Amplification of Chosen-Ciphertext Security

Huijia Lin[1] and Stefano Tessaro[2]

[1] MIT/Boston University
[2] MIT
{huijia,tessaro}@csail.mit.edu

Abstract. Understanding the minimal assumptions from which we can build a public-key encryption scheme secure against chosen-ciphertext attacks (a CCA-secure scheme, for short) is a central question in both practical and theoretical cryptography. Following the large body of work on hardness and correctness amplification, we ask the question of how far we can *weaken* a CCA-secure encryption scheme so that an efficient construction of a fully CCA-secure scheme from it can still be given.

We consider a *weak* CCA-secure encryption scheme that has decryption error $(1-\alpha)/2$ and is only weakly CCA secure in the sense that an adversary can distinguish encryptions of different messages with possibly large advantage $\beta < 1 - 1/\mathsf{poly}$. We show that whenever $\alpha^2 > \beta$, the weak correctness and the weak CCA security properties can be simultaneously amplified to obtain a fully CCA-secure encryption scheme with negligible decryption error. Our approach relies both on a new hardcore lemma for the setting of CCA security, and on an extension of a recently proposed approach to obtain CCA security by Hohenberger, Lewko, and Waters (EUROCRYPT '12) to handle large decryption errors.

Previously, such an amplification result was only known in the simpler case of security against chosen-plaintext attacks, as shown by Dwork, Naor, and Reingold (EUROCRYPT '04) and by Holenstein and Renner (CRYPTO '05).

1 Introduction

1.1 Public-key encryption and CCA security

The seminal work of Goldwasser and Micali [16] introduced the notion of *semantic security* as the basic security requirement for public-key encryption: It requires that no polynomial-time adversary can distinguish encryptions of any two messages m_0 and m_1 of its choice, except with negligible advantage, given only the public key. This is usually referred to as *security against a chosen plaintext attack*, or CPA security, for short. However, it turns out that many applications require a stronger notion of security known as *(adaptive) chosen-ciphertext security* (CCA security, for short) [44], where the above indistinguishability requirement must hold true even for adversaries with the additional ability to query a decryption oracle; for this reason, CCA security is considered to be the golden standard for secure public-key encryption.

In contrast to the case of CPA security, where simple constructions from generic assumptions (such as trapdoor permutations (TDP)) can be given, delivering CCA-secure schemes from general assumptions proved itself to be a much more challenging problem. In particular, building a CCA-secure scheme from a CPA-secure one remains a major longstanding open problem. Constructions additionally relying on non-interactive zero-knowledge proof systems (NIZKs) are known [40,12,44,46]. But, so far, all constructions of NIZKs require the existence (enhanced) TDPs, which are not known to be implied by CPA-secure encryption; furthermore, known constructions based on NIZKs are all non-black-box. It is in fact likely that no black-box construction of a CCA-secure scheme from a CPA-secure one exists, as confirmed at least for a certain natural class of constructions [14]. For this reason, efficient constructions have been instead given from more concrete families of assumptions, such as hash proof systems and variants thereof [9,48], lossy TDFs [43], correlated-product secure TDFs [45], adaptive TDFs [33], or using random oracles [2,3].

1.2 Our results: From weak to strong CCA security

In this paper, we ask and answer the following question:

> *"How far can we weaken CCA security and still provide a black-box construction of a CCA-secure encryption scheme from a scheme only satisfying the weaker notion?"*

Our approach builds upon the large body of works on *security amplification*, which considered a wide range of cryptographic primitives such as one-way functions and permutations [50,15,10,18], pseudorandom functions and permutations [38,36,11,37,47], collision-resistant hash functions [5], cryptographic puzzles and CAPTCHAs [4,29,31], watermarking schemes [27], two-party protocols like commitment and oblivious transfer [49,19,7,26], as well as interactive arguments [1,42,17,21,6]. Interestingly, limited work has been devoted to amplification of *public-key encryption*. The problem was first considered by Dwork, Naor, and Reingold [13] for CPA-secure public-key encryption. Constructions achieving better parameters were later proposed by Holenstein [24] and by Holenstein and Renner [25]. However, the question of amplifying *CCA* security has remained wide open ever since. This is the question that we tackle and solve in this work.[3]

Modeling weak CCA encryption. Our model of weak CCA encryption extends naturally the model of weak CPA encryption considered in [13,25]. We start from a bit-encryption[4] scheme with key generation algorithm Gen, encryption algorithm Enc, and decryption algorithm Dec, and weaken it in two different directions, allowing both for *non-negligible decryption errors* as well as for *non-negligible adversarial advantage* in a chosen-ciphertext attack. More concretely, for two given parameters $0 < \alpha, \beta \leq 1$, where $\alpha \geq 1/p(\kappa)$ and $\beta < 1 - 1/q(\kappa)$ for some polynomials p and q, we assume the following two conditions:

(i) **α-weak decryptability:** The decryption error over a random key-pair and a random bit is at most $\frac{1-\alpha}{2}$, i.e. $\mathsf{Pr}\left[(\mathsf{pk},\mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}, b \xleftarrow{\$} \{0,1\} : \mathsf{Dec}(\mathsf{sk},\mathsf{Enc}(\mathsf{pk},b)) = b\right] \geq \frac{1+\alpha}{2}$. We stress that this is a very weak guarantee, as it is taken over *random* choices of the keys and of the bit b, as well as of the coins used to encrypt b.

(ii) **β-weak security:** We consider the usual CCA-security game where an adversary obtains first the public key, and later a challenge ciphertext encrypting a random bit b. Moreover, the adversary can ask arbitrary decryption queries, with the sole exception that after the adversary obtains the challenge ciphertext, it cannot ask for its decryption. The task of the adversary is to output a guess b', and we are going to require that $\mathsf{Pr}[b' = b] \leq \frac{1+\beta}{2}$ for all polynomial-size adversaries.

Justifying weak CCA security. There are several reasons why assuming the existence of such a weak scheme is reasonable. Let us mention some natural examples.

- Within the general agenda of achieving CCA security from general assumptions, we may envision that a construction of a weak CCA scheme is potentially much easier to find than a construction of a full-fledged CCA-secure encryption scheme.
- An existing scheme designed to be CCA-secure may end up being less secure than expected due to the discovery of a better concrete attack or due to implementation errors, as in the recently discussed case of faulty key generation for RSA-based systems [34,22].
- It may be generally easier to build a CCA-secure scheme with large decryption errors. For example, as pointed out in [32], an encryption scheme with a simple, easily learnable, decryption algorithm must have large decryption error. In contrast to CPA encryption, reducing the decryption error turns out to be a major challenge in the case of CCA encryption, *even* if the scheme is already fully CCA secure.

[3] Note that in the *secret*-key setting, amplification of CCA security is, at least in principle, known to be feasible, as any weak form of CCA security implies weak one-way functions, and these are sufficient to build CCA-secure symmetric-key encryption via standard techniques.

[4] As every meaningful encryption scheme has at least the ability to encrypt a binary value, this is the weakest possible assumption in terms of message space of the basic scheme.

OUR MAIN RESULT. The question we are going to ask is whether for a certain α, β, there exists a transformation which delivers a CCA-secure encryption scheme from any scheme which has α-weak decryptability and β-weak security. We provide an affirmative answer to this question.

Theorem 1 (Main theorem, informal). *If $\alpha^2 > \beta$, there exists a black-box construction transforming any scheme with α-weak decryptability and β-weak security into a CCA-secure encryption scheme with negligible decryption error.*

We cannot rule out that constructions achieving a wider range of parameters α and β exist. In fact, we remark that the problem of determining the optimal parameters is open even in the simpler case of amplifying weak CPA security. While the constraint $\alpha^2 > \beta$ is shown [25] to be necessary for a restricted class of CPA black-box amplifiers, we see little value in extending this result to CCA security, as our amplifier itself is not within this class.

1.3 Our techniques

We now turn to a high-level overview of our techniques to amplify weak CCA encryption. Our approach relies on different existing techniques, which we are going to extend, such as those for simultaneously amplifying weak correctness and weak CPA security [25], and those for extending the message space of CCA-secure encryption schemes [39,23]. We start by reviewing these works, before turning to a description of our two main new techniques, namely hardcore lemmas for CCA-security and heavy-ciphertext pre-sampling, and how they are used.

AMPLIFICATION OF CPA ENCRYPTION. Given a public-key bit-encryption scheme PKE with α-weak decryptability and β-weak security with respect to chosen-plaintext attacks, the Holenstein-Renner (HR) construction [25] produces a fully CPA-secure encryption scheme with negligible decryption error. To encrypt each message m, the HR construction invokes the basic bit-encryption scheme PKE to encrypt several fresh random bits $b_1, \cdots, b_n$ under n public keys $\mathsf{pk}_1, \cdots, \mathsf{pk}_n$, producing ciphertexts $c_1, \cdots, c_n$; the bits $b_1, \cdots, b_n$ are then carefully "combined" to generate a one-time-pad k for hiding the actual message m, as well as some additional ciphertext component c'; the additional component c' is used by the legitimate receiver, given the secret keys, to reconstruct the one-time pad, but it should not leak any information about k to the adversary. The final ciphertext is $c = (c_1, \ldots, c_n, c', m \oplus k)$.

The reason why such a combiner can exist is that the probability that the legitimate receiver, given the secret keys, can learn each individual bit b_i from c_i is $(1+\alpha)/2$, which we expect to be sufficiently larger than the probability that the adversary learns b_i from c_i *without* the secret keys. To make this intuition sound, one uses Impagliazzo's hardcore lemma [28] and its tighter version by Holenstein [24]: It implies that if PKE is β-weakly CPA secure, then, for each i, with probability $1-\beta$ (over the choice of b_i, the randomness for sampling pk_i and encrypting b_i), the encryption of b_i is a "hard instance", meaning that given its encryption c_i, the bit b_i is (computationally) indistinguishable from a random independent bit. This gap between what an honest decryptor and an eavesdropper can recover can be leveraged by an information-theoretically secure one-way key-agreement protocol as in the setting of Maurer [35], which turns out to provide directly the right type of combiner.

FROM BIT CCA ENCRYPTION TO STRING CCA ENCRYPTION. It is well known that a CPA-secure string encryption scheme can be built from a CPA-secure bit-encryption scheme via simple parallel encryption of each bit. However, this approach does not lift to extending the message space of CCA-secure bit encryption, as an adversary can easily maul a challenge ciphertext $c_1 \cdots c_i \cdots c_n$ of a n-bit string $b_1 \cdots b_i \cdots b_n$ into another ciphertext $c_1 \cdots c_i' \cdots c_n$ of a related string $b_1 \cdots 0 \cdots b_n$, and thus win in the CCA security game—additional structure is needed to retain CCA security. Myers and shelat [39] showed that although this approach is not CCA secure, it satisfies a weaker adaptive security property—called UCCA security—which requires indistinguishability to hold for adversaries that can query a decryption oracle on any ciphertext $c_1, \cdots, c_n$ of their choice, except those that "quote" the challenge ciphertext, denoted as $c_1^*, \cdots, c_n^*$, at any of its components, that is $c_i = c_i^*$ for some i. Then, Myers and shelat, and later Hohenberger, Lewko, and Waters (HLW) [23], showed how

to construct a string CCA-secure scheme $\overline{\mathsf{PKE}}$ from such a UCCA-secure string encryption scheme PKE_s.[5] Here we briefly review the HLW construction. It uses PKE_s as an *inner* encryption scheme $\mathsf{PKE}_{\mathsf{in}} = \mathsf{PKE}_s$ and two *outer* schemes $\mathsf{PKE}_{\mathsf{out},1}$, $\mathsf{PKE}_{\mathsf{out},2}$ that are CCA-1 and CPA secure respectively. To encrypt a message m, the encryption algorithm proceeds by encrypting m together with two random strings $r_{\mathsf{out},1}$ and $r_{\mathsf{out},2}$ into an *inner ciphertext* $c_{\mathsf{in}} = \mathsf{Enc}_{\mathsf{in}}(\mathsf{pk}_{\mathsf{in}}, (m, r_{\mathsf{out},1}, r_{\mathsf{out},2}))$; it then encrypts the inner ciphertext into two outer ciphertexts $(c_{\mathsf{out},1}, c_{\mathsf{out},2})$ using $r_{\mathsf{out},1}$ and $r_{\mathsf{out},2}$ respectively as the randomness for encryption, that is, $c_{\mathsf{out},i} = \mathsf{Enc}_{\mathsf{out},i}(\mathsf{pk}_{\mathsf{out},i}, c_{\mathsf{in}}; r_{\mathsf{out},i})$ for $i = 1, 2$; the final ciphertext is simply $(c_{\mathsf{out},1}, c_{\mathsf{out},2})$. At a high level, the two outer schemes prevent the adversary from issuing a decryption query for a ciphertext whose embedded inner ciphertext "quotes" that in the challenge ciphertext, hence reducing CCA to UCCA security.

Our Approach. A seemingly plausible attempt for constructing a CCA-secure encryption scheme from a weak scheme PKE with α-decryptability and β-weak CCA-security is to first try to show that the HR construction PKE', when instantiated with PKE as the basic bit-encryption scheme, is UCCA secure, and subsequently plugging PKE' as the inner encryption scheme into the HLW construction $\overline{\mathsf{PKE}}$, and show that it yields a CCA-secure encryption scheme.

Unfortunately, we encounter the following two challenges: First, it is unclear whether the weak CCA security of PKE is amplified through the construction of PKE' to UCCA security; in particular, known hardcore lemmas [28,24] only hold for games where the challenger is stateless, but the challenger in the CCA security game is stateful (it changes its behavior before and after the challenge ciphertext is generated). Second, it turns out that the security proof of the HLW construction requires the basic scheme PKE to have "unpredictability"—that is, a random cipheretxt (of a random bit) of PKE has high entropy and is almost impossible to blindly guess—which holds trivially for any fully-secure CPA encryption scheme with negligible decryption error, but is not satisfied by a weak CCA encryption scheme.

Overcoming these two difficulties turns out to be quite challenging and requires the adoption of new techniques, for which we now provide a high-level overview.

Step 1: The Hardcore Lemma for CCA security and XCCA security. To overcome the first difficulty, we prove a variant of Impagliazzo's hardcore lemma which applies to CCA security (Theorem 2 below): It implies that if a scheme is weakly β-CCA-secure, then with probability $1 - \beta$ (over the randomness for choosing a random plaintext bit, for key generation, and for encryption), given an encryption of a random bit b, b is indistinguishable from a random *independent* bit even to adversaries with access to the decryption oracle. Our new hardcore lemma can be used to prove that PKE' satisfies an even stronger adaptive security property than UCCA, called XCCA (read as "cross"-CCA), which guarantees indistinguishability even for adversaries with access to decryption oracles that decrypts ciphertext of the basic scheme PKE under each individual component key of PKE', subject to the restriction that the decryption oracle for the i-th component does not answer queries that "quote" the corresponding component in the challenge ciphertext. As we will see shortly, this stronger security guarantee is quintessential for overcoming the second difficulty.

Finally, rather than presenting a direct proof of the hardcore lemma for CCA security, we provide a general characterization of games for which hardcore lemmas exist, which extends beyond games for which such lemmas are known [28,24,47]. Our hardcore lemma for CCA-security is then simply derived as a special case. We believe this step to be of independent interest.

Step 2: From XCCA security to CCA security. We prove that the CCA security of $\overline{\mathsf{PKE}}$ can be based on the stronger XCCA security of the inner encryption PKE', even if the underlying basic scheme PKE is not sufficiently "unpredictable" – in contrast to the proof in [23]. This requires a substantially different analysis than the one of [23], and in particular a new reduction. Concretely, we overcome lack of unpredictability by introducing a new technique called *heavy-ciphertext pre-sampling*. Roughly speaking, this technique allows the security reduction (from CCA security of $\overline{\mathsf{PKE}}$ to XCCA security

[5] In fact, [23] showed a more general construction of string CCA encryption schemes from any encryption scheme that is DCCA secure and unpredictable. In particular, UCCA security is a special case of DCCA security.

of PKE') to proactively predict and decrypt all highly likely ciphertexts of PKE, and the challenging task is to prove that these are the only components of the inner challenge ciphertext an adversary may indeed easily "quote" after seeing the challenge ciphertext.

2 Preliminaries

We start by introducing some basic notations, and then move to reviewing definitions for public-key encryption schemes, their correctness and their security. We review some basic constructions and techniques to deal with encryption schemes in Appendix A.

2.1 Basic concepts and notation

Throughout this paper, a function $f : \mathbb{N} \to [0, 1]$ is *negligible* if it vanishes faster than the inverse of any polynomial p, i.e., for all $c > 0$ there exists κ_c such that $f(\kappa) \leq \kappa^{-c}$ for all $\kappa \geq \kappa_c$. The probability distribution of a random variable X is usually denoted as P_X, and we occasionally use the shorthand $\mathsf{P}_X(x)$ for $\mathsf{Pr}[X = x]$. Adversaries are going to be modeled as non-uniform families of (randomized) circuits for ease of exposition, but all results extend with some work to the uniform setting, as we occasionally point out.

2.2 Weak and Strong CCA Secure Encryption

A *public-key encryption scheme* with message space $\mathcal{M} \subseteq \{0,1\}^*$ is a triple $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, where (i) Gen is the (randomized) *key generation algorithm*, outputting a pair $(\mathsf{pk}, \mathsf{sk})$ consisting of a *public-* and a *secret-key*, respectively (ii) Enc is the (randomized) *encryption algorithm* outputting a ciphertext $c = \mathsf{Enc}(\mathsf{pk}, m)$ for any message $m \in \mathcal{M}$ and a valid public key pk; and (iii) Dec is the deterministic *decryption algorithm* such that $\mathsf{Dec}(\mathsf{sk}, c) \in \mathcal{M} \cup \{\perp\}$. All algorithms additionally take (implicitly) as input the security parameter 1^κ in unary form, and the message space $\mathcal{M}$ may also depend on the security parameter κ. Whenever $\mathcal{M} = \{0,1\}$, we say that the scheme is a *bit-encryption* scheme. We sometimes need to make the randomness used by Gen and Enc explicit: In these cases, we write $\mathsf{Gen}(r)$ and $\mathsf{Enc}(\mathsf{pk}, m; r)$ to highlight the fact that random coins r are used to generate keys by Gen and to encrypt the message m, respectively.

Correctness of PKE. Throughout this paper, we say that the encryption scheme PKE with message space $\mathcal{M}$ has *decryption error* δ if $\mathsf{Pr}\left[(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen},\ m \xleftarrow{\$} \mathcal{M} :\ \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)) \neq m\right] \leq \delta$, where the probability is additionally over the random coins of Enc. Moreover, we say that a scheme is *almost perfectly correct*, if for an overwhelming fraction of randomness r used by the key generation algorithm, for $(\mathsf{pk}, \mathsf{sk}) = \mathsf{Gen}(r)$, and all messages $m \in \mathcal{M}$, we have $\mathsf{Pr}[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)) = m] = 1$.

Security of PKE. In general, security of the scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is defined via the following security game involving a *challenger* and an adversary $\mathcal{A}$:

Game $\mathsf{CCA2}^{\mathcal{A}}_{\mathsf{PKE}}$:

- **(i)** The challenger generates $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}$ and $b \xleftarrow{\$} \{0,1\}$, and gives pk to $\mathcal{A}$.
- **(ii)** The adversary $\mathcal{A}$ asks decryption queries c, which are answered with the message $\mathsf{Dec}(\mathsf{sk}, c) \in \mathcal{M} \cup \{\perp\}$.
- **(iii)** The adversary $\mathcal{A}$ inputs (m_0, m_1) with $|m_0| = |m_1|$ to the challenger, and receives a challenge ciphertext $c^* \xleftarrow{\$} \mathsf{Enc}(\mathsf{pk}, m_b)$.
- **(iv)** The adversary $\mathcal{A}$ asks further decryption queries $c \neq c^*$, which are answered with the message $\mathsf{Dec}(\mathsf{sk}, c) \in \mathcal{M} \cup \{\perp\}$.
- **(v)** The adversary $\mathcal{A}$ outputs a bit b', and *wins* the game if $b' = b$.

We refer to decryption queries in phase **(ii)** and **(iv)** as *before-the-fact* and *after-the-fact* decryption queries, respectively. Moreover, in the case that PKE is a bit-encryption scheme we assume without loss of generality that $(m_0, m_1) = (0, 1)$, and hence $\mathsf{Enc}(\mathsf{pk}, b)$ is the challenge ciphertext. We also define the $\mathsf{CCA2}$-*advantage* of the adversary $\mathcal{A}$ as $\mathbf{Adv}^{\mathsf{CCA2}}_{\mathsf{PKE}}(\mathcal{A}) = 2 \cdot \mathsf{Pr}[b' = b] - 1$. We say that an

encryption scheme is *CCA-secure* if $\mathbf{Adv}^{\mathsf{CCA2}}_{\mathsf{PKE}}(\mathcal{A})$ is negligible for all polynomial-size adversaries $\mathcal{A}$. We say it is *q-CCA-secure* if this holds for adversaries making at most q decryption queries, whereas it is *CPA-secure* if it is 0-CCA-secure. The following notation will also be convenient.

Definition 1. *For $\alpha, \beta \in [0,1]$, a bit-encryption scheme* PKE *is* (α,β)-CCA-secure *if the following two properties hold:* **(i)** PKE *has decryption error* $(1-\alpha)/2$*, and* **(ii)** *For any polynomial-size adversary $\mathcal{A}$, we have* $\mathbf{Adv}^{\mathsf{CCA2}}_{\mathsf{PKE}}(\mathcal{A}) \leq \beta$.

In passing, we point out that CPA-secure encryption with negligible decryption error implies one-way functions [30], and in turn implies pseudorandom generators [20], all in a black-box way.

3 Hardness Amplification and the Hardcore Lemma for CCA Security

Informally speaking, Impagliazzo's Hardcore Lemma [28] asserts that if for a predicate P it is mildly hard to compute $P(x)$ on a random input x given side information $f(x)$ (i.e., say this can be done with probability at most $\frac{1+\varepsilon}{2}$), then there must exist a sufficiently large subset $\mathcal{S}$ (the "hardcore set") of the inputs such that when sampling x' from $\mathcal{S}$, it is infeasible to predict $P(x')$ from $f(x')$ noticeably better than by random guessing. A tight proof where the set $\mathcal{S}$ contains a $(1-\varepsilon)$-fraction of the inputs was given by Holenstein [24]. The main contribution of this section is deriving a similar statement for (weak) CCA-secure encryption to be used in the analysis of our construction of a CCA-secure encryption scheme later.

Recently, Tessaro [47] gave a hardcore lemma for interactive primitives, which is however not sufficient to capture CCA security, as it only considers challengers with state independent of the interaction. Here, in contrast, we present a new abstraction of existing proofs of hardcore lemmas, which is of independent interest. Not only we apply it to derive the hardcore lemma for CCA security of bit-encryption, but it also yields previous more restricted statements [28,47] as special cases.

Bit-guessing games. Let us took a more abstract look at games (such as the CCA-security game) where the adversary is asked to guess a bit. Formally, we describe a *bit-guessing game* as a tuple $G = (\mathsf{P}_X, \mathcal{C}, P)$, where P_X is a probability distribution with support $\mathcal{X}$, $\mathcal{C}$ is an interactive *stateful* machine taking an auxiliary input $x \in \mathcal{X}$, and $P : \mathcal{X} \to \{0,1\}$ is a predicate. Combined with an adversary $\mathcal{A}$, G defines the following game: First, an input $x \xleftarrow{\$} \mathsf{P}_X$ is sampled. The game then continues with the interaction between the challenger $\mathcal{C}(x)$ and an adversary $\mathcal{A}$, which then outputs a bit $b' \xleftarrow{\$} \mathcal{A}^{\mathcal{C}(x)}$ (here the oracle $\mathcal{C}(x)$ keeps state). The goal of the adversary is to guess the bit $P(x)$. In particular, we define the *G-advantage of $\mathcal{A}$ relative to a distribution* P as

$$\mathbf{Adv}^{G}_{\mathsf{P}}(\mathcal{A}) = 2 \cdot \mathsf{Pr}\left[x \xleftarrow{\$} \mathsf{P},\ b' \xleftarrow{\$} \mathcal{A}^{\mathcal{C}(x)} :\ b = P(x)\right] - 1\ . \tag{1}$$

We say that G is (s,ε)-*hard* if $\mathbf{Adv}^{G}_{\mathsf{P}_X}(\mathcal{A}) \leq \varepsilon$ for all s-size adversaries $\mathcal{A}$. Definitions extend naturally to the asymptotic setting.

Hardcore lemmas and measures. A *measure* $\mathcal{M}$ for a bit-guessing game G is a mapping $\mathcal{M} : \mathcal{X} \to [0,1]$, and its *density* is $\mu(\mathcal{M}) = \sum_{x\in\mathcal{X}} \mathsf{P}_X(x) \cdot \mathcal{M}(x)$. We associate with $\mathcal{M}$ the probability distribution $\mathsf{P}_{\mathcal{M}}$ such that $\mathsf{P}_{\mathcal{M}}(x) := \mathsf{P}_X(x) \cdot \mathcal{M}(x)/\mu(\mathcal{M})$ for all $x \in \mathcal{X}$. The role of a measure is that of adjoining an event $\mathcal{E}$ to the sampling of $x \xleftarrow{\$} \mathsf{P}_X$ such that $\mathsf{Pr}\left[\mathcal{E} \,\middle|\, X = x\right] = \mathcal{M}(x)$; then in particular $\mathsf{Pr}\left[\mathcal{E}\right] = \mu(\mathcal{M})$, and $\mathsf{Pr}\left[X = x \,\middle|\, \mathcal{E}\right] = \mathsf{P}_{\mathcal{M}}(x)$.

We ask the question of which bit-guessing games admit a *hardcore measure*: Assuming the game G is ε-hard for some $\varepsilon \in [0,1]$, we seek for a measure $\mathcal{M}$ with large density (e.g. $\mu(\mathcal{M}) \geq 1-\varepsilon$) such that conditioned on the associated event $\mathcal{E}$, the game G is very hard to win. In [47], a proof that this is true for the case where $\mathcal{C}(x)$ is stateless for each x was given. Our new approach extends this to possibly stateful challengers, as in the case of CCA security.

Abstract hardcore lemmas. We give a simple sufficient condition on a bit-guessing game $G = (\mathsf{P}_X, \mathcal{C}, P)$ to admit a hardcore lemma. The condition is formulated in terms of the ability, for any given x, to estimate the probability that a binary-output adversary for G, sampled according to a

given distribution over circuits, outputs one when run on $\mathcal{C}(x)$. In particular, we call an oracle O a *size s circuit sampler* for G if, upon each invocation, it returns the description of a valid adversary $\mathcal{A}$ for G of size s. For each such O, we define $p_1^{\mathsf{O},G}(x)$ as the probability that a randomly sampled adversary $\mathcal{A} \xleftarrow{\$} \mathsf{O}$ outputs one when run with $\mathcal{C}(x)$, i.e.,

$$p_1^{G,\mathsf{O}}(x) := \mathsf{Pr}\left[\mathcal{B} \xleftarrow{\$} \mathsf{O}, b' \xleftarrow{\$} \mathcal{B}^{\mathcal{C}(x)} : b' = 1\right] . \tag{2}$$

The following definition captures the notion of a good estimation algorithm for $p_1^{G,\mathsf{O}}(x)$ which can only interact with $\mathcal{C}(x)$ and obtain samples from O, bur does not learn x and must be equally successful on all such x.

Definition 2 (p_1-estimator). *Let $G = (\mathsf{P}_X, \mathcal{C}, P)$ be a bit-guessing game. Then, a (s, s', q, γ, η)-p_1-estimator for G is a size s circuit $\mathcal{E}$ outputting a real number in $[0,1]$, such that for all size-s' circuit samplers O, for all x, and with $p_1(x) = p_1^{G,\mathsf{O}}(x)$,*

$$\mathsf{Pr}\left[\mathcal{B}_1, \ldots, \mathcal{B}_q \xleftarrow{\$} \mathsf{O},\ \overline{p_1} \xleftarrow{\$} \mathcal{E}^{\mathcal{C}(x)}(\mathcal{B}_1, \ldots, \mathcal{B}_q) :\ |\overline{p_1} - p_1(x)| > \gamma\right] < \eta\,. \tag{3}$$

Note that in particular $q \cdot s' \leq s$. The following theorem relates the existence of a hardcore lemma for a certain game G with the existence of a p_1-sampler for G. Its proof abstracts the ones of [24,47] for special cases, and is deferred to Appendix B.

Proposition 1 (The Abstract Hardcore Lemma). *Let $s \in \mathbb{N}$ and $\varepsilon \in [0,1]$. Let $G = (\mathsf{P}_X, \mathcal{C}, P)$ be a bit-guessing game which is (s,ε)-hard. Then, for all $\gamma > 0$, if for some $s' = s'(\gamma)$ there exists an $(s, s', q, \gamma(1-\varepsilon)/4, \gamma(1-\varepsilon)/4)$-$p_1$-estimator for G, then there exists a measure $\mathcal{M} = \mathcal{M}_\gamma$ such that:*

(i) $\mu(\mathcal{M}) \geq 1 - \varepsilon$, **(ii)** *For all s'-size adversaries $\mathcal{B}$,* $\mathbf{Adv}^{G}_{\mathsf{P}_\mathcal{M}}(\mathcal{B}) \leq \gamma$.

The hardcore lemma for CCA-security. We are now going to show a hardcore lemma for CCA-security as an application of Proposition 1. Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key *bit* encryption scheme such that Gen and Enc take randomness of lengths ρ_{Gen} and ρ_{Enc}, respectively. Formally, we consider the bit-guessing game $\mathsf{CCA2}[\mathsf{PKE}] = (\mathsf{P}_X, \mathcal{CCA}, P)$ where P_X is the uniform distribution on $\{0,1\}^{\rho_{\mathsf{Gen}}} \times \{0,1\}^{\rho_{\mathsf{Enc}}} \times \{0,1\}$, whereas $\mathcal{CCA}(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}, b)$ is the challenger for the CCA-security game for PKE with challenge bit b, public key and secret key $(\mathsf{pk}, \mathsf{sk}) = \mathsf{Gen}(r_{\mathsf{Gen}})$, and challenge ciphertext $c^* = \mathsf{Enc}(\mathsf{pk}, b; r_{\mathsf{Enc}})$. Moreover, we define $P(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}, b) = b$. The following lemma gives an appropriate p_1-estimator for $\mathsf{CCA2}[\mathsf{PKE}]$.

Lemma 1. *For all $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with message space $\{0,1\}$, and all $s' \in \mathbb{N}$, $\gamma, \eta \in (0,1]$, there exists a (s, s', q, γ, η)-p_1-estimator for $\mathsf{CCA2}[\mathsf{PKE}]$ with $q = O(\log(1/\eta)/\gamma^2)$ and $s = s' \cdot q + O(1)$.*

Proof. The estimator $\mathcal{E}$, given pk from $\mathcal{CCA}$, runs sequentially each of $\mathcal{B}_1, \ldots, \mathcal{B}_q$ on input pk until they output their query $(0,1)$. All before-the-fact decryption queries are answered using the challenger $\mathcal{CCA}$. It then obtains a challenge ciphertext c^*, and then resumes the execution of $\mathcal{B}_i$'s from the last state before outputting $(0,1)$, again using the challenger to reply to decryption queries. Finally, let b'_i be the output of $\mathcal{B}_i$; the estimator $\mathcal{E}$ outputs the average $z = (1/q) \cdot \sum_{i=1}^{q} b'_i$. The error is at most γ with probability at most η by the Chernoff bound. □

We stress that the above lemma is only true for bit-encryption. Should we consider a larger set of messages, each $\mathcal{B}_i$ could ask a different message pair, and the above estimation technique would fail.

The following theorem is a simple combination of Proposition 1 and Lemma 1.

Theorem 2 (Hardcore Lemma for CCA Security). *Let $\alpha, \beta \in [0,1]$, and let $s \in \mathbb{N}$. Moreover, let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption scheme with message space $\{0,1\}$, and assume that $\mathbf{Adv}^{\mathsf{CCA2}}_{\mathsf{PKE}}(\mathcal{A}) \leq \beta$ for all s-size adversaries $\mathcal{A}$.*

Then, for all $\gamma > 0$, there exists a measure $\mathcal{M}$ such that $\mu(\mathcal{M}) \geq 1 - \beta$, and $\mathbf{Adv}^{\mathsf{CCA2}[\mathsf{PKE}]}_{\mathsf{P}_\mathcal{M}}(\mathcal{B}) \leq \gamma$ for all adversaries $\mathcal{B}$ with size s', where $s = O(s' \cdot \log(1/\gamma(1-\varepsilon))/\gamma^2(1-\varepsilon)^2)$.

SOME REMARKS. We provide some important remarks on extensions of the above results.

Remark 1. We remark that the above results are for the non-uniform setting. This makes the presentation of the main ideas somewhat simpler, but we note that the abstract hardcore lemma above extends to uniform security following the approach of [24], provided one can efficiently simulate the interaction between an adversary and the given challenger.

Remark 2. Games where $\mathcal{C}(x)$ is stateless for each x easily yield a good sampler via sequential repetition, and therefore the results of [28,24,47] all easily follow from the abstract hardcore lemma.

Remark 3. Without giving further detail, we briefly point out that [47] provides a stronger result for the case where $\mathcal{C}(x)$ is possibly not efficient, as e.g. $|x|$ is exponentially large, yet the interaction between an adversary and $\mathcal{C}(x)$ for a *random* x is efficiently simulatable; this won't be necessary here, but similar techniques can be applied.

4 From Weak to Strong CCA Security: The Construction and its Security

In this section, we present our construction to transform an (α, β)-CCA encryption scheme into a fully CCA-secure encryption scheme. We start by reviewing some information-theoretic tools underlying our construction, before turning to its description and security.

4.1 Information-theoretically secure key-agreement

We consider the problem of two parties, Alice and Bob, agreeing on a secret key with *unconditional security* in a setting where they each hold values $X_1, \ldots X_n$ and $Y_1 \ldots, Y_n$, respectively, in presence of an adversary obtaining values $Z_1, \ldots, Z_n$; in particular, (X_i, Y_i, Z_i) are sampled independently from a given tripartite probability distributions P_{XYZ} for all $i = 1, \ldots, n$. That is, (X_i, Y_i, Z_i) are possibly correlated for each i, but independent across distinct indices $i \neq j$. Moreover, Alice and Bob are connected via an authenticated channel, allowing them to exchange messages, which is however wiretapped by the adversary. The problem of secret-key agreement in this setting was first considered by Maurer [35]. Here, we consider the special case where the channel only allows *one-way* communication from Alice to Bob. The following definition formally captures such a protocol.

Definition 3 (One-way key-agreement). *Let $\varepsilon, \delta : \mathbb{N} \to [0,1]$, and let $n, \ell : \mathbb{N} \to \mathbb{N}$ be monotonically increasing functions. Moreover, let $\mathcal{P} = \{\mathcal{P}_\kappa\}_{\kappa \in \mathbb{N}}$ be a family of sets of tripartite probability distribution P_{XYZ}. An $(\mathcal{P}, \varepsilon, \delta, n, \ell)$-*one-way key agreement protocol *is a probabilistic polynomial-time protocol $\mathsf{OKA} = (\mathsf{KAEnc}, \mathsf{KADec})$ such that for all $\kappa \in \mathbb{N}$, all $\mathsf{P}_{XYZ} \in \mathcal{P}_\kappa$, and for independent samples $(X_1, Y_1, Z_1), \ldots, (X_n, Y_n, Z_n) \xleftarrow{\$} \mathsf{P}_{XYZ}$ (where $n = n(\kappa)$), the following two properties hold:*

Correctness. *With probability $1 - \delta(\kappa)$, both parties output the same key, i.e.,*

$$\mathsf{Pr}\left[(C, K) \xleftarrow{\$} \mathsf{KAEnc}(1^\kappa, X_1, \ldots, X_n);\ K' \xleftarrow{\$} \mathsf{KADec}(1^\kappa, Y_1, \ldots, Y_n; C) : K = K'\right] \geq 1 - \delta(\kappa)\ .$$

Security. $\mathsf{SD}((C, K, Z_1, \ldots, Z_n); (C, K', Z_1, \ldots, Z_n)) \leq \varepsilon(\kappa)$, *where $(C, K) \xleftarrow{\$} \mathsf{KAEnc}(1^\kappa, X_1, \ldots, X_n)$, and $K' \xleftarrow{\$} \{0,1\}^{\ell(\kappa)}$, and SD denotes statistical distance.*

In the following for some $\alpha, \beta : \mathbb{N} \to [0,1]$, we consider a special set $\mathcal{D}(\alpha, \beta)$ of tripartite probability distributions introduced by Holenstein and Renner [25].

Definition 4 ([25]). *Let $\alpha, \beta : \mathbb{N} \to [0,1]$. We define $\mathcal{D}(\alpha, \beta) = \{\mathcal{D}_\kappa(\alpha, \beta)\}_{\kappa \in \mathbb{N}}$ such that for all $\kappa \in \mathbb{N}$, $\mathsf{P}_{XYZ} \in \mathcal{D}_\kappa(\alpha, \beta)$ if a triple $(X, Y, Z) \xleftarrow{\$} \mathsf{P}_{XYZ}$ satisfies the following:* **(i)** $\mathsf{Pr}[X = 0] = \mathsf{Pr}[X = 1] = \frac{1}{2}$, *i.e., X is a uniform bit,* **(ii)** $\mathsf{Pr}[X = Y] \geq \frac{1+\alpha(\kappa)}{2}$, *and* **(iii)** *there exists an event $\mathcal{E}$, defined on (X, Z), such that $\mathsf{Pr}\left[X = 0 \,\middle|\, Z = z, \mathcal{E}\right] = \mathsf{Pr}\left[X = 1 \,\middle|\, Z = z, \mathcal{E}\right] = \frac{1}{2}$ for all z, and $\mathsf{Pr}[\mathcal{E}] \geq 1 - \beta(\kappa)$.*

We now discuss feasibility of one-way KA protocols for $\mathcal{D}(\alpha, \beta)$. The following was proved by Holenstein and Renner [25] and will be useful below. We give an asymptotic reformulation.

Proposition 2. *Let $\alpha, \beta : \mathbb{N} \to [0,1]$ be such that $\alpha^2 > \beta + \Omega(1)$, and let $\ell : \mathbb{N} \to \mathbb{N}$ be a polynomial function. Then, there exists a polynomial-time $(\mathcal{D}(\alpha,\beta), \varepsilon, \delta, n, \ell)$-one-way KA protocol such that $n(\kappa) = \frac{1}{7} \cdot \ell(\kappa) \cdot (\alpha^2 - \beta - O(1))$ and moreover, $\varepsilon(\kappa)$ is negligible in $n(\kappa)$, and $\delta(\kappa) = 2^{-\Theta(n(\kappa))}$.*

There is no a-priori reason why α^2 and β could not be closer, yet no better gap can be proven given existing constructions of capacity-achieving error-correcting codes. In the error-free case, however, the following better result is proven in Appendix C.

Proposition 3. *Let p be a polynomial, $\varepsilon' : \mathbb{N} \to [0,1]$, and $\ell : \mathbb{N} \to \mathbb{N}$ be a polynomial function. Then, there exists a $\mathcal{D}(1, 1 - \frac{1}{p(\kappa)}, \varepsilon, \delta, n, \ell)$-one-way KA protocol where $n(\kappa) = 2/(1-\beta(\kappa)) \cdot (\ell(\kappa) + 2\log(1/\varepsilon'(\kappa)) + O(1))$, $\varepsilon(\kappa) \leq O(\sqrt{\varepsilon'(\kappa)})$, and $\delta(\kappa) = 0$.*

4.2 The main construction

Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a bit-encryption scheme which is (α,β)-secure. Assuming the existence of an information-theoretically secure one-way key agreement protocol for $\mathcal{D}(\alpha,\beta)$, we present a construction of a CCA-secure public-key encryption scheme $\overline{\mathsf{PKE}} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}})$, with message length $\ell = \ell(\kappa)$ and negligible decryption error, which makes black-box use of the basic scheme PKE.

At the highest level, our construction $\overline{\mathsf{PKE}}$ follows the paradigm recently proposed by Hohenberger, Lewko, and Waters [23]. In particular, it consists of an inner scheme $\mathsf{PKE}_{\mathsf{in}} = (\mathsf{Gen}_{\mathsf{in}}, \mathsf{Enc}_{\mathsf{in}}, \mathsf{Dec}_{\mathsf{in}})$ and two outer schemes $\mathsf{PKE}_{\mathsf{out},1} = (\mathsf{Gen}_{\mathsf{out},1}, \mathsf{Enc}_{\mathsf{out},1}, \mathsf{Dec}_{\mathsf{out},1})$ and $\mathsf{PKE}_{\mathsf{out},2} = (\mathsf{Gen}_{\mathsf{out},2}, \mathsf{Enc}_{\mathsf{out},2}, \mathsf{Dec}_{\mathsf{out},2})$, all three of which will be built from PKE, and specified below. For $\star \in \{\mathsf{in}, (\mathsf{out},1), (\mathsf{out},2)\}$, let us further denote by $\ell_\star, \rho_\star$ and $t_\star$ the message, randomness, and ciphertext lengths of $\mathsf{PKE}_\star$, respectively. We are going to require $\ell_{\mathsf{in}} = \ell + \rho_{\mathsf{out},1} + \rho_{\mathsf{out},2}$ as well as $\ell_{\mathsf{out},1} = \ell_{\mathsf{out},2} = t_{\mathsf{in}}$. A formal description of $\overline{\mathsf{PKE}}$ is given in Figure 1, on the left: We encrypt the message m, together with two random values $r_{\mathsf{out},1}$ and $r_{\mathsf{out},2}$, obtaining an *inner* ciphertext c_{in}, which is then encrypted twice with the two outer schemes, using $r_{\mathsf{out},1}$ and $r_{\mathsf{out},2}$ as the respective random coins. Decryption recovers the message by decrypting the ciphertext via $\mathsf{Dec}_{\mathsf{out},1}$ and $\mathsf{Dec}_{\mathsf{in}}$ using the corresponding secret keys, and then checks validity of the ciphertext by re-encrypting the inner ciphertext using the public keys and the recovered random coins.

We now turn to describing the construction of the component schemes $\mathsf{PKE}_{\mathsf{in}}$, $\mathsf{PKE}_{\mathsf{out},1}$ and $\mathsf{PKE}_{\mathsf{out},2}$ from the basic scheme PKE.

The inner scheme. Let $\mathsf{OKA} = (\mathsf{KAEnc}, \mathsf{KADec})$ be a $(\mathcal{D}(\alpha,\beta), \varepsilon, \delta, n, \ell_{\mathsf{in}})$-one-way key agreement protocol such that ε and δ are negligible, and known (recall that PKE is (α,β)-CCA secure). We define $\mathsf{PKE}_{\mathsf{in}} = (\mathsf{Gen}_{\mathsf{in}}, \mathsf{Enc}_{\mathsf{in}}, \mathsf{Dec}_{\mathsf{in}})$ as in Figure 1, on the right: It behaves as the construction from [25] to amplify security and correctness of a weak *CPA*-secure encryption scheme. (We will prove below that it actually achieves stronger security when using an (α,β)-CCA secure encryption scheme.) It encrypts random bits $b_1, \ldots, b_n$ with the basic scheme, and then generates a session key k via $\mathsf{KAEnc}(b_1, \ldots, b_n)$, and a ciphertext c', and uses the key k as an one-time pad. Decryption via KADec is then obvious. It is easy to see that the decryption error of this scheme is inherited from OKA, i.e., it is upper bounded by exactly δ.

The outer schemes. We now instantiate the two outer schemes. The following description is fairly high-level, but sufficient to fully specify the construction. We refer the reader unfamiliar with the basic components to Appendix A for a self-contained review.

We first derive a CPA-secure public-key encryption scheme $\mathsf{PKE}_{\mathsf{out}}^{\ell,\rho}$ with message length $\ell = \mathsf{poly}(\kappa)$ and randomness length $\rho = \omega(\log(\kappa))$ from the basic scheme PKE which also enjoys almost-perfect correctness:[6]

1. We use the same construction as in $\mathsf{PKE}_{\mathsf{in}}$ to achieve a CPA-secure scheme $\mathsf{PKE}'_{\mathsf{out}}$, with message length truncated to 1-bit. CPA-security of the resulting scheme follows from the proof in [25] or from the stronger Lemma 2 below. Let ρ be the randomness length of $\mathsf{PKE}'_{\mathsf{out}}$.

[6] In the following, we are not going to optimize the complexity of the scheme; it is clear that some modifications can be done to save on complexity.

Scheme $\overline{\mathsf{PKE}} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}})$:

Key generation $\overline{\mathsf{Gen}}(1^\kappa)$**:**
- $(\mathsf{pk}_{\mathsf{in}}, \mathsf{sk}_{\mathsf{in}}) \xleftarrow{\$} \mathsf{Gen}_{\mathsf{in}}(1^\kappa)$, $(\mathsf{pk}_{\mathsf{out},i}, \mathsf{sk}_{\mathsf{out},i}) \xleftarrow{\$} \mathsf{Gen}_{\mathsf{out},i}(1^\kappa)$ for $i = 1, 2$.
- Return $(\overline{\mathsf{pk}} = (\mathsf{pk}_{\mathsf{in}}, \mathsf{pk}_{\mathsf{out},1}, \mathsf{pk}_{\mathsf{out},2}), \overline{\mathsf{sk}} = (\mathsf{sk}_{\mathsf{in}}, \mathsf{sk}_{\mathsf{out},1}, \mathsf{pk}_{\mathsf{out},1}, \mathsf{pk}_{\mathsf{out},2}))$.

Encryption $\overline{\mathsf{Enc}}(\overline{\mathsf{pk}}, m)$**:** // $m \in \{0,1\}^\ell$
- $r_{\mathsf{out},1} \xleftarrow{\$} \{0,1\}^{\rho_{\mathsf{out},1}}$ and $r_{\mathsf{out},2} \xleftarrow{\$} \{0,1\}^{\rho_{\mathsf{out},2}}$
- $c_{\mathsf{in}} \xleftarrow{\$} \mathsf{Enc}_{\mathsf{in}}(\mathsf{pk}_{\mathsf{in}}, m \,\|\, r_{\mathsf{out},1} \,\|\, r_{\mathsf{out},2})$
- $c_{\mathsf{out},i} \leftarrow \mathsf{Enc}_{\mathsf{out},i}(\mathsf{pk}_{\mathsf{out},i}, c_{\mathsf{in}}; r_{\mathsf{out},i})$ for $i = 1, 2$.
- Output ciphertext $c_{\mathsf{out},1} \,\|\, c_{\mathsf{out}_2}$.

Decryption $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, c = c_{\mathsf{out},1} \,\|\, c_{\mathsf{out}_2})$**:**
- $c'_{\mathsf{in}} \leftarrow \mathsf{Dec}_{\mathsf{out},1}(\mathsf{sk}_{\mathsf{out},1}, c_{\mathsf{out},1})$
- $m' \,\|\, r'_{\mathsf{out},1} \,\|\, r'_{\mathsf{out},2} \leftarrow \mathsf{Dec}_{\mathsf{in}}(\mathsf{sk}_{\mathsf{in}}, c'_{\mathsf{in}})$
- If $\mathsf{Enc}_{\mathsf{out},i}(\mathsf{pk}_{\mathsf{out},i}, c'_{\mathsf{in}}; r'_{\mathsf{out},i}) = c_{\mathsf{out},i}$ for $i = 1, 2$ then return m, else return $\perp$.

Scheme $\mathsf{PKE}_{\mathsf{in}} = (\mathsf{Gen}_{\mathsf{in}}, \mathsf{Enc}_{\mathsf{in}}, \mathsf{Dec}_{\mathsf{in}})$:

Key generation $\mathsf{Gen}_{\mathsf{in}}(1^\kappa)$**:**
- $(\mathsf{pk}_1, \mathsf{sk}_1), \ldots, (\mathsf{pk}_n, \mathsf{sk}_n) \xleftarrow{\$} \mathsf{Gen}(1^\kappa)$
- Return $(\mathbf{pk} = (\mathsf{pk}_1, \ldots, \mathsf{pk}_n), \mathbf{sk} = (\mathsf{sk}_1, \ldots, \mathsf{sk}_n))$.

Encryption $\mathsf{Enc}_{\mathsf{in}}(\mathbf{pk}, m)$**:** // $m \in \{0,1\}^{\ell_{\mathsf{in}}}$
- $b_1, \ldots, b_n \xleftarrow{\$} \{0,1\}$
- $c_i \xleftarrow{\$} \mathsf{Enc}(\mathbf{pk}[i], b_i)$ for all $i = 1, \ldots, n$
- $(k, c') \xleftarrow{\$} \mathsf{KAEnc}(b_1, \ldots, b_n)$
- Return ciphertext $(c_1, \ldots, c_n, c', m \oplus k)$.

Decryption $\mathsf{Dec}_{\mathsf{in}}(\mathbf{sk}, c = (c_1, \ldots, c_n, c', c''))$**:**
- $b'_i \leftarrow \mathsf{Dec}(\mathbf{sk}[i], c_i)$ for $i = 1, \ldots, n$
- $k' \leftarrow \mathsf{KADec}(b'_1, \ldots, b'_n; c')$
- Return plaintext $m' = c'' \oplus k'$.

Fig. 1. Descriptions of public-key encryption schemes $\overline{\mathsf{PKE}}$ and $\mathsf{PKE}_{\mathsf{in}}$.

2. For δ being the decryption error of $\mathsf{PKE}'_{\mathsf{out}}$, we apply the transformation given in Appendix A by Lemma 4 to enforce almost-perfect correctness, reducing randomness length to $\rho' = \frac{1}{4} \cdot \log(1/\delta(\kappa)) = \omega(\log(\kappa))$ via a PRG $G : \{0,1\}^{\rho'} \to \{0,1\}^\rho$, whose existence is implied by the existence of $\mathsf{PKE}'_{\mathsf{out}}$ in a black-box fashion [30,20].
3. We then use parallel repetition of ℓ copies of $\mathsf{PKE}''_{\mathsf{out}}$ to obtain $\mathsf{PKE}^{\ell,\rho}_{\mathsf{out}}$, possibly using a PRG again to shorten the overall randomness length to ρ.

We are going to let $\mathsf{PKE}_{\mathsf{out},2} = \mathsf{PKE}^{\ell_{\mathsf{out},2},\rho_{\mathsf{out},2}}_{\mathsf{out}}$, whereas to obtain the first outer scheme $\mathsf{PKE}_{\mathsf{out},1}$, we are going to invoke Theorem 4 based on $\mathsf{PKE}^{\ell_{\mathsf{out},1},\rho}_{\mathsf{out}}$ (for some $\rho = \mathsf{poly}(\kappa)$), and then finally use a PRG to reduce the randomness length to $\rho_{\mathsf{out},1}$. The resulting scheme is then 1-CCA secure, and is almost-perfectly correct.

4.3 CCA Security of $\overline{\mathsf{PKE}}$

We turn to our main result and show that our construction $\overline{\mathsf{PKE}}$ is indeed CCA secure.

Theorem 3. *Let ε and δ be two negligible functions. Assume that PKE is (α, β)-CCA-secure, and OKA is a $(\mathcal{D}(\alpha, \beta), \varepsilon, \delta, n, \ell_{\mathsf{in}})$-one-way key-agreement protocol. Then, $\overline{\mathsf{PKE}}$ is a CCA-secure encryption scheme with negligible decryption error.*

In particular, by Propositions 2 and 3, we achieve amplification whenever $\alpha^2 > \beta + \Omega(1)$, and whenever $\alpha = 1$ and $\beta < 1 - \frac{1}{p(\kappa)}$ for some polynomial p.

Overview of the Security Proof. Towards showing the CCA security of $\overline{\mathsf{PKE}}$, we first show that it follows from Theorem 2 that the inner encryption scheme $\mathsf{PKE}_{\mathsf{in}}$ satisfies a strong adaptive security property, which we refer to as XCCA (to be read as "cross"-CCA) security. We are then going to reduce the CCA security of $\overline{\mathsf{PKE}}$ to the XCCA security of $\mathsf{PKE}_{\mathsf{in}}$ using the 1-CCA security of $\mathsf{PKE}_{\mathsf{out},1}$ and the CPA security of $\mathsf{PKE}_{\mathsf{out},2}$, combined with their almost perfect correctness. This second step resembles the proof of [23] only at a first glance, as it will require a completely different technique to handle the fact that ciphertexts of the basic scheme PKE are not sufficiently unpredictable.

Before proceeding to describing the two steps in more details, we first describe the XCCA security game. For simplicity, here we only define the XCCA game w.r.t. the concrete scheme $\mathsf{PKE}_{\mathsf{in}}$; one can easily generalize the definition to a larger class of encryption schemes whose ciphertext contains multiple component ciphertexts of a base encryption scheme, similarly to [39]; we omit the details here. The game proceeds almost identically to the CCA game except that instead of having access to the decryption oracle for the whole encryption scheme, the adversary has access to the decryption oracles of the basic encryption scheme PKE using each of the component secret keys; the i'th decryption oracle using the i'th component secret key is denoted as $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$. As a result, the adversary cannot make any after-the-fact decryption queries that is the same as any of the component ciphertexts encrypted using one of the component public keys $\mathbf{pk}[i]$ in the challenge ciphertext.

Game $\mathsf{XCCA}^{\mathcal{A}}_{\mathsf{PKE}_{\mathsf{in}}}$**:**

- **(i)** The challenger generates $(\mathbf{pk}, \mathbf{sk}) \xleftarrow{\$} \mathsf{Gen}_{\mathsf{in}}$ and $b \xleftarrow{\$} \{0,1\}$, and gives $\mathbf{pk}$ to $\mathcal{A}$.
- **(ii)** The adversary $\mathcal{A}$ asks decryption queries (i, c), which are answered with the message $\mathsf{Dec}(\mathbf{sk}[i], c) \in \{0, 1, \perp\}$.
- **(iii)** The adversary $\mathcal{A}$ outputs (m_0, m_1) with $|m_0| = |m_1| = \ell_{\mathsf{in}}$ to the challenger, and receives a challenge ciphertext $c^* \xleftarrow{\$} \mathsf{Enc}_{\mathsf{in}}(\mathbf{pk}, m_b)$, where $c^* = (c_1, \cdots, c_n, c', c'')$.
- **(iv)** The adversary $\mathcal{A}$ asks further decryption queries $(i, c \neq c_i)$, which are answered with the message $\mathsf{Dec}(\mathbf{sk}[i], c) \in \{0, 1, \perp\}$.
- **(v)** The adversary $\mathcal{A}$ outputs a bit b', and *wins* the game if $b' = b$.

Similar to the CCA game, we define the XCCA-*advantage* of the adversary $\mathcal{A}$ as $\mathbf{Adv}^{\mathsf{XCCA}}_{\mathsf{PKE}_{\mathsf{in}}}(\mathcal{A}) = 2 \cdot \Pr[b' = b] - 1$. We say that $\mathsf{PKE}_{\mathsf{in}}$ is XCCA-secure if no polynomial sized adversary can achieve a non-negligible advantage in the XCCA game.

We remark that the XCCA game is closely related to the notion of UCCA security defined in [39], and the similar notion of DCCA security in [23]: In comparison, in the UCCA security game w.r.t. $\mathsf{PKE}_{\mathsf{in}}$, the adversary only has access to the decryption oracle of the *whole encryption scheme*, but is not allowed to make any after-the-fact query that quotes any of the component ciphertexts in the challenge ciphertext (in DCCA a more fine grained control on disallowed queries is considered). As we will see shortly, the stronger security guarantee given by XCCA is crucial for our proof to succeed.

With the definition of the XCCA game in mind, the remainder of the proof proceeds via the following two lemmas.

Lemma 2. *Let ε and δ be two negligible functions. Assume that PKE is (α, β)-secure, and OKA is a $(\mathcal{D}(\alpha, \beta), \varepsilon, \delta, n, \ell_{\mathsf{in}})$-one-way KA protocol. Then, $\mathsf{PKE}_{\mathsf{in}}$ is XCCA-secure.*

Lemma 3. *Assume that $\mathsf{PKE}_{\mathsf{in}}$, $\mathsf{PKE}_{\mathsf{out},1}$ and $\mathsf{PKE}_{\mathsf{out},2}$ are respectively XCCA, 1-CCA and CPA secure, and $\mathsf{PKE}_{\mathsf{out},1}$ and $\mathsf{PKE}_{\mathsf{out},2}$ have almost-perfect correctness, then $\overline{\mathsf{PKE}}$ is CCA secure.*

We now turn to describing the high level ideas behind the proofs of both lemmas, and defer their formal proofs to Appendices D and E, respectively.

Proof Sketch of Lemma 2: We are going to use the hardcore lemma for CCA-security (Theorem 2) to show that $\mathsf{PKE}_{\mathsf{in}}$ is XCCA secure. Informally speaking, in the XCCA game, with respect to each random bit b_i used to generate the component c_i of the challenge ciphertext, the adversary is participating in an independently and randomly executed CCA game for PKE: Indeed, each random bit b_i is encrypted using an independently and randomly chosen public key $\mathbf{pk}[i]$ and random coins, and the adversary has access to the decryption oracle $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$. Thus, by the hardcore lemma, each of these CCA games has probability $1 - \beta$ of delivering an "hard instance", and thus the corresponding bit b_i remains hidden to the adversary, i.e., it looks (pseudo-)random with probability $1 - \beta$. More precisely, each triple $(b_i, \mathsf{Dec}(\mathbf{sk}[i], c_i), c_i)$, with $c_i \xleftarrow{\$} \mathsf{Enc}(\mathbf{pk}[i], b_i)$ is computationally indistinguishability from a sample

from a valid distribution from $\mathcal{D}(\alpha,\beta)$. In this case, then it simply follows from the fact that OKA is a $(\mathcal{D}(\alpha,\beta),\varepsilon,\delta,n,\ell_{\mathsf{in}})$-one-way key agreement scheme that the key k output by $\mathsf{KAEnc}(b_1,\cdots,b_n)$ remains random and thus the message m_b is hidden.

Proof Sketch of Lemma 3: We base the CCA security of $\overline{\mathsf{PKE}}$ on the XCCA security of $\mathsf{PKE}_{\mathsf{in}}$ via a black-box reduction. At a high level, the reduction $\mathcal{B}$ participates in the XCCA game for $\mathsf{PKE}_{\mathsf{in}}$ and internally emulates the CCA game for $\overline{\mathsf{PKE}}$ to a CCA-adversary $\mathcal{A}$ succeeding with non-negligible advantage γ as follows:

- It receives the public key $\mathbf{pk}$ in the XCCA game and internally generates the public key $\overline{\mathsf{pk}}$ by sampling key pairs $(\mathsf{pk}_{\mathsf{out},1},\mathsf{sk}_{\mathsf{out},1})$ and $(\mathsf{pk}_{\mathsf{out},2},\mathsf{sk}_{\mathsf{out},2})$ for the two outer schemes to produce $\overline{p}k=(\mathbf{pk},\mathsf{pk}_{\mathsf{out},1},\mathsf{pk}_{\mathsf{out},2})$ and gives it to $\mathcal{A}$.
- To emulate the challenge ciphertext c^* of $\overline{\mathsf{PKE}}$ that encrypts either m_0 or m_1 chosen by $\mathcal{A}$ in the emulated CCA game, $\mathcal{B}$ first chooses random $r_{\mathsf{out},1}$ and $r_{\mathsf{out},2}$, and obtains the challenge ciphertext c^*_{in} of $\mathsf{PKE}_{\mathsf{in}}$ that encrypts $m_b\|r_{\mathsf{out},1}\|r_{\mathsf{out},2}$ for a random $b\in\{0,1\}$ chosen in the XCCA game. It then produces c^* honestly by encrypting $c^*_{\mathsf{out},1}=\mathsf{Enc}_{\mathsf{out},1}(c^*_{\mathsf{in}};r_{\mathsf{out},1})$ and $c^*_{\mathsf{out},2}=\mathsf{Enc}_{\mathsf{out},2}(c^*_{\mathsf{in}};r_{\mathsf{out},2})$.
- Finally, it emulates the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}},\cdot)$ for $\mathcal{A}$ by using the secret key $\mathsf{sk}_{\mathsf{out},1}$ it sampled itself, as well as the decryption oracles $\{\mathsf{Dec}(\mathsf{sk}[i],\cdot)\}_{i\in[n]}$ in the XCCA game.

It is easy to see that as long as $\mathcal{A}$ does not ask any after-the-fact queries whose inner ciphertext (embedded in the first outer ciphertext) "quotes" the inner challenge ciphertexts c^*_{in}, i.e., it does not share a common component ciphertext, $\mathcal{B}$ always decrypts queries from $\mathcal{A}$ perfectly and consequently also emulates the view of $\mathcal{A}$ perfectly.

It is therefore tempting to try to show that the probability that $\mathcal{A}$ "quotes" is negligible. Indeed, this is the approach taken by [39,23]. The rationale in their proof is that if the basic scheme PKE has unpredictability — a random ciphertext of a random bit has high entropy and is hard to blindly guess — then the fact that $\mathcal{A}$ manages to quote would violate the 1-CCA security of the first outer scheme or the CPA-security of the second outer scheme. In [23], a series of hybrids is used to remove the circular dependence between the inner challenge ciphertext and the randomness used in its two outer encryptions, and move to a setting where $\mathcal{A}$'s view is *statistically* independent from the inner challenge ciphertext, but the quoting probability is negligibly close to the original one. One can then easily show that unpredictability of PKE yields that quoting occurs with negligible probability only.

Unfortunately, this approach fails completely in our setting, as our basic encryption scheme PKE does not ensure unpredictability; in fact, it is possible to build an (α,β)-CCA-secure scheme where ciphertexts have very low min-entropy. We address this via a new technique, called *heavy ciphertext pre-sampling*: We observe that if $\mathcal{A}$ can blindly guess some component ciphertext c_i in c^*_{in}, then c_i is a ciphertext value which appears with sufficiently large probability when encrypting a random bit under $\mathbf{pk}[i]$. Hence, we can hope that the same value is hit by the reduction $\mathcal{B}$ by simply generating a large number of random encryptions (of random bits) of PKE under $\mathbf{pk}[i]$; call these pre-sampled ciphertexts. Since the component ciphertexts in c^*_{in} are generated identically to the pre-sampled ciphertexts, the probability that $\mathcal{A}$'s guess collides with the former is the same as the probability it collides with any of the pre-sampled ciphertexts. Setting the size of the pre-sampling large enough, say $\mathsf{poly}(1/\varepsilon)$, the reduction can exhaust all the component ciphertexts that $\mathcal{A}$ may "quote" with probability $1-\varepsilon$, for any ε. Furthermore, due to the strong security provided by the XCCA game, the reduction $\mathcal{B}$, with access to the decryption oracles of the component ciphertexts, can obtain the decrypted values of these pre-sampled ciphertexts before-the-fact. This is crucial, since even if *we know* that a ciphertext is obtained by encrypting some bit d, its actual decryption could well be equal $1-d$ due to the weak α-correctness.

Intuitively this solves the problem, as whenever $\mathcal{A}$ makes an after-the-fact query that "quotes" c^*_{in}, $\mathcal{B}$ can still decrypt by using either the external decryption oracles (for components that do not quote) *or* the decrypted values of the pre-sampled ciphertexts (for these that quote). This will allow us to show that $\mathcal{B}$ succeeds in emulating the view of $\mathcal{A}$ with high probability, and thus the CCA security of $\overline{\mathsf{PKE}}$ reduces to the XCCA security of $\mathsf{PKE}_{\mathsf{in}}$.

Acknowledgments. We wish to thank Russell Impagliazzo and Thomas Ristenpart for insightful discussions at early stages of this work.

This work was partially supported by NSF grant CCF-1018064. This material is based on research sponsored by DARPA under agreement number FA8750-11-2-0225. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

References

1. Mihir Bellare, Russell Impagliazzo, and Moni Naor. Does parallel repetition lower the error in computationally sound protocols? In *38th Annual Symposium on Foundations of Computer Science*, pages 374–383. IEEE Computer Society Press, October 1997.
2. Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In V. Ashby, editor, *ACM CCS 93: 1st Conference on Computer and Communications Security*, pages 62–73. ACM Press, November 1993.
3. Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In Alfredo De Santis, editor, *Advances in Cryptology – EUROCRYPT'94*, volume 950 of *Lecture Notes in Computer Science*, pages 92–111. Springer, May 1994.
4. Ran Canetti, Shai Halevi, and Michael Steiner. Hardness amplification of weakly verifiable puzzles. In Joe Kilian, editor, *TCC 2005: 2nd Theory of Cryptography Conference*, volume 3378 of *Lecture Notes in Computer Science*, pages 17–33. Springer, February 2005.
5. Ran Canetti, Ronald L. Rivest, Madhu Sudan, Luca Trevisan, Salil P. Vadhan, and Hoeteck Wee. Amplifying collision resistance: A complexity-theoretic treatment. In Alfred Menezes, editor, *Advances in Cryptology – CRYPTO 2007*, volume 4622 of *Lecture Notes in Computer Science*, pages 264–283. Springer, August 2007.
6. Kai-Min Chung and Feng-Hao Liu. Parallel repetition theorems for interactive arguments. In Daniele Micciancio, editor, *TCC 2010: 7th Theory of Cryptography Conference*, volume 5978 of *Lecture Notes in Computer Science*, pages 19–36. Springer, February 2010.
7. Kai-Min Chung, Feng-Hao Liu, Chi-Jen Lu, and Bo-Yin Yang. Efficient string-commitment from weak bit-commitment. In Masayuki Abe, editor, *Advances in Cryptology – ASIACRYPT 2010*, volume 6477 of *Lecture Notes in Computer Science*, pages 268–282. Springer, December 2010.
8. Ronald Cramer, Goichiro Hanaoka, Dennis Hofheinz, Hideki Imai, Eike Kiltz, Rafael Pass, Abhi Shelat, and Vinod Vaikuntanathan. Bounded CCA2-secure encryption. In Kaoru Kurosawa, editor, *Advances in Cryptology – ASIACRYPT 2007*, volume 4833 of *Lecture Notes in Computer Science*, pages 502–518. Springer, December 2007.
9. Ronald Cramer and Victor Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In Lars R. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 45–64. Springer, April / May 2002.
10. Giovanni Di Crescenzo and Russell Impagliazzo. Security-preserving hardness-amplification for any regular one-way function. In *31st Annual ACM Symposium on Theory of Computing*, pages 169–178. ACM Press, May 1999.
11. Yevgeniy Dodis, Russell Impagliazzo, Ragesh Jaiswal, and Valentine Kabanets. Security amplification for interactive cryptographic primitives. In Omer Reingold, editor, *TCC 2009: 6th Theory of Cryptography Conference*, volume 5444 of *Lecture Notes in Computer Science*, pages 128–145. Springer, March 2009.
12. Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography. In *23rd Annual ACM Symposium on Theory of Computing*, pages 542–552. ACM Press, May 1991.
13. Cynthia Dwork, Moni Naor, and Omer Reingold. Immunizing encryption schemes from decryption errors. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology – EURO-*

CRYPT 2004, volume 3027 of *Lecture Notes in Computer Science*, pages 342–360. Springer, May 2004.

14. Yael Gertner, Tal Malkin, and Steven Myers. Towards a separation of semantic and CCA security for public key encryption. In Salil P. Vadhan, editor, *TCC 2007: 4th Theory of Cryptography Conference*, volume 4392 of *Lecture Notes in Computer Science*, pages 434–455. Springer, February 2007.
15. Oded Goldreich, Russell Impagliazzo, Leonid A. Levin, Ramarathnam Venkatesan, and David Zuckerman. Security preserving amplification of hardness. In *FOCS '90*, pages 318–326, 1990.
16. Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.
17. Iftach Haitner. A parallel repetition theorem for any interactive argument. In *50th Annual Symposium on Foundations of Computer Science*, pages 241–250. IEEE Computer Society Press, October 2009.
18. Iftach Haitner, Danny Harnik, and Omer Reingold. On the power of the randomized iterate. *SIAM J. Comput.*, 40(6):1486–1528, 2011.
19. Shai Halevi and Tal Rabin. Degradation and amplification of computational hardness. In Ran Canetti, editor, *TCC 2008: 5th Theory of Cryptography Conference*, volume 4948 of *Lecture Notes in Computer Science*, pages 626–643. Springer, March 2008.
20. Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.
21. Johan Håstad, Rafael Pass, Douglas Wikström, and Krzysztof Pietrzak. An efficient parallel repetition theorem. In Daniele Micciancio, editor, *TCC 2010: 7th Theory of Cryptography Conference*, volume 5978 of *Lecture Notes in Computer Science*, pages 1–18. Springer, February 2010.
22. Nadia Heninger, Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. Mining your ps and qs: Detection of widespread weak keys in network devices. In *Proceedings of the 21st USENIX Security Symposium*, 2012.
23. Susan Hohenberger, Allison B. Lewko, and Brent Waters. Detecting dangerous queries: A new approach for chosen ciphertext security. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 663–681. Springer, April 2012.
24. Thomas Holenstein. Key agreement from weak bit agreement. In Harold N. Gabow and Ronald Fagin, editors, *37th Annual ACM Symposium on Theory of Computing*, pages 664–673. ACM Press, May 2005.
25. Thomas Holenstein and Renato Renner. One-way secret-key agreement and applications to circuit polarization and immunization of public-key encryption. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 478–493. Springer, August 2005.
26. Thomas Holenstein and Grant Schoenebeck. General hardness amplification of predicates and puzzles (extended abstract). In Yuval Ishai, editor, *TCC 2011: 8th Theory of Cryptography Conference*, volume 6597 of *Lecture Notes in Computer Science*, pages 19–36. Springer, March 2011.
27. Nicholas Hopper, David Molnar, and David Wagner. From weak to strong watermarking. In Salil P. Vadhan, editor, *TCC 2007: 4th Theory of Cryptography Conference*, volume 4392 of *Lecture Notes in Computer Science*, pages 362–382. Springer, February 2007.
28. Russell Impagliazzo. Hard-core distributions for somewhat hard problems. In *FOCS '95*, pages 538–545, 1995.
29. Russell Impagliazzo, Ragesh Jaiswal, and Valentine Kabanets. Chernoff-type direct product theorems. *Journal of Cryptology*, 22(1):75–92, January 2009.
30. Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity-based cryptography. In *30th Annual Symposium on Foundations of Computer Science*, pages 230–235. IEEE Computer Society Press, October / November 1989.

31. Charanjit S. Jutla. Almost optimal bounds for direct product threshold theorem. In Daniele Micciancio, editor, *TCC 2010: 7th Theory of Cryptography Conference*, volume 5978 of *Lecture Notes in Computer Science*, pages 37–51. Springer, February 2010.
32. Michael J. Kearns and Leslie G. Valiant. Cryptographic limitations on learning boolean formulae and finite automata. *J. ACM*, 41(1):67–95, 1994.
33. Eike Kiltz, Payman Mohassel, and Adam O'Neill. Adaptive trapdoor functions and chosen-ciphertext security. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 673–692. Springer, May 2010.
34. Arjen K. Lenstra, James P. Hughes, Maxime Augier, Joppe W. Bos, Thorsten Kleinjung, and Christophe Wachter. Public keys. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 626–642. Springer, August 2012.
35. Ueli M. Maurer. Protocols for secret key agreement by public discussion based on common information. In Ernest F. Brickell, editor, *Advances in Cryptology – CRYPTO'92*, volume 740 of *Lecture Notes in Computer Science*, pages 461–470. Springer, August 1993.
36. Ueli M. Maurer and Stefano Tessaro. Computational indistinguishability amplification: Tight product theorems for system composition. In Shai Halevi, editor, *Advances in Cryptology – CRYPTO 2009*, volume 5677 of *Lecture Notes in Computer Science*, pages 355–373. Springer, August 2009.
37. Ueli M. Maurer and Stefano Tessaro. A hardcore lemma for computational indistinguishability: Security amplification for arbitrarily weak PRGs with optimal stretch. In Daniele Micciancio, editor, *TCC 2010: 7th Theory of Cryptography Conference*, volume 5978 of *Lecture Notes in Computer Science*, pages 237–254. Springer, February 2010.
38. Steven Myers. Efficient amplification of the security of weak pseudo-random function generators. In Birgit Pfitzmann, editor, *Advances in Cryptology – EUROCRYPT 2001*, volume 2045 of *Lecture Notes in Computer Science*, pages 358–372. Springer, May 2001.
39. Steven Myers and Abhi Shelat. Bit encryption is complete. In *50th Annual Symposium on Foundations of Computer Science*, pages 607–616. IEEE Computer Society Press, October 2009.
40. Moni Naor and Moti Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *22nd Annual ACM Symposium on Theory of Computing*. ACM Press, May 1990.
41. Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*, volume 1 of *MIT Press Books*. The MIT Press, 1994.
42. Rafael Pass and Muthuramakrishnan Venkitasubramaniam. An efficient parallel repetition theorem for Arthur-Merlin games. In David S. Johnson and Uriel Feige, editors, *39th Annual ACM Symposium on Theory of Computing*, pages 420–429. ACM Press, June 2007.
43. Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In Richard E. Ladner and Cynthia Dwork, editors, *40th Annual ACM Symposium on Theory of Computing*, pages 187–196. ACM Press, May 2008.
44. Charles Rackoff and Daniel R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In Joan Feigenbaum, editor, *Advances in Cryptology – CRYPTO'91*, volume 576 of *Lecture Notes in Computer Science*, pages 433–444. Springer, August 1992.
45. Alon Rosen and Gil Segev. Chosen-ciphertext security via correlated products. In Omer Reingold, editor, *TCC 2009: 6th Theory of Cryptography Conference*, volume 5444 of *Lecture Notes in Computer Science*, pages 419–436. Springer, March 2009.
46. Amit Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *40th Annual Symposium on Foundations of Computer Science*, pages 543–553. IEEE Computer Society Press, October 1999.
47. Stefano Tessaro. Security amplification for the cascade of arbitrarily weak PRPs: Tight bounds via the interactive hardcore lemma. In Yuval Ishai, editor, *TCC 2011: 8th Theory of Cryptography Conference*, volume 6597 of *Lecture Notes in Computer Science*, pages 37–54. Springer, March 2011.

48. Hoeteck Wee. Efficient chosen-ciphertext security via extractable hash proofs. In Tal Rabin, editor, *Advances in Cryptology – CRYPTO 2010*, volume 6223 of *Lecture Notes in Computer Science*, pages 314–332. Springer, August 2010.
49. Jürg Wullschleger. Oblivious-transfer amplification. In Moni Naor, editor, *Advances in Cryptology – EUROCRYPT 2007*, volume 4515 of *Lecture Notes in Computer Science*, pages 555–572. Springer, May 2007.
50. Andrew C. Yao. Theory and applications of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science*, pages 80–91. IEEE Computer Society Press, November 1982.

A Basic Tools

We reviews some basic constructions in the realm of secure encryption that are known from the literature, and that are going to be used throughout the paper.

FROM ONE-BIT TO MULTI-BIT CPA SECURITY. Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a CPA-secure bit-encryption scheme with negligible decryption error. Then, the following construction PKE^ℓ is a CPA-secure encryption scheme with ℓ-bit message space (we omit the simple standard proof via a hybrid argument). Key generation produces a public-key / secret-key pair $(\mathbf{pk}, \mathbf{sk})$, were $(\mathbf{pk}[i], \mathbf{sk}[i])$ are obtained by running Gen with independent randomness. Encryption of an ℓ-bit message m is defined as $\mathsf{Enc}^\ell(\mathbf{pk}, m) = \mathsf{Enc}(\mathbf{pk}[1], m_1) \,\|\, \cdots \,\|\, \mathsf{Enc}(\mathbf{pk}[\ell], m_\ell)$, where $m_1, \ldots, m_\ell$ are the bits of m. Decryption is obvious. Note that if PKE is almost-perfectly correct, then so is PKE^ℓ.

FROM CPA TO 1-CCA SECURITY. We will invoke the following result by Cramer *et al* [8] to build a 1-CCA secure encryption scheme from a CPA-secure one.

Theorem 4 ([8]). *There exists an efficient black-box construction of a 1-CCA secure public-key encryption scheme with negligible decryption error from any CPA-secure encryption scheme with negligible decryption error. Moreover, if the underlying CPA-secure scheme is almost-perfectly correct, then the resulting 1-CCA secure scheme is also almost perfectly correct.*

FROM NEGLIGIBLE ERROR TO ALMOST-PERFECT CORRECTNESS. We review a technique by Dwork, Naor, and Reingold [13] to increase correctness of schemes with negligible decryption error to almost-perfect security via sparsification of the randomness space.

Concretely, let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an encryption scheme with message space $\{0,1\}^{m(\kappa)}$ and randomness length $\rho(\kappa)$. Fix now ρ' such that $\rho'(\kappa) \leq \rho(\kappa)$ for all κ. We construct a new public-key encryption scheme $\mathsf{PKE}' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$, using a PRG $G : \{0,1\}^{\rho'(\kappa)} \to \{0,1\}^{\rho(\kappa)}$ as follows. First, we have Gen', on input 1^κ, generates $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\kappa)$ as well as a random string $\overline{r} \xleftarrow{\$} \{0,1\}^{\rho(\kappa)}$. The public key is then $\mathsf{pk}' = (\mathsf{pk}, \overline{r})$, whereas the secret key is $\mathsf{sk}' = \mathsf{sk}$. In particular we have $\mathsf{Dec}' = \mathsf{Dec}$. Moreover, we have $\mathsf{Enc}'((\mathsf{pk}, \overline{r}), m)$ first generates $r' \xleftarrow{\$} \{0,1\}^{\rho'}$, and then outputs $\mathsf{Enc}(\mathsf{pk}, m; G(r') \oplus \overline{r})$.

Lemma 4. *Assume that $G : \{0,1\}^{\rho'(\kappa)} \to \{0,1\}^{\rho(\kappa)}$ is a secure PRG, then PKE' satisfies the following two properties:*

- **(i)** *If PKE has decryption error δ, then PKE' is perfectly correct for a fraction $1 - 2^{m(\kappa)+\rho'(\kappa)} \cdot \delta(\kappa)$ of the randomness used to generate keys.*
- **(ii)** *For all adversaries $\mathcal{A}$, $\mathbf{Adv}^{\mathsf{CCA2}}_{\mathsf{PKE}'}(\mathcal{A}) \leq \mathbf{Adv}^{\mathsf{CCA2}}_{\mathsf{PKE}}(\mathcal{A}) + \nu(\kappa)$, for some negligible function ν.*

Proof. It follows directly from the fact that PKE has decryption error δ that for all $r' \in \{0,1\}^{\rho'(\kappa)}$ the following inequality holds.

$$\mathsf{Pr}\left[(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}(1^\kappa),\ \overline{r} \xleftarrow{\$} \{0,1\}^{\rho(\kappa)},\ x \xleftarrow{\$} \mathcal{M} \ :\ \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, x; G(r') \oplus \overline{r})) \neq x\right] < \delta(\kappa)\ ,$$

as $G(r') \oplus \overline{r}$ is a uniformly distributed random $\rho(\kappa)$-bit string. Therefore, by a union bound we have that,

$$\mathsf{Pr}\Big[(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}(1^\kappa),\ \overline{r} \xleftarrow{\$} \{0,1\}^{\rho(\kappa)},\ x \xleftarrow{\$} \mathcal{M} \ :\ \exists r' \in \{0,1\}^{\rho'(\kappa)} \text{ s.t. } \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, x; G(r') \oplus \overline{r})) \neq x\Big] < \delta(\kappa) 2^{\rho'(\kappa)}\ .$$

Now, the probability that (pk, sk) and $\overline{r}$ are chosen such that there exists x and r' with an decryption error, that is, $\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, x; G(r') \oplus \overline{r})) \neq x$ is upper bounded by

$$\mathsf{Pr}\Big[(\mathsf{pk},\mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}(1^\kappa),\ \overline{r} \xleftarrow{\$} \{0,1\}^{\rho(\kappa)}\ :\ \exists\, x \in \mathcal{M},\ r' \in \{0,1\}^{\rho'(\kappa)} \text{ s.t. } \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, x; G(r') \oplus \overline{r})) \neq x\Big] < \delta(\kappa) 2^{\rho'(\kappa)+m(\kappa)}\ .$$

This concludes the proof for correctness. The statement about security follows from PRG security of G, since $G(r') \oplus \overline{r}$ is pseudorandom. □

B Proof of the Abstract Hardcore Lemma (Proposition 1)

Proof (of Proposition 1). Let us fix $\gamma > 0$, and assume towards a contradiction that an $(s, s', q, \gamma(1-\varepsilon)/4, \gamma(1-\varepsilon)/4)$-$p_1$-estimator $\mathcal{E}$ for G exists, yet the claim of the proposition is false. That is, for all measures $\mathcal{M}$ with $\mu(\mathcal{M}) \geq 1-\varepsilon$, there exists an s'-size adversary $\mathcal{B}$ with $\mathbf{Adv}^G_{\mathsf{P}_\mathcal{M}}(\mathcal{B}) > \gamma$.

We can think of this situation in terms of the following two-player zero-sum game. Player 1 chooses as its pure strategy a deterministic adversary $\mathcal{B}$ of size s', whereas Player 2 chooses as its pure strategy a measure $\mathcal{M}$ such that $\mu(\mathcal{M}) = 1-\varepsilon$ and $\mathcal{M}(x) \in \{0,1\}$ for all, except at most one, x. The payoff for $\mathcal{M}$ and $\mathcal{B}$ is then exactly $\mathsf{Pr}\left[x' \xleftarrow{\$} \mathsf{P}_\mathcal{M},\ b' \xleftarrow{\$} \mathcal{B}^{\mathcal{C}(x')} :\ b' = P(x')\right]$. It is easy to see that both sets of pure strategies are finite. Given mixed strategies defined by probability distributions p_1 and p_2 over pure strategies, it is easy to verify that their expected payoff equals

$$\mathsf{Pr}\left[\mathcal{B} \xleftarrow{\$} \mathsf{O}_1,\ x' \xleftarrow{\$} \mathsf{P}_{\mathcal{M}_2},\ b' \xleftarrow{\$} \mathcal{B}^{\mathcal{C}(x')} :\ b' = P(x')\right]$$

where O_1 is an s'-size circuit sampler sampling $\mathcal{B}$ with probability $p_1(\mathcal{B})$, whereas $\mathcal{M}_2$ is such that $\mathcal{M}_2(x) = \sum_\mathcal{M} p_2(\mathcal{M}) \cdot \mathcal{M}(x)$, and has again density $\mu(\mathcal{M}_2) \geq 1-\varepsilon$.

The above assumption of the proposition being false tells us that for every mixed strategy of Player 2 there exists a pure strategy for Player 1 such that the payoff is at least $\frac{1+\gamma}{2}$. By Von Neumann's Min-Max Theorem (cf. e.g. [41]), we also have that there exists a *mixed* strategy of Player 1 such that for all pure strategies of Player 2 the expected payoff is larger than $\frac{1+\gamma}{2}$. In other words, this means that there exists a size s' circuit sampler O such that for all measures $\mathcal{M}$ with $\mu(\mathcal{M}) = 1-\varepsilon$ and $\mathcal{M}(x) \in \{0,1\}$ for all but one x,

$$\mathsf{Pr}\left[x' \xleftarrow{\$} \mathcal{M},\ \mathcal{B} \xleftarrow{\$} \mathsf{O},\ b' \xleftarrow{\$} \mathcal{B}^{\mathcal{C}(x')} :\ b' = P(x')\right] > \frac{1+\gamma}{2}\ . \tag{4}$$

We are now going to use the oracle O to obtain an oracle adversary $\mathcal{A}^\mathsf{O}$ breaking ε-security of the game G. First define δ and δ_1 such that for all $x \in \mathcal{X}$,

$$\begin{aligned} \delta(x) &:= 2 \cdot \mathsf{Pr}\left[\mathcal{B} \xleftarrow{\$} \mathsf{O},\ b' \xleftarrow{\$} \mathcal{B}^{\mathcal{C}(x)} :\ b' = P(x)\right] - 1 \\ \delta_1(x) &:= 2 \cdot \mathsf{Pr}\left[\mathcal{B} \xleftarrow{\$} \mathsf{O},\ b' \xleftarrow{\$} \mathcal{B}^{\mathcal{C}(x)} :\ b' = 1\right] - 1\ . \end{aligned} \tag{5}$$

Note that in particular $\delta(x) = \delta_1(x)$ iff $P(x) = 1$, whereas $\delta(x) = -\delta(x)$ otherwise. We also observe that $\delta_1(x) = 2 \cdot p_1^{G,\mathsf{O}}(x) - 1$, and the definition of O yields $\mathsf{E}_{x' \xleftarrow{\$} \mathsf{P}_\mathcal{M}}[\delta(x')] > \gamma$ for all measures $\mathcal{M}$ with $\mu(\mathcal{M}) = 1-\varepsilon$ and $\mathcal{M}(x) \in \{0,1\}$ for all but one x. We now are going to define one such measure: To this end we fix an ordering $x_1, x_2, \ldots$ of the elements of $\mathcal{X}$ such that $\delta(x_1) \leq \delta(x_2) \leq \cdots$, and define $\mathcal{M}^* : \mathcal{X} \to [0,1]$ to be the unique measure with $\mu(\mathcal{M}^*) = 1-\varepsilon$ and such that there exists an i^* with $\mathcal{M}^*(x_i) = 1$ for all $i < i^*$, $\mathcal{M}^*(x_{i^*}) \in [0,1]$, and $\mathcal{M}^*(x_i) = 0$ for all $i > i^*$. We let $\delta^* = \delta(x_{i^*})$ and note that $\delta^* > \gamma$, as otherwise $\mathsf{E}_{x' \xleftarrow{\$} \mathsf{P}_{\mathcal{M}^*}}[\delta(x')] \leq \gamma$.

Recall now that $\mathcal{E}$ is the $(s, s', q, \gamma(1-\varepsilon)/4, \gamma(1-\varepsilon)/4)$-$p_1$-estimator guaranteed to exist. We consider the following adversary $\mathcal{A}^\mathsf{O}$:

Adversary $\mathcal{A}^{\mathsf{O}}$: // (Inefficient) adversary for game G interacting with $\mathcal{C}(x)$

(1) Sample $\mathcal{B}_1, \ldots, \mathcal{B}_q \xleftarrow{\$} \mathsf{O}$
(2) $z \xleftarrow{\$} \mathcal{E}(\mathcal{B}_1, \ldots, \mathcal{B}_q)$
(3) $\overline{\delta}_1 \leftarrow \max\{-\delta^*, \min\{\delta^*, 2 \cdot z - 1\}\}$ // value rounded to $[-\delta^*, \delta^*]$
(4) Output 1 with probability $\frac{1}{2} + \frac{\overline{\delta}_1}{2\delta^*}$.

We now consider the experiment where $\mathcal{A}^{\mathsf{O}}$ interacts with $\mathcal{C}(x)$ for $x \xleftarrow{\$} \mathsf{P}_X$, and shows that it guesses $P(x)$ with probability larger than $\frac{1+\varepsilon}{2}$. Define the event bad that estimate of z is more than $\gamma(1-\varepsilon)/4$ off the actual value of $p_1(x)$. Recall that $\mathsf{Pr}[\mathsf{bad}] \leq \gamma(1-\varepsilon)/4$ by definition. First, note that for each input $x \in \mathcal{X}$, the probability that $\mathcal{A}^{\mathsf{O}}$ guesses the right bit $P(x)$ when interacting with $\mathcal{C}(x)$ is $\frac{1}{2} + \frac{\overline{\delta}(x)}{2\delta^*}$, where $\overline{\delta}(x) = \overline{\delta}_1$ if $P(x) = 1$, and $\overline{\delta}(x) = -\overline{\delta}_1$ if $P(x) = 0$. Then, note that conditioned on bad not occurring, we have $\left|\overline{\delta}(x) - \delta(x)\right| < \gamma(1-\varepsilon)/2$. Summarizing, the probability that $\mathcal{A}$ guesses $P(x)$ satisfies

$$\begin{aligned}
\mathsf{Pr}\left[x \xleftarrow{\$} \mathsf{P}_X,\ b' \xleftarrow{\$} \left(\mathcal{A}^{\mathsf{O}}\right)^{\mathcal{C}(x)} :\ b' = P(x)\right] &= \frac{1}{2} + \frac{1}{2\delta^*} \cdot \mathop{\mathsf{E}}_{x \xleftarrow{\$} \mathsf{P}_X}\left[\overline{\delta}(x)\right] \\
&\geq \frac{1}{2} + \frac{1}{2\delta^*} \cdot \mathop{\mathsf{E}}_{x \xleftarrow{\$} \mathsf{P}_X}\left[\delta(x)\right] - \frac{\gamma(1-\varepsilon)}{4\delta^*} - \frac{\gamma(1-\varepsilon)}{4} \\
&\geq \frac{1}{2} + \frac{1}{2\delta^*} \cdot \mathop{\mathsf{E}}_{x \xleftarrow{\$} \mathsf{P}_X}\left[\delta(x)\right] - \frac{\gamma(1-\varepsilon)}{2\delta^*}\ .
\end{aligned}$$

In particular, the probability is larger than $\frac{1+\varepsilon}{2}$ by the fact that

$$\mathop{\mathsf{E}}_{x \xleftarrow{\$} \mathsf{P}_X}\left[\delta(x)\right] = (1-\varepsilon) \cdot \mathop{\mathsf{E}}_{x' \xleftarrow{\$} \mathcal{M}^*}\left[\delta(x')\right] + \varepsilon \cdot \mathop{\mathsf{E}}_{x'' \xleftarrow{\$} \overline{\mathcal{M}^*}}\left[\delta(x'')\right] > (1-\varepsilon) \cdot \gamma + \varepsilon \cdot \delta^*\ .$$

To conclude the proof, we observe that an adversary $\mathcal{A}$, without access to O, guessing with probability also larger than $\frac{1+\varepsilon}{2}$ can be obtained by non-deterministically fixing the choice of $\mathcal{B}_1, \ldots, \mathcal{B}_q$ to the optimal one. □

C Proof of Proposition 3

Proof (of Proposition 3). As shown in [25], it suffices to give a protocol for the distribution P_{XYZ} where $X = Y$ is a random bit, Z is such that it equals $\perp$ with probability $1-\beta$, and X with probability β. Note that conditioned on $Z = \perp$, X is uniform. Clearly, $\mathsf{P}_{XYZ} \in \mathcal{D}(1, \beta)$. Let now $(X_1, Y_1, Z_1), \ldots, (X_n, Y_n, Z_n) \xleftarrow{\$} \mathsf{P}_{XYZ}$. By the Chernoff Bound, the adversary sees $Z_i = \perp$ for at least $(1-\beta) \cdot n/2 = \ell + 2\log(1/\varepsilon) + O(1)$ components of $(Z_1, \ldots, Z_n)$, except with probability $e^{-(1-\beta)n/8} = O(\sqrt{\varepsilon'} \cdot e^{-\ell(\kappa)/4})$. The encryption KAEnc sets the ciphertext C to be the seed S of a two-universal family of hash functions with input length $n(\kappa)$, and output length $\ell(\kappa)$, whereas the derived key is $K = h_S(X_1, \ldots, X_n)$. Naturally, KADec also outputs $K' = h_S(Y_1, \ldots, Y_n) = K$ given S. Given any outcome of $Z_1, \ldots, Z_n$, and conditioned on the even that at least a fraction $(1-\beta)/2$ of the coordinates has value $\perp$, $X_1, \ldots, X_n$ has entropy $\ell(\kappa) + 2\log(1/\varepsilon'(\kappa)) + O(1)$, and the resulting key K is ε'-close to uniform by the Leftover-hash Lemma [20], given $C = S$ and $Z_1, \ldots, Z_n$. □

D Formal Proof of Lemma 2

Let us fix an arbitrary polynomial p, and let γ such that $\gamma(\kappa) = 1/p(\kappa)$, and moreover, let $\mathcal{A}$ be a polynomial-size adversary. We are going to prove that for any such p and any such $\mathcal{A}$, the advantage of $\mathcal{A}$ in breaking XCCA security of $\mathsf{PKE}_{\mathsf{in}}$ is at most $n(\kappa) \cdot \gamma(\kappa) + \varepsilon(\kappa)$. Since $\varepsilon(\kappa)$ is negligible, $n(\kappa)$ is polynomial, and p can be chosen to be arbitrarily large, it follows that $\mathsf{PKE}_{\mathsf{in}}$ is XCCA secure.

Let us consider the XCCA game with adversary $\mathcal{A}$, denoted $\mathsf{XCCA}^{\mathcal{A}}(\kappa)$. First off, for ease of notation, we let $\mathbf{r}_{\mathsf{Gen}}[i]$ be the randomness used by Gen to generate the pair $(\mathbf{pk}[i], \mathbf{sk}[i]) = \mathsf{Gen}(\mathbf{r}_{\mathsf{Gen}}[i])$.

Moreover, in the process of generating the challenge ciphertext $c^* = (c_1, \ldots, c_n, c', c'') \xleftarrow{\$} \mathsf{Enc}_{\mathsf{in}}(\mathbf{pk}, m_b)$, we define $\mathbf{b}[i]$ and $\mathbf{r}[i]$ as the random bit and randomness used to generate the i-th ciphertext component $c_i = \mathsf{Enc}(\mathbf{pk}[i], \mathbf{b}[i]; \mathbf{r}[i])$, respectively. Let now $s''(\kappa)$ be the size of $\mathcal{A}$, and let $\mathcal{M}$ be the measure guaranteed to exist for the game $\mathsf{CCA2}$ by Theorem 2 for adversaries of size $s(\kappa)$, where

$$s(\kappa) = O((s''(\kappa) + \mathsf{poly}(\kappa))/\gamma(\kappa)^2(1-\beta(\kappa))^2) \ .$$

(We are not going to specify the function $s(\kappa)$ exactly, but it will clear that it can be defined more precisely depending on the construction of adversary $\mathcal{A}_i$ in the proof of Claim 1 below, which will be equal to $s'(\kappa) = s''(\kappa) + \mathsf{poly}(\kappa)$.)

The proof proceeds by introducing two additional games, $\mathsf{XCCA}_1^{\mathcal{A}}(\kappa)$ and $\mathsf{XCCA}_2^{\mathcal{A}}(\kappa)$. We are going to prove that the probabilities of $\mathcal{A}$ winning $\mathsf{XCCA}^{\mathcal{A}}(\kappa)$ and $\mathsf{XCCA}_1^{\mathcal{A}}(\kappa)$ are closely related by the hardcore lemma for CCA-security, whereas we are going to show that the probabilities are close for $\mathsf{XCCA}_1^{\mathcal{A}}(\kappa)$ and $\mathsf{XCCA}_2^{\mathcal{A}}(\kappa)$ because of the information-theoretic security of the underlying one-way key agreement protocol. Finally, it will be easy to see that no adversary can win in $\mathsf{XCCA}_2^{\mathcal{A}}(\kappa)$.

Concretely, we first modify the $\mathsf{XCCA}^{\mathcal{A}}(\kappa)$ game into game $\mathsf{XCCA}_1^{\mathcal{A}}(\kappa)$ in that the process of sampling the challenge ciphertext c^* is modified as follows:

(1) For all $i \in [n]$, first, we generate $\mathbf{r}[i]$ and $\mathbf{b}[i]$ uniformly at random and independently as before, and let the i-th ciphertext component be $c_i = \mathsf{Enc}(\mathbf{pk}[i], \mathbf{b}[i]; \mathbf{r}[i])$.
(2) Then, for all $i \in [n]$, we sample a bit $\mathbf{c}[i]$ which equals 1 with probability $p_i = \mathcal{M}(\mathbf{r}_{\mathsf{Gen}}[i], \mathbf{r}[i], \mathbf{b}[i])$, and 0 with probability $1 - p_i$. Intuitively, this corresponds to deciding whether $(\mathbf{r}_{\mathsf{Gen}}[i], \mathbf{r}[i], \mathbf{b}[i])$ are a "hard" instance or not according to the hardcore measure $\mathcal{M}$.
(3) Again for all $i \in [n]$, we sample bit $\mathbf{b}'[i]$ such that $\mathbf{b}'[i] = \mathbf{b}[i]$ if $\mathbf{c}[i] = 0$, whereas $\mathbf{b}'[i] \xleftarrow{\$} \{0,1\}$ if $\mathbf{c}[i] = 1$, i.e., in the latter case $\mathbf{b}'[i]$ is set to a uniform random bit.
(4) We then let $(k, c') \xleftarrow{\$} \mathsf{KAEnc}(\mathbf{b}'[1], \ldots, \mathbf{b}'[n])$, and $c'' = m_b \oplus k$, i.e., we use the bits $\mathbf{b}'[i]$ instead of $\mathbf{b}[i]$ to generate the key used in the encryption.
(5) The final challenge ciphertext is $c^* = (c_1, \ldots, c_n, c', c'')$.

We also consider a final game $\mathsf{XCCA}_2^{\mathcal{A}}(\kappa)$ where the key k is simply sampled randomly and independently of anything else.

The final statement follows by the straightforward combination of the following three claims, which are proved individually below.

Claim 1 *For all adversaries $\mathcal{A}$ of size s'', we have*

$$\mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(\kappa)\right] \leq \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}_1^{\mathcal{A}}(\kappa)\right] + n(\kappa) \cdot \gamma(\kappa) \ .$$

Claim 2 *For all adversaries $\mathcal{A}$ (possibly computationally unbounded),*

$$\mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}_1^{\mathcal{A}}(\kappa)\right] \leq \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}_2^{\mathcal{A}}(\kappa)\right] + \varepsilon(\kappa) \ .$$

Claim 3 *For all adversaries $\mathcal{A}$ (possibly computationally unbounded),*

$$\mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}_2^{\mathcal{A}}(\kappa)\right] = \frac{1}{2} \ .$$

Proof (Of Claim 1). We define hybrid experiments $\mathsf{XCCA}^{\mathcal{A}}(i, \kappa)$ for $i = 0, \ldots, n$ which are defined as $\mathsf{XCCA}_1^{\mathcal{A}}(\kappa)$, but with the exception that $\mathbf{b}'[j] = \mathbf{b}[j]$ holds for all $j = 1, \ldots, n-i$, whereas $\mathbf{b}'[j]$ is defined as in $\mathsf{XCCA}_1^{\mathcal{A}}(\kappa)$ otherwise. In particular, by definition it is easy to see that $\mathsf{XCCA}^{\mathcal{A}}(0, \kappa)$ and $\mathsf{XCCA}^{\mathcal{A}}(n, \kappa)$ equal $\mathsf{XCCA}^{\mathcal{A}}(\kappa)$ and $\mathsf{XCCA}_1^{\mathcal{A}}(\kappa)$, respectively. Therefore, with $q_i = \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(i, \kappa)\right]$, we clearly obtain

$$\mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(\kappa)\right] - \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}_1^{\mathcal{A}}(\kappa)\right] = \sum_{i=0}^{n-1} q_i - q_{i+1} \ .$$

We are now going to upper bound with γ the difference $q_i - q_{i+1}$ for all $i = 0, \ldots, n-1$, which concludes the proof.

To this end, assume towards a contradiction that $q_i - q_{i+1} > \gamma$. Then, we consider a variant of $\mathsf{XCCA}^{\mathcal{A}}(i+1,\kappa)$ called $\mathsf{XCCA}^{\mathcal{A}}(-i,\kappa)$ where, whenever $\mathbf{c}[i]=1$, instead of setting $\mathbf{b}'[i]$ to a random bit, we set $\mathbf{b}'[i] = 1-\mathbf{b}[i]$, and let q_{-i} be the corresponding success probability of $\mathcal{A}$ in $\mathsf{XCCA}^{\mathcal{A}}(-i,\kappa)$. Then, $q_i - q_{-i} > 2\gamma$, since $q_{i+1} = \frac{1}{2}q_i + \frac{1}{2}q_{-i}$.

Using this, we are going to construct an adversary $\mathcal{A}_i$ which contradicts the hardcore lemma. The adversary will need to know $\mathcal{M}$ in the following description, which may not be efficiently implementable. However, we note that we can de-randomize the $\mathcal{A}_i$, and in this case, it is easy to see computing $\mathcal{M}$ is not necessary any more by fixing the best randomness.

Adversary $\mathcal{A}_i(\mathsf{pk})$: // Plays the CCA2 game for PKE

(1) Sample public keys $(\mathbf{pk}[j], \mathbf{sk}[j]) \xleftarrow{\$} \mathsf{Gen}(1^\kappa)$ for $j \neq i$, and set $\mathbf{pk}[i] \leftarrow \mathsf{pk}$.
(2) Choose a simulated challenge bit $d \xleftarrow{\$} \{0,1\}$
(3) Simulate an execution of $\mathcal{A}$ in the XCCA game for $\mathsf{PKE}_{\mathsf{in}}$ with challenge bit d and keys $(\mathbf{pk}, \mathbf{sk})$. In particular, $\mathcal{A}$'s decryption queries (j, c) for $j \neq i$ are answered directly as $\mathsf{Dec}(\mathbf{sk}[j], c)$ by $\mathcal{A}_i$, whereas queries (i, c) are forwarded directly to the decryption oracle.
(4) When $\mathcal{A}$ requests the encryption of (m_0, m_1) with $|m_0| = |m_1| = \ell_{\mathsf{in}}$, the adversary $\mathcal{A}_i$ construct the challenge ciphertext $\widetilde{c}^* = (\widetilde{c}_1, \ldots, \widetilde{c}_n, \widetilde{c}', \widetilde{c}'')$ as follows, and gives it back to $\mathcal{A}$:
  (a) It sets up $\mathbf{b}[j]$, $\mathbf{c}[j]$, and $\mathbf{b}'[j]$ for $j \neq i$ as in $\mathsf{XCCA}_1^{\mathcal{A}}$.
  (b) Sets $e \xleftarrow{\$} \{0,1\}$
  (c) It obtains the challenge ciphertext $c^* = \mathsf{Enc}(\mathsf{pk}, b; r_{\mathsf{Enc}})$ from the underlying game, and sets $\widetilde{c}_i = c^*$.
  (d) It encrypts $\widetilde{c}_j \xleftarrow{\$} \mathsf{Enc}(\mathbf{pk}[j], \mathbf{b}[j])$ for $j \neq i$.
  (e) It runs $(k, \widetilde{c}') \xleftarrow{\$} \mathsf{KAEnc}(\mathbf{b}'[1], \ldots, \mathbf{b}'[i-1], e, \mathbf{b}[i+1], \ldots, \mathbf{b}[n])$ and sets $\widetilde{c}'' = m_d \oplus k$
(5) It then goes on answering after-the-fact decryption queries by $\mathcal{A}$ as above.
(6) When the adversary $\mathcal{A}$ outputs $d' \in \{0,1\}$, if $d = d'$ then $\mathcal{A}_i$ outputs e, otherwise output $1-e$.

Note that the complexity of $\mathcal{A}_i$ is $s'(\kappa)$ as defined above, and we now analyze the success probability of $\mathcal{A}_i$ in contradicting the hardcore lemma. Concretely, $\mathcal{A}_i$ interacts with $\mathcal{CCA}(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}, b)$ for $(r_{\mathsf{Gen}}, r_{\mathsf{Enc}}, b) \xleftarrow{\$} \mathsf{P}_{\mathcal{M}}$, and outputs a guess b'. We are going to prove that $b' = b$ with probability larger than $\frac{1+\gamma}{2}$, contradicting the hardcore lemma.

We first observe that the probability that $\mathcal{A}_i$ guesses b is the sum of the probability that the simulated $\mathcal{A}$ guesses d given $e = b$ and the probability that $\mathcal{A}$ is wrong in guessing d given $e = 1-b$, i.e., it equals

$$\mathsf{Pr}\left[e=b\right] \cdot \mathsf{Pr}\left[d' = d \,\middle|\, \mathbf{b}[i] = b\right] + \mathsf{Pr}\left[e = 1-b\right] \cdot \mathsf{Pr}\left[d' = 1-d \,\middle|\, \mathbf{b}[i] = 1-b\right]$$
$$= \frac{1}{2} + \frac{\mathsf{Pr}\left[d' = d \,\middle|\, e = b\right] - \mathsf{Pr}\left[d' = d \,\middle|\, e = 1-b\right]}{2},$$

since $\mathsf{Pr}\left[e=b\right] = \mathsf{Pr}\left[e = 1-b\right] = \frac{1}{2}$. Also, note that in both $\mathsf{XCCA}^{\mathcal{A}}(i,\kappa)$ and $\mathsf{XCCA}^{\mathcal{A}}(-i,\kappa)$, conditioned on $\mathbf{c}[i]=1$, the randomness $\mathbf{r}_{\mathsf{Gen}}[i]$ to generate $(\mathbf{pk}[i], \mathbf{sk}[i])$, the randomness $\mathbf{r}_{\mathsf{Enc}}[i]$ and $\mathbf{b}[i]$) used to generate the challenge ciphertext are sampled exactly according to $\mathsf{P}_{\mathcal{M}}$, as it is easy to verify, and therefore $\mathcal{A}_i$, conditioned on $e = b$, perfectly simulates an execution of $\mathsf{XCCA}^{\mathcal{A}}(i,\kappa)$ conditioned on $\mathbf{c}[i]=1$. With a similar argument, we have that $\mathcal{A}_i$, conditioned on $e = 1-b$, perfectly simulates an execution of $\mathsf{XCCA}^{\mathcal{A}}(-i,\kappa)$ conditioned on $\mathbf{c}[i]=1$. Since in both games, $\mathsf{Pr}\left[\mathbf{c}[i]=1\right] = \mu(\mathcal{M}) \leq 1$,

then for $\Delta = \mathsf{Pr}\left[d' = d \,\middle|\, e = b\right] - \mathsf{Pr}\left[d' = d \,\middle|\, e = 1-b\right]$, we have

$$\Delta = \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(i,\kappa) \,\middle|\, \mathbf{c}[i] = 1\right] - \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(-i,\kappa) \,\middle|\, \mathbf{c}[i] = 1\right]$$

$$= \frac{\mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(i,\kappa) \ \wedge\ \mathbf{c}[i] = 1\right] - \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(-i,\kappa) \ \wedge\ \mathbf{c}[i] = 1\right]}{\mu(\mathcal{M})}$$

$$= \frac{\mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(i,\kappa)\right] - \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(-i,\kappa)\right]}{\mu(\mathcal{M})} > \gamma\ ,$$

because of the fact that

$$\mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(i,\kappa) \ \wedge\ \mathbf{c}[i] = 0\right] = \mathsf{Pr}\left[\mathcal{A} \text{ wins in } \mathsf{XCCA}^{\mathcal{A}}(-i,\kappa) \ \wedge\ \mathbf{c}[i] = 0\right]\ ,$$

since both $\mathsf{XCCA}^{\mathcal{A}}(i,\kappa)$ and $\mathsf{XCCA}^{\mathcal{A}}(-i,\kappa)$ are identical as long as $\mathbf{c}[i] = 0$. □

Proof (Of Claim 2). In games $\mathsf{XCCA}_1^{\mathcal{A}}$ and $\mathsf{XCCA}_2^{\mathcal{A}}$, let us look at the distribution P_{XYZ} where $X = \mathbf{b}'[i]$, $Z = \mathsf{Enc}(\mathbf{pk}[i], \mathbf{b}[i])$, and Y is arbitrary such that it equals X with probability $\frac{1+\alpha}{2}$ (for any i, as the distribution is independent of i). Then, it is easy to see that $\mathsf{P}_{XYZ} \in \mathcal{D}(\alpha, \beta)$. Moreover, given samples $Z_1, \ldots, Z_n$, as well as $(K, C') \xleftarrow{\$} \mathsf{KAEnc}(X_1, \ldots, X_n)$ and $K' \xleftarrow{\$} \{0,1\}^{\ell_{\mathsf{in}}}$, we can simulate the challenge ciphertext distributions in $\mathsf{XCCA}_1^{\mathcal{A}}$ and $\mathsf{XCCA}_2^{\mathcal{A}}$. It is therefore easy to verify that the statistical distance between the challenge ciphertexts in $\mathsf{XCCA}_1^{\mathcal{A}}$ and $\mathsf{XCCA}_2^{\mathcal{A}}$ is at most $\varepsilon(\kappa)$, and so the difference in the probability of adversary outputting one in both games. □

Proof (Of Claim 3). This is obvious by the fact that the key k being independent of anything else ensures that the encryptions of m_0 and m_1 are perfectly indistinguishable. □

E Formal Proof of Lemma 3

Assume, towards a contradiction, that $\overline{\mathsf{PKE}}$ is not CCA secure: There is a polynomial-sized adversary $\mathcal{A}$ and a polynomial p, such that, for infinitely many $\kappa \in \mathbb{N}$, $\mathcal{A}$ achieves advantage $1/p(\kappa)$ in the CCA game with security parameter κ, denoted as $\mathsf{CCA2}^{\mathcal{A}}_{\overline{\mathsf{PKE}}}(\kappa)$. Then, we construct another polynomial-sized adversary $\mathcal{B}$ that violates the XCCA security of the inner encryption scheme $\mathsf{PKE}_{\mathsf{in}}$; in particular, B achieves advantage $1/2p(\kappa)$ in the game $\mathsf{XCCA}^{\mathcal{B}}_{\mathsf{PKE}_{\mathsf{in}}}(\kappa)$ for infinitely many κ's.

Fix one $\kappa \in \mathbb{N}$. Let $\tau(\kappa)$ be an upper bound on the the size of $\mathcal{A}$. On a high level, the machine $\mathcal{B}$ on input 1^κ participates externally in the $\mathsf{XCCA}^{\mathcal{B}}_{\mathsf{PKE}_{\mathsf{in}}}(\kappa)$ game, and internally tries to emulate an execution of $\mathsf{CCA2}^{\mathcal{A}}_{\overline{\mathsf{PKE}}}(\kappa)$; in particular, $\mathcal{B}$ has access to all the component decryption oracles $\{\mathsf{Dec}(\mathbf{sk}[i], \cdot)\}_{i\in[n]}$ externally, and needs to emulate the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ for the adversary $\mathcal{A}$ internally. More precisely, $\mathcal{B}$ proceeds in the following five stages:

Stage 1—Heavy Ciphertext Pre-sampling: Set

$$\Gamma = \Gamma(\kappa) = \omega(\log(\kappa)) \cdot 4p(\kappa) \cdot \tau(\kappa) \cdot n\ .$$

After receiving externally a public key $\mathbf{pk}$, for each component key $\mathbf{pk}[i]$, $\mathcal{B}$ internally samples Γ random component ciphertexts, that is, for each $j \in [\Gamma]$, $\mathcal{B}$ sets $e_{ij} \xleftarrow{\$} \mathsf{Enc}(\mathbf{pk}[i], r_{ij})$ for a fresh randomly sampled bit r_{ij}. Furthermore, $\mathcal{B}$ obtains the decrypted bit b_{ij} of e_{ij}, by querying the external decryption oracle $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$ on e_{ij}.[7] Then it records (e_{ij}, b_{ij}).

As we will see shortly in Stage 5, the pre-sampled ciphertexts and decrypted bits $\{(e_{ij}, b_{ij})\}_{i\in[n], j\in[\Gamma]}$ are very instrumental for emulating answers to after-the-fact queries from $\mathcal{A}$.

Stage 2—Answering Before-the-Fact Queries: $\mathcal{B}$ internally generates the public- and secret-key pairs of the two outer schemes, $(\mathsf{pk}_{\mathsf{out},i}, \mathsf{sk}_{\mathsf{out},i}) \xleftarrow{\$} \mathsf{Gen}_{\mathsf{out},i}$ for $i = 1, 2$. After sending the adversary $\mathcal{A}$ the public key $\overline{\mathsf{pk}} = (\mathsf{pk}_{\mathsf{in}}, \mathsf{pk}_{\mathsf{out},1}, \mathsf{pk}_{\mathsf{out},2})$, it emulates the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ for $\mathcal{A}$ as follows:

[7] Recall that $b_{ij} \neq r_{ij}$ may well hold, as the decryption algorithm may be subject to large error!

Given a query $c = (c_{\mathsf{out},1}, c_{\mathsf{out},2})$ from $\mathcal{A}$, $\mathcal{B}$ first decrypts $c_{\mathsf{out},1}$ using the secret key $\mathsf{sk}_{\mathsf{out},1}$ to obtain the inner ciphertext $c'_{\mathsf{in}} = (c'_1, \cdots, c'_n, , \gamma', \eta')$. $\mathcal{B}$ does not know the secret key $\mathbf{sk}$ for decrypting the inner ciphertext; instead, it uses the external decryption oracles $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$ to decrypt each of the component ciphertexts $c'_1, \cdots, c'_n$ to obtain the encrypted bits $b'_1, \cdots, b'_n$; it then recovers the encrypted message $(m', r'_{\mathsf{out},1}, r'_{\mathsf{out},2})$ from γ' and η' as algorithm $\mathsf{Dec}_{\mathsf{in}}$ does. Finally, it checks consistency of the ciphertext, that is, whether $c_{\mathsf{out},i} = \mathsf{Enc}_{\mathsf{out},i}(\mathsf{pk}_{\mathsf{out},i}, c'_{\mathsf{in}}; r'_{\mathsf{out},i})$ for $i = 1, 2$, and returns m' if the ciphertext is consistent and $\perp$ otherwise.

Stage 3—Generating Challenge Messages: After receiving the two challenge messages m^*_0, m^*_1 from $\mathcal{A}$, $\mathcal{B}$ samples random strings $r^*_{\mathsf{out},1}$, $r^*_{\mathsf{out},2}$, and sends externally two challenge messages $m^*_0 \| r^*_{\mathsf{out},1} \| r^*_{\mathsf{out},2}$ and $m^*_1 \| r^*_{\mathsf{out},1} \| r^*_{\mathsf{out},2}$.

Stage 4—Emulating Challenge Ciphertext: Upon receiving the challenge ciphertext

$$c^*_{\mathsf{in}} = \mathsf{Enc}_{\mathsf{in}}(\mathbf{pk}, m^*_b \| r^*_{\mathsf{out},1} \| r^*_{\mathsf{out},2})$$

for some random bit b, $\mathcal{B}$ generates the challenge ciphertext of $\overline{\mathsf{PKE}}$ for $\mathcal{A}$ by encrypting c^*_{in} using public keys of the two outer schemes and randomness $r^*_{\mathsf{out},1}$ and $r^*_{\mathsf{out},2}$, that is, $c^*_{\mathsf{out},i} = \mathsf{Enc}_{\mathsf{out},i}(\mathsf{pk}_{\mathsf{out},i}, c^*_{\mathsf{in}}; r^*_{\mathsf{out},i})$ for $i = 1, 2$. It then sends $(c^*_{\mathsf{out},1}, c^*_{\mathsf{out},2})$ to $\mathcal{A}$.

Stage 5—Answering After-the-Fact Queries: $\mathcal{B}$ emulates answers to $\mathcal{A}$'s after-the-fact decryption queries $(c_{\mathsf{out},1}, c_{\mathsf{out},2})$ almost the same as it did with before-the-fact queries in Stage 2, except that, after obtaining the inner ciphertext $c'_{\mathsf{in}} = (c'_1, \cdots, c'_n, \gamma', \eta')$, if one of the component ciphertext c'_i coincides with the corresponding component in the challenge ciphertext c^*_i, $\mathcal{B}$ cannot query the external decryption oracle $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$ (after-the-fact) to obtain the decrypted bit b'_i.

When this happens for some component, $\mathcal{B}$ simply checks whether c'_i is one of the pre-sampled component ciphertext e_{ij}, if so it also obtains the decrypted bit $b'_i = b_{ij}$; otherwise, it outputs fail.

Output: Finally, $\mathcal{B}$ outputs the bit b' returned by $\mathcal{A}$.

We show that except with probability $1/2p(\kappa)$, $\mathcal{B}$ in game $\mathsf{XCCA}^{\mathcal{B}}_{\mathsf{PKE}_{\mathsf{in}}}(\kappa)$ emulates the view of $\mathcal{A}$ in $\mathsf{CCA2}^{\mathcal{A}}_{\overline{\mathsf{PKE}}}(\kappa)$ perfectly; then if $\mathcal{A}$ has advantage $1/p(\kappa)$ in the CCA game, $\mathcal{B}$ achieves $1/2p(\kappa)$ advantage in the XCCA game. Note that $\mathcal{B}$ emulates the public key and challenge ciphertext of $\overline{\mathsf{PKE}}$ for $\mathcal{A}$ perfectly; furthermore, it emulates the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ for $\mathcal{A}$ perfectly in Stage 2 by decrypting before-the-fact queries as the algorithm $\overline{\mathsf{Dec}}$ does using the secret key $\mathsf{sk}_{\mathsf{out},1}$ and the external component decryption oracles $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$. For the same reason, $\mathcal{B}$ also emulates the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ perfectly in Stage 5, provided that it can decrypt all the component ciphertexts using either the external decryption oracles or the set of pre-sampled component ciphertexts and decrypted bits. Therefore, conditioned on that $\mathcal{B}$ does not output fail, it emulates the view of $\mathcal{A}$ perfectly. As we show below in Claim 4, the probability that $\mathcal{B}$ outputs fail is bounded by $1/2p(\kappa)$; then, except with probability $1/2p(\kappa)$, $\mathcal{B}$ emulates the view of $\mathcal{A}$ perfectly, and thus has advantage $1/2p(\kappa)$ in the XCCA game. This contradicts with the XCCA security of $\mathsf{PKE}_{\mathsf{in}}$ and concludes the lemma.

Claim 4 *For all sufficiently large $\kappa \in \mathbb{N}$, the probability that $\mathcal{B}$ outputs* fail *in an execution of $\mathsf{XCCA}^{\mathcal{B}}_{\mathsf{PKE}_{\mathsf{in}}}(\kappa)$ is smaller than $1/2p(\kappa)$.*

Proof. Recall that the outer encryption scheme $\mathsf{PKE}_{\mathsf{out},i}$, $i = 1, 2$, has almost perfect correctness, that is, with overwhelming probability, a randomly generated key pair $(\mathsf{pk}_{\mathsf{out},i}, \mathsf{sk}_{\mathsf{out},i})$ has perfect correctness. It is easy to see that it suffices to show that conditioned on that the two outer encryption key pairs $(\mathsf{pk}_{\mathsf{out},1}, \mathsf{sk}_{\mathsf{out},1})$ and $(\mathsf{pk}_{\mathsf{out},2}, \mathsf{sk}_{\mathsf{out},2})$ sampled by $\mathcal{B}$ have perfect correctness, the probability that $\mathcal{B}$ outputs fail is smaller than $1/3p(\kappa)$. Therefore, below we bound the probability that $\mathcal{B}$ outputs fail, assuming implicitly that the outer encryption keys have perfect correctness; when referring to the value encrypted in the an outer ciphertext, we mean the unique value decrypted from that ciphertext.

Recall that $\mathcal{B}$ outputs fail when $\mathcal{A}$ (in emulation by $\mathcal{B}$) makes an after-the-fact decryption query $(c_{\mathsf{out},1}, c_{\mathsf{out},2})$ that has the inner ciphertext $c'_{\mathsf{in}} = (c'_1, \cdots, c'_n, \gamma', \eta')$ decrypted from $c_{\mathsf{out},1}$ "quote" the inner ciphertext $c^*_{\mathsf{in}} = (c^*_1, \cdots, c^*_n, \gamma^*, \eta^*)$ of the challenge ciphertext (i.e., $\exists i \in [n]$ such that $c'_i = c^*_i$), *yet* the quoted component ciphertext c^*_i is not one of the pre-sampled component ciphertexts (i.e.,

$c_i^* \neq e_{ij}$ for all $j \in [\Gamma]$); we denote this event as quote. Thus, it is equivalent to bound the probability that event quote occurs in an execution with $\mathcal{A}$ as emulated by $\mathcal{B}$. Towards this, we introduce a sequence of hybrids H_0 to H_5, where H_0 emulates the view of $\mathcal{A}$ identically as $\mathcal{B}$ does. We show that in every two subsequent hybrids, the probabilities that event quote occurs differ by at most a negligible amount and the probability that event quote occurs in H_5 is bounded by $1/4p(\kappa)$. Therefore, we derive that the probability that quote occurs in H_0 is at most $1/3p(\kappa)$, and so is the probability that quote occurs in an execution of $\mathcal{B}$.

Hybrid H_0 internally runs $\mathcal{A}$ and emulates its view identically as $\mathcal{B}$ does, by acting as $\mathcal{B}$ and entities in the game $\mathsf{XCCA}^{\mathcal{B}}_{\mathsf{PKE}_{\mathsf{in}}}(\kappa)$ (including the external challenger and the decryption oracles $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$). By construction, we have that:

$$\mathsf{Pr}[\mathsf{quote} \text{ occurs in execution of } \mathcal{B}] = \mathsf{Pr}[\mathsf{quote} \text{ occurs in } H_0] \ .$$

Hybrid H_1 proceeds identically to H_0, except that the inner-ciphertext c^*_{in} of the challenge ciphertext encrypts an all-zero string, instead of $(m^*_b, r^*_{\mathsf{out},1}, r^*_{\mathsf{out},2})$ (for some random bit b). We claim that the probability that quote occurs in H_1 is negligibly close to that in H_0. First note that in both H_0 and H_1, whether the event quote occurs can be efficiently decided using the secret key $\mathsf{sk}_{\mathsf{out},1}$ of the first outer encryption scheme. Furthermore, since in the two hybrids the secret key $\mathbf{sk}$ of the inner encryption scheme is only used for emulating the component decryption oracles $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$, if the probabilities that quote occurs differ by a non-negligible amount, we can use $\mathcal{A}$ to construct a machine to violate the XCCA security of the inner encryption scheme. This summarized by the following subclaim.

SubClaim 1 *There is a negligible function μ_1, such that, for all $\kappa \in \mathbb{N}$,*

$$|\mathsf{Pr}[\mathsf{quote} \textit{ occurs in } H_0] - \mathsf{Pr}[\mathsf{quote} \textit{ occurs in } H_1]| \leq \mu_1(\kappa) \ .$$

Hybrid H_2 proceeds identically to H_1, except that the second outer-ciphertext $c^*_{\mathsf{out},2}$ of the challenge ciphertext encrypts an all-zero string, instead of the inner-ciphertext c^*_{in}. Note that the secret key $\mathsf{sk}_{\mathsf{out},2}$ of the second outer encryption scheme is never used in the execution of H_1 and H_2, and the random string $r_{\mathsf{out},2}$ used to generate $c_{\mathsf{out},2}$ is uniformly randomly sampled and independent of all other messages. (Furthermore, as discussed above, whether quote occurs or not can be efficiently decided using the first outer secret key $\mathsf{sk}_{\mathsf{out},1}$.) Therefore, it follows from the semantic security of the second outer encryption scheme $\mathsf{PKE}_{\mathsf{out},2}$ that the probabilities that quote occurs in H_1 and H_2 differ by at most a negligible amount.

SubClaim 2 *There is a negligible function μ_2, such that, for all $\kappa \in \mathbb{N}$,*

$$|\mathsf{Pr}[\mathsf{quote} \textit{ occurs in } H_1] - \mathsf{Pr}[\mathsf{quote} \textit{ occurs in } H_2]| \leq \mu_2(\kappa) \ .$$

Hybrid H_3 proceeds identically to H_2, except that H_3 emulates the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ for $\mathcal{A}$ using the secret key $\mathsf{sk}_{\mathsf{out},2}$ of the second outer encryption scheme (as opposed to $\mathsf{sk}_{\mathsf{out},1}$ of the first outer encryption scheme), and the secret key $\mathbf{sk}$ of the inner ciphertext (instead of the external decryption oracles $\mathsf{Dec}(\mathbf{sk}[i], \cdot)$ together with the pre-sampled component ciphertexts and decrypted bits $\{e_{ij}, b_{ij}\}_{i\in[n],j\in[\Gamma]}$). More precisely, H_3 emulates $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ as follows: Upon receiving a decryption query $(c_{\mathsf{out},1}, c_{\mathsf{out},2})$ from $\mathcal{A}$, H_3 decrypts the second outer ciphertext $c_{\mathsf{out},2}$ using $\mathsf{sk}_{\mathsf{out},2}$ to obtain an inner ciphertext $c''_{\mathsf{in}} = (c''_1, \cdots, c''_n, \gamma'', \eta'')$; it then decrypts each of c''_i using $\mathbf{sk}[i]$ to obtain the decrypted bit b''_i, and recovers the message $(m'', r''_{\mathsf{out},1}, r''_{\mathsf{out},2})$ as algorithm $\mathsf{Dec}_{\mathsf{in}}$ does; finally it check consistency of the ciphertext by checking whether $c_{\mathsf{out},i} = \mathsf{Enc}_{\mathsf{out},i}(\mathsf{pk}_{\mathsf{out},1}, c''_{\mathsf{in}}; r''_{\mathsf{out},i})$ for $i = 1, 2$. Note that this way of emulating the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ uses only $\mathsf{sk}_{\mathsf{out},2}$ and $\mathbf{sk}$, but not $\mathsf{sk}_{\mathsf{out},1}$ and $\{e_{ij}, b_{ij}\}_{i\in[n],j\in[\Gamma]}$; therefore H_3 never outputs fail. However, the event quote (defined w.r.t. the unique inner-ciphertexts encrypted in the first outer ciphertexts of the decryption queries from $\mathcal{A}$) is still well defined.

We claim that the probability that quote occurs in H_2 and H_3 are identical. This is because the only difference between the two hybrids lies in how the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ is emulated.

In H_2, before event quote occurs, the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ are emulated perfectly. In H_3, given that the key pairs of the two outer encryption schemes have perfect correctness, the plaintext computed using $sk_{\mathsf{out},2}$ and $\mathbf{sk}$ is always the same as that returned by the real decryption oracle using $sk_{\mathsf{out},1}$ and $\mathbf{sk}$. Thus before quote occurs, the views of $\mathcal{A}$ are identical in the two hybrids and so are the probabilities that event quote occurs.

$$\mathsf{Pr}[\mathsf{quote} \text{ occurs in } H_2] = \mathsf{Pr}[\mathsf{quote} \text{ occurs in } H_3] \ .$$

Hybrid H_4 proceeds identically to H_3, except that the first outer-ciphertext $c^*_{\mathsf{out},1}$ of the challenge ciphertext encrypts an all-zero string, instead of the inner-ciphertext c^*_{in}. Note that in both H_3 and H_4 the secret key $\mathsf{sk}_{\mathsf{out},1}$ of the first outer encryption scheme is never used. It seems that, as in Hybrid H_2, it should follow directly from the semantic security of the first outer encryption scheme that the probabilities that quote occurs in the two hybrids differ by at most a negligible amount. However, this argument does not go through: Unlike in H_2 where whether quote occurs can be efficiently decided using $\mathsf{sk}_{\mathsf{out},1}$, when relying on the semantic security of $\mathsf{PKE}_{\mathsf{out},1}$, $\mathsf{sk}_{\mathsf{out},1}$ cannot be used and thus event quote cannot be efficiently detected. This problem can be circumvented by instead relying on the CCA-1 security of $\mathsf{PKE}_{\mathsf{out},1}$.
Assume for contradiction that the probability that quote occurs in H_3 and H_4 differ by an inverse polynomial probability $1/q(\kappa)$, then we can construct an adversary $\mathcal{C}$ that violates the 1-CCA security of $\mathsf{PKE}_{\mathsf{out},1}$. Adversary $\mathcal{C}$ after receiving a public key $\mathsf{pk}_{\mathsf{out},1}$ and a challenge ciphertext $c^*_{\mathsf{out},1}$ encrypting either c^*_{in} or an all-zero string, emulates internally an execution of H_3 or H_4 with $\mathcal{A}$, and outputs the first outer-ciphertext $c_{\mathsf{out},1}$ of a randomly chosen decryption query from $\mathcal{A}$; let c_{in} be the inner-ciphertext encrypted in $c_{\mathsf{out},1}$. By construction of $\mathcal{C}$, the view of $\mathcal{A}$ in emulation by $\mathcal{C}$ when it receives externally an encryption to c^*_{in} is identical to that in H_3, and that when it receives an encryption to an all-zero string is identical to that in H_4. Since $\mathcal{A}$ takes at most $\tau(\kappa)$ steps, it follows from our hypothesis that the probabilities that c_{in} "quotes" c^*_{in} differ by at least $1/q(\kappa)\tau(\kappa)$, when $\mathcal{C}$ receives an encryption to c^*_{in} or an all-zero string. This violates the 1-CCA security of $\mathsf{PKE}_{\mathsf{out},1}$.

SubClaim 3 *There is a negligible function μ_3, such that, for all $\kappa \in \mathbb{N}$,*

$$|\mathsf{Pr}[\mathsf{quote} \textit{ occurs in } H_3] - \mathsf{Pr}[\mathsf{quote} \textit{ occurs in } H_4]| \leq \mu_3(\kappa) \ .$$

Hybrid H_5 proceeds identically to H_4, except that it samples the inner-ciphertext c^*_{in} of the challenge ciphertext and the pre-sampled component-ciphertexts $\{e_{ij}\}_{i\in[n],j\in[\Gamma]}$ at the end of the execution. (The decrypted bits b_{ij}'s of e_{ij}'s are no longer computed.) Note that this is possible because already in H_4, c^*_{in} and e_{ij}'s are no longer used (as the challenge ciphertext $(c^*_{\mathsf{out},1}, c^*_{\mathsf{out},2})$ consists of encryptions to all-zero strings, and the decryption oracle $\overline{\mathsf{Dec}}(\overline{\mathsf{sk}}, \cdot)$ is emulated using $\mathsf{sk}_{\mathsf{out},2}$ and $\mathbf{sk}$). Still, the event quote defined w.r.t. the decryption queries from $\mathcal{A}$, c^*_{in} and e_{ij}'s remains well defined. Since the view of $\mathcal{A}$ in H_5 is identical to that in H_4, we have:

$$\mathsf{Pr}[\mathsf{quote} \text{ occurs in } H_4] = \mathsf{Pr}[\mathsf{quote} \text{ occurs in } H_5]$$

Next we show that the probability that quote occurs in H_5 is bounded by $1/4p(\kappa)$.

SubClaim 4

$$\mathsf{Pr}[\mathsf{quote} \textit{ occurs in } H_5] \leq \frac{1}{4p(\kappa)} \ .$$

Proof. Recall that $\mathcal{A}$ takes at most $\tau(\kappa)$ steps; therefore it makes at most $\tau(\kappa)$ decryption queries in its execution. Towards bounding the probability that quote occurs, we show that for all $i \in [n]$ and $q \in [\tau(\kappa)]$, the probability that quote occurs w.r.t. the i'th component ciphertext in the q'th decryption query from $\mathcal{A}$ is bounded by $1/4p(\kappa)\tau(\kappa)n$. Formally, let $c^q_{\mathsf{in}} = (c^q_1, \cdots, c^q_n, \gamma^q, \eta^q)$ denote the inner-ciphertext encrypted in the first outer-ciphertext of the q'th decryption query, and $c^*_{\mathsf{in}} = (c^*_1, \cdots, c^*_n, \gamma^*, \eta^*)$ that in the challenge ciphertext. Recall that H_5 proceeds by first completing an execution with $\mathcal{A}$, and then sampling the challenge inner-ciphertext c^*_{in} and component ciphertexts

$\{e_{ij}\}$; let $r_{\mathcal{A}}$ denote the randomness used in the execution with $\mathcal{A}$, r_c in generating c_i^*, and r_e in generating the e_{ij}'s. We show

$$\forall i \in [n], q \in [\tau(\kappa)], \qquad \mathsf{Pr}_{r_{\mathcal{A}}, r_c, r_e}[c_i^q = c_i^* \ \wedge \ \forall j \in [\Gamma], \ c_i^q \neq e_{ij}] \leq \frac{1}{4p(\kappa)\tau(\kappa)n} \ , \qquad (*)$$

Then by a union bound, it follows that the probability that quote occurs is bounded by $1/4p(\kappa)$. To show equation (*), since in H_5, the execution with $\mathcal{A}$ completes before c_{in}^* and $\{e_{ij}\}_{i\in[n],j\in[\Gamma]}$ are sampled, it suffices to show that fixing any execution with $\mathcal{A}$ which decides a c_i^q, the probability that c_i^q coincides with c_i^* but not any of e_{ij}'s is bounded by $1/4p(\kappa)\tau(\kappa)n$, that is, equation (*) holds for all possible $r_{\mathcal{A}}$ deciding c_i^q. In other words, we show:

$$\forall i \in [n], q \in [\tau(\kappa)], r_{\mathcal{A}} \qquad \mathsf{Pr}_{r_c, r_e}[c_i^q = c_i^* \ \wedge \ \forall j \in [\Gamma], \ c_i^q \neq e_{ij}] \leq \frac{1}{4p(\kappa)\tau(\kappa)n} \ .$$

Assume for contradiction that there is a fixed execution with $\mathcal{A}$ deciding c_i^q, for which the above inequality does not hold. Then it implies that:

$$\mathsf{Pr}_{r_c}[c_i^q = c_i^*] > \frac{1}{4p(\kappa)\tau(\kappa)n} \ .$$

Since each e_{ij} is generated identically as the i'th component ciphertext c_i^* of the challenge ciphertext is generated, we have that for all $j \in [\Gamma]$.

$$\mathsf{Pr}_{r_e}[c_i^q = e_{ij}] > \frac{1}{4p(\kappa)\tau(\kappa)n}$$

Furthermore, since all e_{ij}'s are generated randomly and independently, we have that,

$$\mathsf{Pr}_{r_e}[\forall j \in [\Gamma], \ c_i^q \neq e_{ij}] < \left(1 - \frac{1}{4p(\kappa)\tau(\kappa)n}\right)^{\Gamma} \leq \mathrm{negl}(\kappa) \ .$$

The last inequality holds since $\Gamma = \omega(\log(\kappa))4p(\kappa)\tau(\kappa)n$. This gives a contradiction, and concludes the proof of this subclaim.