mathbf{Rt}' + \mathbf{m}\lfloor p/2\rfloor)\},$$

where $C := (\mathbf{C}_1, \mathbf{c}_2)$ and $C' := (\mathbf{C}_1', \mathbf{c}_2')$ are encryptions with common randomness $\mathbf{R}$, using independent public keys $pk := (\mathbf{A}, \mathbf{t})$ and $pk' := (\mathbf{A}, \mathbf{t}')$, and common public parameters $\texttt{params} = (\mathbf{A}, n, p, m, \ell)$.[6]

It is easy to see that the above language can be equivalently defined as follows:

$$L_{\mathrm{NY}}^{\mathrm{BMV}} = \{(\mathbf{A}', \mathbf{B}) \in \mathbb{Z}_p^{m\times(n+1)} \times \mathbb{Z}_p^{\ell\times(n+1)} : \exists \mathbf{R} \in \mathbb{Z}_p^{\ell\times m} \text{ s.t. } \mathbf{RA}' = \mathbf{B} \text{ and } \|\mathbf{R}\|_\infty \le \lfloor\sqrt{p}/2\rfloor\},$$

where $\mathbf{A}' := (\mathbf{A}, \mathbf{t} - \mathbf{t}')$. This language can be seen as syndrome decoding or Knapsack LWE [MM11] over $\mathbb{Z}_p$. Stern gave a Sigma-protocol for this language over $\mathbb{Z}_2$, where the norm is the Hamming weight [Ste93]. Unfortunately, the protocol heavily relies on permutations over $\mathbb{Z}_2^n$, which preserve the Hamming weight. To extend the protocol to $\mathbb{Z}_q$ for non-binary $\mathbf{R}$, one would need to extend the permutation to an isometry for the desired norm that keeps the algebraic structure.

Therefore, we design the following Sigma-protocol $\Sigma = (\mathsf{P}, \mathsf{V})$, based on rejection sampling used by Lyubashevsky [Lyu09]. This principle was improved by Ducas *et al.* [DDLL13], and also applied by, e.g., Damgard *et al.* [DPSZ12]. Rejection sampling is a method to sample from an arbitrary target probability distribution f, given a source bound to a different probability distribution g. Here, a sample x is picked from g and is accepted with probability $f(x)/(M\cdot g(x))$ where M is some positive real; in case of rejection, the process is restarted. If $f(x) \le Mg(x)$ for all x, it is not hard to prove that this procedure produces exactly the distribution of f. Furthermore, M is the expected number of times the procedure will need to be restarted, and it is crucial to keep M as small as possible. Moreover, given that rejection sampling can be interpreted as sampling a random point (x_i, y_i) in the area under the distribution $M \cdot g$ and accepting if and only if $y_i \le f(x_i)$, reducing the area between g and f will reduce M.

The following interactive protocol is designed to provide a proof π only about a statement $\mathbf{b} := \mathbf{r}\mathbf{A}'$, for witness $\mathbf{r} \in \mathbf{R}$; however, repeating it ℓ times (which is the number of rows into the actual witness $\mathbf{R}$ matrix), we obtain a proof system for the whole target language. It works as described below, for a witness with norm $\|\mathbf{r}\|_\infty \le v$.

- P picks a vector $\mathbf{y}$ according to the distribution D_d^m, where D_d^m is the discrete uniform distribution over $[-d, d]$; hence, it sets $\mathbf{A}' := (\mathbf{A}, \mathbf{t} - \mathbf{t}')$, as defined for the language $L_{\mathrm{NY}}^{\mathrm{BMV}}$, to compute $\boldsymbol{\alpha} := \mathbf{y}\mathbf{A}'$.
- V replies with a random bit $\beta \leftarrow\$ \{0, 1\}$.
- P computes $\boldsymbol{\gamma} := \beta\mathbf{r} + \mathbf{y}$; if $\|\boldsymbol{\gamma}\|_\infty \le d - v$, then it sends $\boldsymbol{\gamma}$ and otherwise aborts.
- Given a transcript $(\boldsymbol{\alpha}, \beta, \boldsymbol{\gamma})$, the verifier accepts it if and only if:
  1. $\|\boldsymbol{\gamma}\|_\infty \le d - v$;
  2. $\boldsymbol{\gamma}\mathbf{A}' = \boldsymbol{\alpha} + \beta\mathbf{b}$.

Note that we multiplied $\mathbf{y}$ with the matrix $\mathbf{A}'$ in order to obtain the commitment $\boldsymbol{\alpha}$, because for $\boldsymbol{\alpha} = \mathbf{y}$ and $\beta = 1$ the verifier could obtain the witness $\mathbf{r}$ trivially by subtracting $\boldsymbol{\gamma} - \boldsymbol{\alpha}$. The distribution of $\boldsymbol{\gamma}$ depends on the distribution of $\beta\mathbf{r}$, and thus on the distribution of $\mathbf{r}$. In fact,

[6] Strictly speaking, each public key should contain a different matrix $\mathbf{A}$; however the above variant is still secure and yields a smaller key and proof size.

the distribution of $\boldsymbol{\gamma}$ is almost D_d^m shifted by the vector $\mathbf{r}$ when $\beta = 1$; otherwise $\boldsymbol{\gamma} = \mathbf{y}$ and the distribution of $\boldsymbol{\gamma}$ is exactly D_d^m. Instead, the target distribution for γ is D_t^m, that denotes the discrete uniform distribution over $[-d+v, d-v]^m$. For simplicity, we first analyze the case $m = 1$. When $\beta = 0$, the distribution g behaves as the uniform distribution over $[-d, d]$, with probability mass function:

$$g_1(x) = \frac{1}{2d+1}. \tag{5}$$

Meanwhile, when $\beta = 1$, we need the convolution between $g_1(x)$ and the discrete uniform distribution over $[-v, v]$, from which both $\mathbf{y}$ and $\mathbf{r}$ are sampled. The possible outcomes (from the joint distribution) are obtained by multiplying $(2d+1)\cdot(2v+1)$, where $2d+1$ is the amount of possible values for $\mathbf{y}$ and $2v+1$ is the amount of possible values for $\mathbf{r}$. Hence, inside the target interval $[-d+v, d-v]$, the accepting $\boldsymbol{\gamma}$ can assume $2v+1$ combinations of outcomes from the joint distribution, with probability $\frac{2v+1}{(2d+1)\cdot(2v+1)} = \frac{1}{2d+1}$. Moreover, in the intervals $[-d-v, -d+v]$ and $[d-v, d+v]$ we have a trapezoidal behavior, so that we obtain the following probability mass function:

$$g_2(x) = \begin{cases} u(x+d+v+1) & \text{if } -d-v \leq x < -d+v \\ \frac{1}{2d+1} & \text{if } -d+v \leq x < d-v \\ u(d+v-x+1) & \text{if } d-v \leq x < d+v \end{cases}$$

where $u = ((2d+1)\cdot(2v+1))^{-1}$. It follows that $\boldsymbol{\gamma}$ is sampled from the distribution $\frac{1}{2}g_1(x)+\frac{1}{2}g_2(x)$, where both $g_1(x)$ and $g_2(x)$ share the same probability mass function $\frac{1}{2d+1}$ within the interval $[-d+v, d-v]$.

As stated before, the target distribution D_t^m is the discrete uniform distribution over $[-d+v, d-v]$; therefore, the probability mass function of the distribution f is:

$$f(x) = \frac{1}{2(d-v)+1}.$$

With the aim of providing a bound for d, given that $f(x)/g(x) \leq M$, we compute $f(x)/g(x) = \frac{2d+1}{2(d-v)+1}$, which is always zero, except in the interval $[-d+v, d-v]$. Hence, the probability of rejecting is

$$\left(1 - \frac{1}{M}\right) = \frac{2v}{2d+1}.$$

A union bound yields $\frac{2mv}{2d+1}$ in the general case, i.e., $m \in \mathbb{N}$. We set $d - v \leq \frac{\lfloor\sqrt{p}/2\rfloor}{2}$. Further, for, e.g., $v = \Theta(n \log^2 n)$, $d = \Theta(mn \log^2 n)$, $p = \Theta(n^4 \log^4 n)$ and $m = 4n$ for suitable constants, the probability of rejecting is a constant. Therefore the protocol runs in expected polynomial time.

Given the above analysis of the rejection sampling, we can easily prove the special soundness and HVZK properties of our interactive protocol (the correctness property is easily verified). In particular:

- **Special Soundness:** Let $(\boldsymbol{\alpha}, \beta, \boldsymbol{\gamma})$ and $(\boldsymbol{\alpha}, \beta', \boldsymbol{\gamma}')$ be two accepting transcripts, where $\boldsymbol{\gamma} := \mathbf{y} + \beta\mathbf{r}$, $\boldsymbol{\gamma'} := \mathbf{y} + \beta'\mathbf{r}$, and $\beta \neq \beta'$. We can compute

  $$\boldsymbol{\gamma}\mathbf{A}' - \boldsymbol{\gamma}'\mathbf{A}' = (\boldsymbol{\gamma} - \boldsymbol{\gamma}')\mathbf{A}' = (\beta - \beta')\mathbf{b},$$

  where we used the fact that $\boldsymbol{\gamma}\mathbf{A}' = \boldsymbol{\alpha} + \beta\mathbf{b}$ and $\boldsymbol{\gamma}'\mathbf{A}' = \boldsymbol{\alpha} + \beta'\mathbf{b}$. It follows that either $\boldsymbol{\gamma} - \boldsymbol{\gamma}'$ or $\boldsymbol{\gamma}' - \boldsymbol{\gamma}$ is the wanted witness, with magnitude $\|\boldsymbol{\gamma} - \boldsymbol{\gamma}'\|_\infty \leq 2(d-v) \leq \lfloor\sqrt{p}/2\rfloor = \Theta(n^2 \log^2 n)$.

- **HVZK:** Firstly, the simulator samples $\boldsymbol{\gamma} \leftarrow_{\$} D_t^m$ to obtain, uniformly at random, an accepting response. Hence, given that $\boldsymbol{\gamma}\mathbf{A}' = \boldsymbol{\alpha} + \beta\mathbf{b}$, and that β is provided as input to the simulator, we can obtain the last value of the simulation by setting $\boldsymbol{\alpha} := \boldsymbol{\gamma}\mathbf{A}' - \beta\mathbf{b}$. It is easy to see that this simulation strategy yields an identical distribution to the one of honest transcripts $(\boldsymbol{\alpha}, \beta, \boldsymbol{\gamma})$.

A drawback of our protocol is that the extracted witness is by a factor n larger than the witness used to perform the protocol. In order to be compatible with the KDM secure PKE scheme from Section 4.2, we need to choose m and p somewhat larger such that, on the one hand, the correctness holds for an extracted witness (i.e., a secret key of norm $\sqrt{p}$) and, on the other hand, security holds for a secret key of norm $n \log^2 n < \sqrt{p} \approx n^2 \log^2 n$. Fortunately, the leftover-hash lemma still applies for this parameter choice, such that we can still rely on Lemma 2. Hence, we also obtain security for a secret key of smaller norm (i.e., lower entropy). As a consequence, the security relates to a Subset Sum instance of a smaller density, and therefore to a stronger hardness assumption.

6 Conclusion and Open Problems

We have studied a twist of the classical Naor-Yung paradigm [NY90] to boost CPA security to CCA security, both under key-dependent message and key-leakage attacks. The twist consists in having the two ciphertexts in the Naor-Yung PKE scheme share the same randomness.

In order to prove security, we require the underlying CPA-secure PKE scheme to satisfy an additional property. The main benefit of our approach is that one can instantiate the NIZK in the Naor-Yung PKE more efficiently, as we have explored in the random oracle model. We have also constructed a new PKE scheme with KDM-CPA security under the Subset Sum assumption, and showed that such a scheme can be used within our paradigm.

Open problems include to construct a PKE scheme with CPA security under key-leakage attacks directly based on Subset Sum, or alternatively to show that our construction additionally satisfies this property.[7] Also, it would be interesting to analyze KDM security of our scheme with multiple keys, and to construct a PKE scheme with KDM-CCA security directly based on the Subset Sum assumption in the standard model, without relying on NIZK.

References

[ACPS09] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *CRYPTO*, pages 595–618, 2009.

[ADW09a] Joël Alwen, Yevgeniy Dodis, and Daniel Wichs. Leakage-resilient public-key cryptography in the bounded-retrieval model. In *CRYPTO*, pages 36–54, 2009.

[ADW09b] Joël Alwen, Yevgeniy Dodis, and Daniel Wichs. Survey: Leakage resilience and the bounded retrieval model. In *ICITS*, pages 1–18, 2009.

[AGV09] Adi Akavia, Shafi Goldwasser, and Vinod Vaikuntanathan. Simultaneous hardcore bits and cryptography against memory attacks. In *TCC*, pages 474–495, 2009.

[7] The PKE scheme of [LPS10] only achieves a weak for of leakage resilience, where the leakage cannot depend on the public key.

[AP11] Joël Alwen and Chris Peikert. Generating shorter bases for hard random lattices. *Theory Comput. Syst.*, 48(3):535–553, 2011.

[App11] Benny Applebaum. Key-dependent message security: Generic amplification and completeness. In *EUROCRYPT*, pages 527–546, 2011.

[App14] Benny Applebaum. Key-dependent message security: Generic amplification and completeness. *J. Cryptology*, 27(3):429–451, 2014.

[BBS03] Mihir Bellare, Alexandra Boldyreva, and Jessica Staddon. Randomness re-use in multi-recipient encryption schemeas. In *PKC*, pages 85–99, 2003.

[BDPR98] Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. Relations among notions of security for public-key encryption schemes. In *CRYPTO*, pages 26–45, 1998.

[BFW15] David Bernhard, Marc Fischlin, and Bogdan Warinschi. Adaptive proofs of knowledge in the random oracle model. In *PKC*, pages 629–649, 2015.

[BFW16] David Bernhard, Marc Fischlin, and Bogdan Warinschi. On the hardness of proving CCA-security of signed ElGamal. In *PKC*, pages 47–69, 2016.

[BG10] Zvika Brakerski and Shafi Goldwasser. Circular and leakage resilient public-key encryption under subgroup indistinguishability - (or: Quadratic residuosity strikes back). In *CRYPTO*, pages 1–20, 2010.

[BGK11] Zvika Brakerski, Shafi Goldwasser, and Yael Tauman Kalai. Black-box circular-secure encryption beyond affine functions. In *TCC*, pages 201–218, 2011.

[BGV12] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) Fully homomorphic encryption without bootstrapping. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 309–325. ACM, 2012.

[BHHO08] Dan Boneh, Shai Halevi, Michael Hamburg, and Rafail Ostrovsky. Circular-secure encryption from decision Diffie-Hellman. In *CRYPTO*, pages 108–125, 2008.

[BJLM13] Daniel J. Bernstein, Stacey Jeffery, Tanja Lange, and Alexander Meurer. Quantum algorithms for the subset-sum problem. In *PQCrypto*, pages 16–33, 2013.

[Ble98] Daniel Bleichenbacher. Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1. In *CRYPTO*, pages 1–12, 1998.

[BMV16] Silvio Biagioni, Daniel Masny, and Daniele Venturi. Naor-Yung paradigm with shared randomness and applications. In *SCN*, pages 62–80, 2016.

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *ACM CCS*, pages 62–73, 1993.

[BRS02] John Black, Phillip Rogaway, and Thomas Shrimpton. Encryption-scheme security in the presence of key-dependent messages. In *SAC*, pages 62–75, 2002.

[BSW13] Elette Boyle, Gil Segev, and Daniel Wichs. Fully leakage-resilient signatures. *J. Cryptology*, 26(3):513–558, 2013.

[CCS09] Jan Camenisch, Nishanth Chandran, and Victor Shoup. A public key encryption scheme secure against key dependent chosen plaintext and adaptive chosen ciphertext attacks. In *EUROCRYPT*, pages 351–368, 2009.

[CDTV16] Sandro Coretti, Yevgeniy Dodis, Björn Tackmann, and Daniele Venturi. Non-malleable encryption: Simpler, shorter, stronger. In *TCC*, pages 306–335, 2016.

[CGH04] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited. *Journal of the ACM (JACM)*, 51(4):557–594, 2004.

[CL01] Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *EUROCRYPT*, pages 93–118, 2001.

[CS98] Ronald Cramer and Victor Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *CRYPTO*, pages 13–25, 1998.

[DDLL13] Léo Ducas, Alain Durmus, Tancrède Lepoint, and Vadim Lyubashevsky. Lattice signatures and bimodal gaussians. In *CRYPTO*, pages 40–56, 2013.

[DDN91] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography (extended abstract). In *ACM STOC*, pages 542–552, 1991.

[DDV10] Francesco Davì, Stefan Dziembowski, and Daniele Venturi. Leakage-resilient storage. In *SCN*, pages 121–137, 2010.

[DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Trans. Information Theory*, 22(6):644–654, 1976.

[DHLW10] Yevgeniy Dodis, Kristiyan Haralambiev, Adriana López-Alt, and Daniel Wichs. Efficient public-key cryptography in the presence of key leakage. In *ASIACRYPT*, pages 613–631, 2010.

[Döt15] Nico Döttling. Low noise LPN: KDM secure public key encryption and sample amplification. In *PKC*, pages 604–626, 2015.

[DP08] Stefan Dziembowski and Krzysztof Pietrzak. Leakage-resilient cryptography. In *IEEE FOCS*, pages 293–302, 2008.

[DPSZ12] Ivan Damgård, Valerio Pastro, Nigel P. Smart, and Sarah Zakarias. Multiparty computation from somewhat homomorphic encryption. In *CRYPTO*, pages 643–662, 2012.

[DV14] Özgür Dagdelen and Daniele Venturi. A second look at Fischlin's transformation. In *AFRICACRYPT*, pages 356–376, 2014.

[ElG85] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Information Theory*, 31(4):469–472, 1985.

[FHV13] Sebastian Faust, Carmit Hazay, and Daniele Venturi. Outsourced pattern matching. In *ICALP*, pages 545–556, 2013.

[Fis05] Marc Fischlin. Communication-efficient non-interactive proofs of knowledge with online extractors. In *CRYPTO*, pages 152–168, 2005.

[FKMV12] Sebastian Faust, Markulf Kohlweiss, Giorgia Azzurra Marson, and Daniele Venturi. On the non-malleability of the Fiat-Shamir transform. In *INDOCRYPT*, pages 60–79, 2012.

[FMV16] Sebastian Faust, Daniel Masny, and Daniele Venturi. Chosen-ciphertext security from subset sum. In *PKC*, pages 35–46, 2016.

[FNV15] Antonio Faonio, Jesper Buus Nielsen, and Daniele Venturi. Mind your coins: Fully leakage-resilient signatures with graceful degradation. In *ICALP*, pages 456–468, 2015.

[FO99] Eiichiro Fujisaki and Tatsuaki Okamoto. How to enhance the security of public-key encryption at minimum cost. In *PKC*, pages 53–68, 1999.

[FS86] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO*, pages 186–194, 1986.

[Gen09] Craig Gentry. *A fully homomorphic encryption scheme*. PhD thesis, Stanford University, 2009.

[GHV12] David Galindo, Javier Herranz, and Jorge Villar. Identity-based encryption with master key-dependent message security and leakage-resilience. In *European Symposium on Research in Computer Security*, pages 627–642, 2012.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[Hof13] Dennis Hofheinz. Circular chosen-ciphertext security with compact ciphertexts. In *EUROCRYPT*, pages 520–536, 2013.

[IN96] Russell Impagliazzo and Moni Naor. Efficient cryptographic schemes provably as secure as subset sum. *J. Cryptology*, 9(4):199–216, 1996.

[KJJ99] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In *CRYPTO*, pages 388–397, 1999.

[KMHT16] Fuyuki Kitagawa, Takahiro Matsuda, Goichiro Hanaoka, and Keisuke Tanaka. On the key dependent message security of the Fujisaki-Okamoto constructions. In *PKC*, pages 99–129, 2016.

[Koc96] Paul C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *CRYPTO*, pages 104–113, 1996.

[KV09] Jonathan Katz and Vinod Vaikuntanathan. Signature schemes with bounded leakage resilience. In *ASIACRYPT*, pages 703–720, 2009.

[LPS10] Vadim Lyubashevsky, Adriana Palacio, and Gil Segev. Public-key cryptographic primitives provably as secure as subset sum. In *TCC*, pages 382–400, 2010.

[Lyu09] Vadim Lyubashevsky. Fiat-Shamir with aborts: Application to lattice and factoring-based signatures. In *ASIACRYPT*, pages 598–616, 2009.

[MM11] Daniele Micciancio and Petros Mol. Pseudorandom knapsacks and the sample complexity of LWE search-to-decision reductions. In *CRYPTO*, pages 465–484, 2011.

[MPS16] Antonio Marcedone, Rafael Pass, and Abhi Shelat. Bounded KDM security from iO and OWF. In *International Conference on Security and Cryptography for Networks*, pages 571–586, 2016.

[MRS88] Silvio Micali, Charles Rackoff, and Bob Sloan. The notion of security for probabilistic cryptosystems. *SIAM J. Comput.*, 17(2):412–426, 1988.

[NS09] Moni Naor and Gil Segev. Public-key cryptosystems resilient to key leakage. In *CRYPTO*, pages 18–35, 2009.

[NS12] Moni Naor and Gil Segev. Public-key cryptosystems resilient to key leakage. *SIAM J. Comput.*, 41(4):772–814, 2012.

[NVZ14] Jesper Buus Nielsen, Daniele Venturi, and Angela Zottarel. Leakage-resilient signatures with graceful degradation. In *PKC*, pages 362–379, 2014.

[NY90] Moni Naor and Moti Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *ACM STOC*, pages 427–437, 1990.

[PSV07] Rafael Pass, Abhi Shelat, and Vinod Vaikuntanathan. Relations among notions of non-malleability for encryption. In *ASIACRYPT*, pages 519–535, 2007.

[QS01] Jean-Jacques Quisquater and David Samyde. Electromagnetic analysis (EMA): measures and counter-measures for smart cards. In *E-smart*, pages 200–210, 2001.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *ACM STOC*, pages 84–93, 2005.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), 2009.

[RS91] Charles Rackoff and Daniel R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *CRYPTO*, pages 433–444, 1991.

[Sah99] Amit Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *IEEE FOCS*, pages 543–553, 1999.

[SPY$^+$10] François-Xavier Standaert, Olivier Pereira, Yu Yu, Jean-Jacques Quisquater, Moti Yung, and Elisabeth Oswald. Leakage resilient cryptography in practice. In *Towards Hardware-Intrinsic Security - Foundations and Practice*, pages 99–134. Springer, 2010.

[Ste93] Jacques Stern. A new identification scheme based on syndrome decoding. In *CRYPTO*, pages 13–21, 1993.

[Unr12] Dominique Unruh. Quantum proofs of knowledge. In *EUROCRYPT*, pages 135–152, 2012.

[Wee09] Hoeteck Wee. Zero knowledge in the random oracle model, revisited. In *ASIACRYPT*, pages 417–434, 2009.

[Wee16] Hoeteck Wee. KDM-security via homomorphic smooth projective hashing. In *PKC*, pages 159–179, 2016.

[Yao82] Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *IEEE FOCS*, pages 80–91, 1982.