Encryption Schemes Secure against Chosen-Ciphertext Selective Opening Attacks

Serge Fehr[1], Dennis Hofheinz[2,⋆], Eike Kiltz[1,⋆⋆], and Hoeteck Wee[3,⋆⋆⋆]

[1] CWI, Amsterdam
[2] Karlsruhe Institute of Technology
[3] Queens College, CUNY

Abstract. Imagine many small devices send data to a single receiver, encrypted using the receiver's public key. Assume an adversary that has the power to adaptively corrupt a subset of these devices. Given the information obtained from these corruptions, do the ciphertexts from *uncorrupted* devices remain secure?

Recent results suggest that conventional security notions for encryption schemes (like IND-CCA security) do not suffice in this setting. To fill this gap, the notion of *security against selective-opening attacks (SOA security)* has been introduced. It has been shown that lossy encryption implies SOA security against a *passive*, i.e., only eavesdropping and corrupting, adversary (SO-CPA). However, the known results on SOA security against an *active* adversary (SO-CCA) are rather limited. Namely, while there exist feasibility results, the (time and space) complexity of currently known SO-CCA secure schemes depends on the number of devices in the setting above.

In this contribution, we devise a new solution to the selective opening problem that does not build on lossy encryption. Instead, we combine techniques from non-committing encryption and hash proof systems with a new technique (dubbed "cross-authentication codes") to glue several ciphertext parts together. The result is a rather practical SO-CCA secure public-key encryption scheme that does not suffer from the efficiency drawbacks of known schemes. Since we build upon hash proof systems, our scheme can be instantiated using standard number-theoretic assumptions such as decisional Diffie-Hellman (DDH), decisional composite residuosity (DCR), and quadratic residuosity (QR). Besides, we construct a conceptually very simple and comparatively efficient SO-CPA secure scheme from (slightly enhanced) trapdoor one-way permutations.

We stress that our schemes are completely independent of the number of challenge ciphertexts, and we do not make assumptions about the underlying message distribution (beyond being efficiently samplable). In particular, we do not assume efficient conditional re-samplability of the message distribution. Hence, our schemes are secure in *arbitrary* settings, even if it is not known in advance how many ciphertexts might be considered for corruptions.

⋆ Part of this work performed while at CWI. Supported by an NWO Veni grant.
⋆⋆ Supported by the research program Sentinels (`http://www.sentinels.nl`).
⋆⋆⋆ Supported by PSC-CUNY Award # 6014939 40 and the US Army Research laboratory and the UK Ministry of Defence under agreement number W911NF-06-3-0001.

H. Gilbert (Ed.): EUROCRYPT 2010, LNCS 6110, pp. 381–402, 2010.

1 Introduction

The generally accepted notion of security for public-key encryption is indistinguishability of ciphertexts under chosen-ciphertext attacks (IND-CCA, cf. [27, 30, 16]). For IND-CCA security, it must not be possible to tell which one of two adversarially chosen messages is encrypted, even when given access to a decryption oracle. The notion of IND-CCA security has proved extremely useful. On the one hand, it essentially captures the notion of a secure channel against active attacks (see [9, 12]). On the other hand, efficient IND-CCA secure encryption schemes can be constructed under standard number-theoretic assumptions (e.g., [13, 26, 23]).

However, there are realistic scenarios in which IND-CCA security is not known to provide security. For instance, consider a setting in which a large (and possibly a priori unknown) number of small devices send data to a single receiver. Each device encrypts its messages using the receiver's public key. Now assume an adversary that has the power to adaptively corrupt a subset of these devices. Say that, upon corrupting a device, the adversary learns the device's complete internal state, including the random coins used during previous encryptions. In that sense, the adversary may ask for *selective openings* of ciphertexts. The obvious question is: do the *unopened* ciphertexts remain secure? That is, can the adversary conclude anything about the plaintexts sent by uncorrupted devices, beyond of course what is implied already by the revealed plaintexts? While intuitively, the answer should be "no" for a secure public-key encryption system, IND-CCA security does not seem to be immediately useful in this setting. (E.g., [21] shows that whenever encryption constitutes a commitment to the respective message, the scheme cannot be proven secure using black-box techniques. This holds independent of whether the scheme is IND-CCA secure or not.) We clarify that the problem becomes moot if the senders can erase their randomness after sending the encrypted messages (cf. [1]). However, reliable erasure is difficult on a real system. As such, we will only focus on solutions that do not require erasures.

So far, only little is known on the construction of public key encryption schemes that are secure under selective opening attacks (SOA secure) as discussed above. Concretely, [3, 5] have shown that every *lossy* encryption scheme (cf. [29]) is SOA secure against *passive* (i.e., eavesdropping) adversaries. This yields a generic construction of SOA secure encryption that allows for fairly efficient instantiations. However, [3, 5] leave open the question of designing schemes that are SOA secure against *active* adversaries.

Our contribution. We construct practical public key encryption schemes that are SO-CCA secure, i.e., SOA secure against active attacks. Interestingly, we substantially deviate from previous techniques to obtain SOA security. To explain our approach, let us briefly sketch how [3, 5] employ lossy encryption to achieve SOA security.

(Passive) SOA security from lossy encryption. Lossy encryption schemes have the property that the scheme's "real" public key can be substituted with a "lossy" public key. Real and lossy keys are computationally indistinguishable, so – at least in a passive security experiment – this change cannot be detected by an adversary. Now lossy keys have the property that encryptions performed with them yield "lossy" ciphertexts that are statistically independent of the plaintext. In particular, a given lossy ciphertext can

be – in general inefficiently – explained (or, opened) as an encryption of an arbitrary plaintext. Consequently, an SOA adversary cannot distinguish real keys, ciphertexts, and openings from those implied by lossy keys. But in the lossy case, the adversary's view is statistically independent of unopened messages; SOA security follows.

SOA-CCA security from lossy encryption, and its limitations. Now consider an active SOA adversary (i.e., one that is equipped with a decryption oracle). To prove SO-CCA security, now additionally adversarial decryption queries have to be answered. Obviously, this is impossible with fully lossy keys (i.e., keys that *always* encrypt to ciphertexts that are independent of the plaintext). In the IND-CCA case (see [29]), the solution to this dilemma is to make sure that *only* the challenge ciphertext is lossy. Technically, the security proof consider an "all-but-one" (ABO) public key. The ABO public key only encrypts the challenge ciphertext into a lossy ciphertext, and the corresponding ABO secret key can be used to decrypt any ciphertext except the lossy challenge ciphertext (and thus can be used to answer decryption queries).

This technique works well in the IND-CCA case, since there we have only one challenge ciphertext. However, with SOA security, we have to deal with a – possibly huge – vector of challenge ciphertexts that correspond to all openable ciphertexts. We would need "all-but-many" public keys that allow to make *only* the challenge ciphertexts lossy. (In fact, this is the exact approach taken by [20].) However, a counting argument shows that now the public-key size is at least linear in the maximal number of challenge ciphertexts. In realistic examples as the one above, there might be thousands of openable challenge ciphertexts. Hence, the lossy encryption approach leads to fairly impractical schemes, which have huge keys, and which actually achieve only *bounded* SO-CCA security. The latter means that the number of challenge ciphertexts for which the scheme is secure, is limited once the public key is chosen. If the number of potentially openable ciphertexts happens to exceed this limit, nothing is guaranteed anymore.

Another limitation of this approach is that, unless a lossy ciphertext is *efficiently* openable (a property which is not known to hold for most lossy encryption schemes), the lossy encryption paradigm only achieves (bounded) so-called IND-SO-CCA security. This in particular means that SOA security is only guaranteed for joint message distributions that are *efficiently conditionally re-samplable*. This means that even when *conditioned* on an arbitrary fixed subvector of messages, the remaining messages need to be efficiently samplable.[1] Many realistic settings (e.g., encryptions of ciphertexts, commitments, or signatures for fixed messages) correspond to *not* efficiently conditionally re-samplable message distributions. So without extra assumptions, lossy encryption implies only bounded SOA security in a restricted class of settings.

Our approach. We show SOA security using techniques from non-committing, resp. deniable encryption (e.g., [10, 15, 24, 11]). Non-committing encryption (NCE) schemes allow for "equivocable" ciphertexts that are computationally indistinguishable from real

[1] We remark that it is not obvious from [20] that their IND-SO-CCA secure scheme (Section 4.6) requires this additional condition on the distribution of the challenge messages. However, if this condition is not satisfied, then the challenger in the ideal game (in the definition of IND-SO-CCA security) is *inefficient*, and as such it *cannot* be argued in the security proof that in the ideal game the real public key can be replaced by a lossy key.

ciphertexts, but can be efficiently opened arbitrarily.[2] To achieve security against selective opening attacks, we rely on an idea from the deniable encryption scheme of Canetti et al. [11]. In their scheme, an encryption of 0 corresponds to a random string and that of 1 corresponds to a pseudorandom string (with a sparse range); it is easy to see that 1-encryptions are equivocable and can be opened as both 0 and 1. We will similar ideas in our schemes, which allows us to turn *all* SOA challenge ciphertexts into equivocable ones *one by one*. (Recall that in a sense, the reason why the lossy encryption paradigm does not mesh well with SO-CCA security is that lossy encryption only provides a handle to turn *all* challenge ciphertexts into lossy ones *at once*.) Finally, when all challenge ciphertexts are equivocable, we can argue that they do not contain any information about the unopened messages, and SOA security follows. Unlike previous constructions based on lossy encryption, we do not change the distribution of the public key in either our simulation or in the analysis.

We stress that the complexity our scheme does not depend on the number of challenge ciphertexts. So at the time of, say, constructing a PKI using our scheme, the number of potentially openable ciphertexts does not have to be known. We also remark that our approach achieves SOA security against *arbitrary* message distributions. We do not need to make extra assumptions on the underlying encryption scheme, or on the message distribution.

We first showcase our approach with a conceptually very simple scheme that is SO-CPA secure, i.e., SOA secure against passive attacks. Interestingly, we can base our proof upon general complexity assumptions, i.e., on the assumption of (a slightly enhanced version of) trapdoor one-way permutations. Going further, by our discussion above, NCE techniques do not necessarily suffer from the limitations of lossy encryption when it comes to active attacks. However, we have yet to describe how to handle decryption queries in the security proof, and, indeed, the simple SO-CPA secure scheme needs to be adjusted in several non-trivial ways in order to obtain our SO-CCA secure scheme.

Our scheme. In our SO-CCA secure scheme, encryption of a (multi-bit) message is performed bitwise, with one ciphertext element per bit. If the plaintext bit is 1, the corresponding ciphertext element X is an element of the language $\mathcal{L}$ associated with a hash proof system (HPS, cf. [14]). If the bit is 0, the ciphertext element is a random element, which will most likely be not in $\mathcal{L}$. Additionally, the ciphertext contains an authentication tag T, whose key K is the HPS key[3] associated to X in case $X \in \mathcal{L}$ (computed with the help of the witness), and a random key is taken in case $X \notin \mathcal{L}$. Decryption checks if the authentication tag T is verified correctly by the HPS key $\hat{K}$ computed from X (by means of the HPS secret key), which is the case iff $X \in \mathcal{L}$, i.e., 1 was encrypted. This approach is somewhat similar to the original Cramer-Shoup cryptosystem ([13, 14]), only that the HPS keys are used for authentication and not to directly pad a message.

[2] NCE talks about openings in which *secret keys*, as opposed to encryption randomness, are released. As a consequence, NCE schemes are comparatively inefficient and have severe limitations (see [28]). Our work shows that when "opening" refers to encryption randomness only, then NCE techniques allow for quite practical schemes.

[3] We adopt the notation of [22, 25] to view a HPS as a key encapsulation mechanism, i.e., to call HPS instances "ciphertexts" and HPS proofs "keys."

Opening a ciphertext part as an encryption of 1 means releasing a witness for $X \in \mathcal{L}$. Opening as an encryption of 0 means releasing the randomness used to randomly sample X. The crucial observation now is that 1-encryptions are equivocable: to open a 1-encryption as a 0-encryption, simply claim that X and K were randomly sampled, and provide the corresponding coins. Hence, equivocating all challenge ciphertexts means substituting them by all-one encryptions. This can be done as follows. For any $X \notin \mathcal{L}$, first the corresponding randomly chosen key K is replaced by the corresponding HPS key (which does not change the adversary's view due to statistical properties of the HPS), and then X is replaced by $X \in \mathcal{L}$ (which is indistinguishable to the adversary due to the assumed hardness of $\mathcal{L}$).

In order to have CCA security, it is important that the above changes can be done (and argued) while at the same time being able to answer decryption queries. This is indeed the case in our construction since decryption queries can be answered with the help of the HPS secret key, while the hardness of distinguishing $X \in \mathcal{L}$ from $X \notin \mathcal{L}$ holds even when given the HPS secret key.

The formal security proof uses ideas similar to those of Cramer and Shoup. We stress, however, that our proof is structured quite differently, since additional complications arise due to the fact that each ciphertext contains several X's (one for each plaintext bit), and we have several challenge ciphertexts. Due to this, it will be crucial how exactly and in which order the challenge ciphertexts are substituted by all-one encryptions. Furthermore, we need an authentication tag T that allows to "glue" together in a non-malleable way the L HPS ciphertexts $X_1, \ldots, X_L$, obtained by encrypting an L-bit message, via their corresponding keys $K_1, \ldots, K_L$.

Cross-authentication code. In order to "glue" HPS ciphertexts together, we make use of a new kind of information-theoretic authentication technique, which we call *cross-authentication*. Recall that in standard authentication, the authentication tag is computed from the message and the key, and can then be used to verify the authenticity of the message with the help of the key. In a cross-authentication code (XAC), the authentication tag is instead computed from a *list* $K_1, \ldots, K_L$ of keys (and there is no designated message). It should be possible to verify the correctness of the tag T with *any* single key K_i from the list, and it should be hard for an adversary to forge a tag T' that is accepted by one of the keys, even if the adversary is given all the remaining keys and a correctly computed tag T. To the best of our knowledge, this concept has not been studied before. It is an important ingredient to our construction but might also find other applications as well. We give a formal definition and propose an efficient construction.

Other related work. Dwork et al. [17] study SOA security of commitments, and provide a connection to the Fiat-Shamir methodology. Hemenway et al. [20] were the first to devise SO-CCA secure public-key encryption schemes. Their most efficient schemes have compact ciphertexts of size independent of the number of challenge ciphertexts. Yet, all their constructions follow the lossy encryption paradigm and thus suffer from the drawbacks that are inherent to that approach. Hence, unless the lossy encryption satisfies some additional property, they only prove the weaker IND-SO-CCA security notion, which in particular requires the distribution of the challenge messages to be efficiently conditionally re-samplable. Furthermore, the size of their public and secret keys still depends on the number of challenge ciphertexts. In contrast, our constructions

are comparatively efficient, completely independent of the number of challenge ciphertexts, and do not make assumptions about the distribution of the challenge messages (beyond the usual requirement of being efficiently samplable). Bellare et al. [4] propose a (passively) SOA secure identity-based encryption scheme that is also based on NCE techniques. However, their result does not directly yield a SO-CCA secure public-key encryption scheme, say, by applying the IBE→PKE transformation of Boneh, Canetti, Halevi, and Katz [8]. (In a nutshell, the reason is that [8] use a one-time signature scheme that may lose its guarantees under selective opening attacks.)

2 Preliminaries

Notation. For $n \in \mathbb{N}$, let $[n] := \{1, \ldots, n\}$. Throughout the paper, $k \in \mathbb{N}$ denotes the security parameter. For a finite set X, we denote by $x \leftarrow X$ the process of sampling x uniformly from X. For a probabilistic algorithm A, we denote $y \leftarrow A(x; R)$ the process of running A on input x and with randomness R, and assigning y the result. We let $\mathcal{R}_A$ denote the randomness space of A; we require $\mathcal{R}_A$ to be of the form $\mathcal{R}_A = \{0,1\}^r$. We write $y \leftarrow A(x)$ for $y \leftarrow A(x; R)$ with uniformly chosen $R \in \mathcal{R}_A$, and we write $y_1, \ldots, y_m \leftarrow A(x)$ for $y_1 \leftarrow A(x), \ldots, y_m \leftarrow A(x)$ with fresh randomness in each execution. By $\mathsf{time}_A = \mathsf{time}_A(k) \in \mathbb{N} \cup \{\infty\}$, we denote the supremum of the running time of an algorithm A when running on security parameter k. If time_A is polynomial in k, then A is PPT.

Trapdoor one-way permutations and collision resistant hashing. Informally, a trapdoor one-way permutation should be hard to invert, unless given a trapdoor.

Definition 1 (Trapdoor one-way permutation). *A family of trapdoor one-way permutations $\mathcal{F}$ consists of three PPT algorithms* Gen, Eval *and* Inv *with the following properties.* $\mathsf{Gen}(1^k)$ *outputs the description of a permutation $f : \mathcal{D}_f \to \mathcal{D}_f$ and a trapdoor τ, and* $\mathsf{Eval}(f, x) = f(x)$ *and* $\mathsf{Inv}(\tau, x) = f^{-1}(x)$ *for all $x \in \mathcal{D}_f$. Furthermore, for every PPT algorithm A, the following function is negligible in k:*

$$\Pr\left[A(pk, f(x)) = x \mid (f, \tau) \leftarrow \mathsf{Gen}(1^k), x \leftarrow \mathcal{D}_f\right].$$

Note that we do not distinguish between the function f and its description output by Gen. Furthermore, to simplify notation, we usually leave the algorithms Gen, Eval and Inv implicit and write $(f, f^{-1}) \leftarrow \mathcal{F}$ to denote that a public/secret-key pair is generated using $\mathsf{Gen}(1^k)$, and we write $f(x)$ and $f^{-1}(x)$ to denote that $\mathsf{Eval}(f, x)$ and $\mathsf{Inv}(\tau, x)$ are executed.

Informally, a hash function H is collision resistant if it is infeasible to find two distinct preimages x, x' with $\mathsf{H}(x) = \mathsf{H}(x')$.

Definition 2 (Collision-resistant hash function). *A* collision-resistant hash function *$\mathcal{H}$ with domain $\mathcal{D} = \mathcal{D}_k$ and range $\mathcal{R} = \mathcal{R}_k$ consists of two PPT algorithms* Gen *and* Eval *with the following properties.* $\mathsf{Gen}(1^k)$ *outputs the description of a function* $\mathsf{H} : \mathcal{D} \to \mathcal{R}$ *such that* $\mathsf{Eval}(K, x) = \mathsf{H}(x)$ *for all $x \in \mathcal{D}$. Furthermore, for every PPT algorithm B, the following function is negligible in k:*

$$\mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H},B}(k) := \Pr\left[x \neq x' \wedge \mathsf{H}(x) = \mathsf{H}(x') \mid \mathsf{H} \leftarrow \mathsf{Gen}(1^k), (x, x') \leftarrow B(\mathsf{H})\right]$$

Similarly to above, we do not distinguish between the function H and its description output by Gen and we usually leave the algorithms Gen and Eval implicit and write $\mathsf{H} \leftarrow \mathcal{H}$ to denote that H is generated by Gen.

Encryption schemes and security under selective openings. A public-key encryption scheme consists of three algorithms $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$. Key generation $\mathsf{Gen}(1^k)$ outputs a public key pk and a secret key sk. Encryption $\mathsf{Enc}(pk, M)$ takes a public key pk and a message M, and outputs a ciphertext C. Decryption $\mathsf{Dec}(sk, C)$ takes a secret key sk and a ciphertext C, and outputs a message M. For correctness, we want $\mathsf{Dec}(sk, C) = M$ for all M and all $(pk, sk) \leftarrow \mathsf{Gen}(1^k)$, and with overwhelming probability over $C \leftarrow (pk, M)$.

Following [17, 21, 3, 5, 20], we present a definition for security under selective openings that captures security of an encryption scheme under adaptive attacks. The definition is simulation-based (much like semantic security [19]), and demands that whatever an adversary that sees a vector of ciphertexts deduces can also be deduced by a simulator that does not see any ciphertexts. To model adaptive corruptions, our notion also allows both adversary and simulator to request "openings" of adaptively selected ciphertexts. (Since the simulator does not actually get to see any ciphertexts, it may only ask to see selected components of an initially unknown message vector.)

Definition 3 (SO-CPA, SO-CCA security). *A public-key encryption scheme* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* chosen-plaintext secure under selective openings *(short:* SO-CPA secure*) iff for every polynomially bounded* $n = n(k) > 0$*, every PPT function* R*, and every stateful PPT machine* A *(the adversary), there is a stateful PPT machine* S *(the simulator), such that* $\mathsf{Adv}^{\mathsf{cpa\text{-}so}}_{\mathsf{PKE},A,S,R}$ *is negligible. Here*

$$\mathsf{Adv}^{\mathsf{cpa\text{-}so}}_{\mathsf{PKE},A,S,R}(k) := \Pr\left[\mathsf{Exp}^{\mathsf{cpa\text{-}so\text{-}real}}_{\mathsf{PKE},A,R}(k) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{so\text{-}ideal}}_{S,R}(k) = 1\right],$$

where the experiments $\mathsf{Exp}^{\mathsf{cca\text{-}so\text{-}real}}_{\mathsf{PKE},A,R}(k)$ *and* $\mathsf{Exp}^{\mathsf{so\text{-}ideal}}_{S,R}(k)$ *are defined as follows:*

Experiment $\mathsf{Exp}^{\mathsf{cpa\text{-}so\text{-}real}}_{\mathsf{PKE},A,R}$	**Experiment** $\mathsf{Exp}^{\mathsf{so\text{-}ideal}}_{S,R}$
$(pk, sk) \leftarrow \mathsf{Gen}(1^k)$	$\mathcal{M} \leftarrow S(\mathtt{dist})$
$\mathcal{M} \leftarrow A(\mathtt{dist}, pk)$	$\mathbf{M} := (M^i)_{i\in[n]} \leftarrow \mathcal{M}$
$\mathbf{M} := (M^i)_{i\in[n]} \leftarrow \mathcal{M}$	$I \leftarrow S(\mathtt{select}, (1^{\lvert M^i\rvert})_{i\in[n]})$
$\mathbf{R} := (R^i)_{i\in[n]} \leftarrow (\mathcal{R}_{\mathsf{Enc}})^n$	$out_S \leftarrow S(\mathtt{output}, (M^i)_{i\in I})$
$\mathbf{C} := (C^i)_{i\in[n]} := (\mathsf{Enc}(pk, M^i; R^i))_{i\in[n]}$	return $R(\mathcal{M}, \mathbf{M}, out_S)$
$I \leftarrow A(\mathtt{select}, \mathbf{C})$	
$out_A \leftarrow A(\mathtt{output}, (M^i, R^i)_{i\in I})$	
return $R(\mathcal{M}, \mathbf{M}, out_A)$	

Furthermore, we define

$$\mathsf{Adv}^{\mathsf{cca\text{-}so}}_{\mathsf{PKE},A,S,R}(k) := \Pr\left[\mathsf{Exp}^{\mathsf{cca\text{-}so\text{-}real}}_{\mathsf{PKE},A,R}(k) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{so\text{-}ideal}}_{S,R}(k) = 1\right]$$

for an experiment $\mathsf{Exp}^{\mathsf{cca\text{-}so\text{-}real}}_{\mathsf{PKE},A,R}$ *that is defined like* $\mathsf{Exp}^{\mathsf{cpa\text{-}so\text{-}real}}_{\mathsf{PKE},A,R}$*, but grants the adversary (in all stages of the attack) access to a decryption oracle* $\mathsf{Dec}(sk, \cdot)$*. We require that* A

never queries $\mathsf{Dec}(sk, \cdot)$ *on a challenge ciphertext* C^i*. We say that* PKE *is* chosen-ciphertext secure under selective openings *(short:* SO-CCA secure*) if for all* n*,* R*, and* A *there exists* S *such that* $\mathsf{Adv}^{\mathsf{cca\text{-}so}}_{\mathsf{PKE},A,S,R}(k)$ *is negligible.*

A few remarks about Definition 3 are in place:

- We assume that the distribution $\mathcal{M}$ that A outputs is encoded as a circuit that samples n-tuples of messages according to this distribution. Since A is PPT, this enforces efficient samplability of $\mathcal{M}$. Efficient samplability of $\mathcal{M}$ is a standard and much weaker requirement than the *efficient conditional re-samplability* requirement from the indistinguishability-based selective opening security definitions IND-SO-ENC [3, 5] or IND-SO-CCA2 [20]. We also note that since A chooses $\mathcal{M}$ adaptively (i.e., dependent on pk), SO-CCA security as defined above implies IND-CCA security (see [2] for a convenient formalization).
- We stress that Definition 3 requires the specified security property to hold for *any* (polynomially bounded) n. This is in contrast to the schemes in [20], in which the public key pk depends on n, so once pk is chosen, security is only guaranteed for challenge ciphertexts of bounded length.
- Our notion of "opening of a ciphertext" corresponds to sender corruptions: as an opening, we release plaintext and encryption randomness, but not decryption key. While this clearly poses a significant restriction, it is in a certain sense the best we can hope for without resorting to non-black-box or non-committing encryption techniques (see [21, Section 5]).
- Like [17, 21, 3, 5, 20], we model only one layer of adaptivity. (That is, the adversary may choose only once a subset of ciphertexts to be opened.) More realistic notions would model several stages of adaptive corruption, but would also be substantially more complicated in description and handling. We stress that our SO-CCA secure encryption scheme to be presented does not rely on the assumption of only one corruption stage.
- We allow the length of the messages transmitted by the various senders to vary depending on the randomness of the message distribution $\mathcal{M}$ and the identity of the sender, and we provide this information (i.e. the message lengths $|M^1|, \ldots, |M^n|$) to the simulator. Indeed, we cannot prevent the adversary from always choosing to corrupt the $n/2$ senders that send the longest messages.

Sender-equivocable encryption schemes. We formalize the notion of sender equivocability, which (for CPA security) is similar to non-committing encryption except the adversary is only allowed to corrupt the sender but not the receiver. In addition, we require that to equivocate, the simulator only needs to know the random coins used to generate the simulated ciphertext (and not those for the simulated public key). This latter requirement is needed because unlike the set-up for non-committing encryption, all ciphertexts are generated using the same public key in the selective opening attacks.

Definition 4 (NC-CPA, NC-CCA security). *A public-key encryption scheme* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* sender-equivocable *(short:* NC-CPA secure*) iff there is a stateful PPT machine* S *(the simulator) such that for every stateful PPT machine* A *(the adversary)* $\mathsf{Adv}^{\mathsf{cpa\text{-}nc}}_{\mathsf{PKE},A,S}$ *is negligible. Here*

$$\mathsf{Adv}^{\mathsf{cpa\text{-}nc}}_{\mathsf{PKE},A,S}(k) := \Pr\left[\mathsf{Exp}^{\mathsf{cpa\text{-}nc\text{-}real}}_{\mathsf{PKE},A}(k)=1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{cpa\text{-}nc\text{-}ideal}}_{\mathsf{PKE},A}(k)=1\right],$$

where the experiments $\mathsf{Exp}^{\mathsf{cca\text{-}nc\text{-}real}}_{\mathsf{PKE},A}(k)$ *and* $\mathsf{Exp}^{\mathsf{cpa\text{-}nc\text{-}ideal}}_{\mathsf{PKE},A}(k)$ *are defined as follows:*

Experiment $\mathsf{Exp}^{\mathsf{cpa\text{-}nc\text{-}real}}_{\mathsf{PKE},A}$	**Experiment** $\mathsf{Exp}^{\mathsf{cpa\text{-}nc\text{-}ideal}}_{\mathsf{PKE},A}$
$(pk, sk) \leftarrow \mathsf{Gen}(1^k)$	$(pk, sk) \leftarrow \mathsf{Gen}(1^k)$
$(M, z) \leftarrow A(\mathtt{dist}, pk)$	$(M, z) \leftarrow A(\mathtt{dist}, pk)$
$R \leftarrow \mathcal{R}_{\mathsf{Enc}}$	$C \leftarrow S(\mathtt{sim}, pk, 1^{\lvert M\rvert})$
$C := \mathsf{Enc}(pk, M; R)$	$R \leftarrow S(\mathtt{open}, M)$
return $A(\mathtt{output}, M, C, R, z)$	return $A(\mathtt{output}, M, C, R, z)$

Furthermore, we define

$$\mathsf{Adv}^{\mathsf{cca\text{-}nc}}_{\mathsf{PKE},A,S}(k) := \Pr\left[\mathsf{Exp}^{\mathsf{cca\text{-}nc\text{-}real}}_{\mathsf{PKE},A}(k)=1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{cca\text{-}nc\text{-}ideal}}_{\mathsf{PKE},A}(k)=1\right]$$

for an experiment $\mathsf{Exp}^{\mathsf{cca\text{-}nc\text{-}real}}_{\mathsf{PKE},A}$ *that is defined like* $\mathsf{Exp}^{\mathsf{cpa\text{-}nc\text{-}real}}_{\mathsf{PKE},A}$ *but grants the adversary* A *(in all stages of the attack) access to a decryption oracle* $\mathsf{Dec}(sk, \cdot)$*. We also consider an experiment* $\mathsf{Exp}^{\mathsf{cca\text{-}nc\text{-}ideal}}_{\mathsf{PKE},A}$ *that is defined like* $\mathsf{Exp}^{\mathsf{cpa\text{-}nc\text{-}ideal}}_{\mathsf{PKE},A}$*, but also grants* A *access to* $\mathsf{Dec}(sk, \cdot)$*. In both experiments, we require that* A *never queries* $\mathsf{Dec}(sk, \cdot)$ *on the challenge ciphertext* C*. We say that* PKE *is* chosen-ciphertext secure under selective openings *(short:* NC-CCA secure*) if there exists* S *such that for all* A*,* $\mathsf{Adv}^{\mathsf{cca\text{-}nc}}_{\mathsf{PKE},A,S}(k)$ *is negligible.*

The next lemma says that if an encryption scheme is NC-CPA secure (resp. NC-CCA secure), then it is also SOCPA secure (resp. SOCCA secure). An analogous statement was shown in [10] in the context of non-committing encryption and adaptive corruptions; the main technical difference is that we achieve security amidst selective opening attacks with respect to a single public key.

Lemma 1 (NC-CPA security implies SO-CPA security). *Suppose* PKE *is NC-CPA secure with simulator* S*. Then, for every adversary* A *and every function* R*, there exists an adversary* B *and a simulator* S'*, such that*

$$\left|\mathsf{Adv}^{\mathsf{cpa\text{-}so}}_{\mathsf{PKE},A,S',R}(k)\right| \leq n \left|\mathsf{Adv}^{\mathsf{cpa\text{-}nc}}_{\mathsf{PKE},B,S}(k)\right|. \tag{1}$$

We have $\mathsf{time}_{S'} \approx \mathsf{time}_A + n \cdot \mathsf{time}_S + \mathsf{time}_R$*. Moreover, if* PKE *is NC-CCA secure, then we have that*

$$\left|\mathsf{Adv}^{\mathsf{cca\text{-}so}}_{\mathsf{PKE},A,S',R}(k)\right| \leq n \left|\mathsf{Adv}^{\mathsf{cca\text{-}nc}}_{\mathsf{PKE},B,S}(k)\right|. \tag{2}$$

with the same relation $\mathsf{time}_{S'} \approx \mathsf{time}_A + n \cdot \mathsf{time}_S + \mathsf{time}_R$*.*

The proof idea is very simple: the SOCPA simulator S' generates n equivocable ciphertexts independently, one for each sender and forward these ciphertexts to the adversary A. When A asks for an opening set I, S' relays this set to its own experiment, receives the corresponding messages in I, and opens the ciphertexts in the simulation suitably.

Proof (sketch). We first establish the claim for NC-CPA vs SO-CPA. Here, the simulator S' internally simulates a copy of A and proceeds as follows:

- On input $\texttt{dist}$, run $(pk, sk) \leftarrow \mathsf{Gen}(1^k)$ and output $\mathcal{M} \leftarrow A(\texttt{dist}, pk)$;
- On input $(\texttt{select}, (1^{|M^i|})_{i\in[n]})$, run $\mathbf{C} = (C^i)_{i\in[n]} \leftarrow (S(\texttt{sim}, pk, 1^{|M^i|}))_{i\in[n]}$ and output $I \leftarrow A(\texttt{select}, \mathbf{C})$
- On input $(\texttt{output}, (M^i)_{i\in I})$, compute $R^i \leftarrow S(\texttt{open}, M^i)$ for $i \in I$ and return $out_A \leftarrow A(\texttt{output}, (M^i, R^i)_{i\in I})$

The analysis proceeds via a series of games, where in Game j, $j = 0, 1, \ldots, n$, the first j ciphertexts are generated using $S(\texttt{sim}, pk)$, and the corresponding randomness using $S(\texttt{open}, pk, M^i)$. The last $n-j$ ciphertexts are generated using $\mathsf{Enc}(pk, M^i; R^i)$ with randomness R^i. We claim that the sum (over $j = 1, \ldots, n$) of the distinguishing probabilities between Game $j-1$ and Game j is bounded by $n\mathsf{Adv}^{\mathsf{cpa\text{-}nc}}_{\mathsf{PKE},B,S}(k)$, where B uniformly guesses $j \in [n]$, and internally simulates a copy of A as follows:

- On input $\texttt{dist}, pk$, compute $\mathcal{M} \leftarrow A(\texttt{dist}, pk)$ and $\mathbf{M} = (M^i)_{i\in[n]} \leftarrow \mathcal{M}$, and output $(M, z) = (M_j, \mathbf{M})$.
- On input $(\texttt{output}, M, C, R, z)$, compute

$$C^i = \begin{cases} S(\texttt{sim}, pk, 1^{|M^i|}) & \text{if } i < j \\ C & \text{if } i = j \\ \mathsf{Enc}(pk, M^i; R^i) & \text{if } i > j \end{cases}$$

  compute $I \leftarrow A(\texttt{select}, \mathbf{C})$, $out_A \leftarrow A(\texttt{output}, (M^i, R^i)_{i\in I})$ and output $R(\mathcal{M}, \mathbf{M}, out_A)$.

For NC-CCA vs SO-CCA, the simulator is exactly as above, except it also simulates $\mathsf{Dec}(sk, \cdot)$ which it can since it knows (pk, sk).

3 Warmup: An NC-CPA Secure Scheme

We focus on constructing NC-CPA and NC-CCA secure schemes, which by Lemma 1, are respectively SO-CPA and SO-CCA secure.

Ingredients. As a warmup for our NC-CCA secure scheme, and to explain one of the key ideas, we construct an efficient NC-CPA secure scheme from a slightly enhanced version of trapdoor one-way permutations. Namely, we require that there exist algorithms for sampling the domain $\mathcal{D}_f$, and for explaining an arbitrary $x \in \mathcal{D}_f$ as a result of sampling $\mathcal{D}_f$:

Definition 5 (Efficiently samplable and explainable domain). *A domain $\mathcal{D}_f$ is* efficiently samplable and explainable *iff there exist PPT algorithms* Sample *and* Explain *such that* $\mathsf{Sample}(\mathcal{D}_f; R)$ *is uniformly distributed over $\mathcal{D}_f$ for $R \leftarrow \mathcal{R}_{\mathsf{Sample}}$, and* $\mathsf{Explain}(\mathcal{D}_f, x)$ *outputs R that is uniformly distributed subject to* $\mathsf{Sample}(\mathcal{D}_f; R) = x$ *for any $x \in \mathcal{D}_f$.*

Explainability is a vital property in the construction of non-committing encryption schemes (see Damgård and Nielsen [15]; there, an essentially equivalent property is called "invertible sampling"). We stress that the domain of most "natural" trapdoor one-way permutations satisfies Definition 5.[4] Note that for families of trapdoor one-way permutations, explainability implies that the family is *enhanced* in the sense of Goldreich [18], Appendix C.1.

Hence, let $\mathcal{F}$ be a family of trapdoor one-way permutations $f : \mathcal{D}_f \to \mathcal{D}_f$ with efficiently samplable and explainable domain $\mathcal{D}_f$ (for every $f \in \mathcal{F}$), and hard-core predicate $h : \mathcal{D}_f \to \{0,1\}$. For $(f, f^{-1}) \leftarrow \mathcal{F}$ and $\ell = \ell(k)$, define

$$\mathsf{BM}_{f,\ell}(x) := (h(x), h(f(x)), \ldots, h(f^{\ell-1}(x))) \in \{0,1\}^\ell.$$

It is well-known that BM is pseudorandom, even given $f^\ell(x)$. Formally:

Theorem 1 (Blum and Micali [6]). *Let $\mathcal{F}$ a family of trapdoor one-way permutations $f : \mathcal{D}_f \to \mathcal{D}_f$ with hard-core predicate $h : \mathcal{D}_f \to \{0,1\}$. Then, for every PPT distinguisher D and every polynomially bounded $\ell = \ell(k)$, the function*

$$\mathsf{Adv}^{\mathsf{prg}}_{\mathcal{F},\ell,D}(k) := \Pr\left[D(f^\ell(x), \mathsf{BM}_{f,\ell}(x)) = 1\right] - \Pr\left[D(x, K) = 1\right]$$

is negligible in k, where $(f, f^{-1}) \leftarrow \mathcal{F}$, $x \leftarrow \mathcal{D}_f$, and $K \leftarrow \{0,1\}^k$.

The scheme. For $\mathcal{F}$ as above and a message space of $\{0,1\}$, our NC-CPA secure encryption scheme $\mathsf{NCCPA} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is defined as:

$\mathsf{Gen}(1^k)$**.** Sample $(f, f^{-1}) \leftarrow \mathcal{F}$, and return $(pk, sk) = (f, f^{-1})$.

$\mathsf{Enc}(pk, M; R)$**.** Parse $pk = f$, $M \in \{0,1\}$, and $R = (R^x, K_0) \in \mathcal{R}_{\mathsf{Sample}} \times \{0,1\}^k$. Set $x \leftarrow \mathsf{Sample}(\mathcal{D}_f; R^x)$ and return

$$C := (y, K) := \begin{cases} (f^k(x), \mathsf{BM}_{f,k}(x)) & \text{if } M = 1 \\ (x, K_0) & \text{if } M = 0. \end{cases}$$

$\mathsf{Dec}(sk, C)$**.** Parse $sk = f^{-1}$ and $C = (y, K)$. Return $M = 1$ if $\mathsf{BM}_{f,k}(f^{-k}(y)) = K$, and $M = 0$ else.[5]

Note that 1-encryptions are always correctly decrypted, while 0-encryptions are wrongly decrypted to 1 with probability 2^{-k}. Furthermore, larger messages can be encrypted by concatenating ciphertexts. (This does not affect NCCPA's NC-CPA security.)

Equivocable ciphertexts and sketch of security proof. The key to proving NC-CPA security is that 1-encryptions are equivocable. More concretely, the NC-CPA simulator S proceeds as follows:

[4] Damgård and Nielsen [15] show that any dense subset $\mathcal{D}_f$ of an efficient samplable domain is both efficiently samplable and explainable as long as $\mathcal{D}_f$ admits an efficient membership test. For the trapdoor permutations based on RSA, the public index is a RSA modulus N and the domain $\mathbb{Z}_N^*$ clearly satisfies these properties. For Rabin's trapdoor permutations based on modular squaring, the public index is a Blum integer N and we need to modify the domain to be the group of signed quadratic residues in $\mathbb{Z}_N^*$.

[5] Note that $\mathsf{BM}_{f,k}(f^{-k}(y))$ can be computed from f^{-1} and y alone.

- On input $(\mathtt{sim}, pk)$ where pk is the public key f, it returns a random 1-encryption given by $(y, K) = (f^k(x), \mathsf{BM}_{f,k}(x))$ with randomness R;
- On input $(\mathtt{open}, M)$ for $M \in \{0, 1\}$, it returns $(\mathsf{Explain}(\mathcal{D}_f, y), K)$ if $M = 0$ and R if $M = 1$.

A straightforward hybrid argument to BM's pseudorandomness shows that this simulation achieves a computationally indistinguishable view for A in real experiment and ideal simulation. We obtain:

Theorem 2 (NCCPA is NC-CPA secure). *For every adversary A and every function R, there exists a simulator S, and a distinguisher D, such that*

$$\left|\mathsf{Adv}^{\mathsf{cpa\text{-}so}}_{\mathsf{SOCCA},A,S,R}(k)\right| \leq n \left|\mathsf{Adv}^{\mathsf{prg}}_{\mathcal{F},k,D}(k)\right|. \tag{3}$$

We have $\mathsf{time}_S \approx \mathsf{time}_A$ *and* $\mathsf{time}_D \approx \mathsf{time}_A + \mathsf{time}_R$.

We omit a more detailed proof, since the proof of Theorem 2 is similar to, but conceptually simpler than the upcoming proof for our NC-CCA secure scheme.

Relation to non-committing encryption. We point out that NCCPA can be seen as a variant of non-committing encryption schemes in [10, 24]. Compared with these schemes, our scheme is more efficient and conceptually simpler. It also allows for an unbounded usage, since we only need to provide encryption random coins (but not secret keys) upon an opening. As such, NCCPA serves as a useful tool to explain how we use equivocable ciphertexts to prove security under selective openings. The main technical difficulties lie in designing and analyzing a chosen-ciphertext secure scheme. This will turn out to be a delicate task that requires some more preparation.

4 Hash Proof Systems with Explainable Domains

We recall the notions of a subset membership problem and of an (extended) hash proof system, as introduced in Cramer and Shoup [14]. In our definitions, we require all properties to hold *perfectly*; this can be relaxed by allowing a negligibly small error probability (which includes that sampling algorithms may produce near-uniform output).

Definition 6 (Subset membership problem). *A* subset membership problem SMP *consists of the following PPT algorithms.*

System parameter generation. $\mathsf{SysGen}(1^k)$ *outputs system parameters ρ that defines a set $\mathcal{X}_\rho$ of ciphertexts and a language $\mathcal{L}_\rho \subseteq \mathcal{X}_\rho$. $\mathcal{X}_\rho$ is required to be efficiently recognizable (given ρ).*

Sampling from $\mathcal{L}_\rho$. $\mathsf{SampleL}(\mathcal{L}_\rho; W)$ *uniformly samples $X \leftarrow \mathcal{L}_\rho$ using randomness W.*

A subset membership problem SMP *is called* hard *iff $\mathcal{X}_\rho$ and $\mathcal{L}_\rho$ are computationally indistinguishable. Concretely, for every PPT distinguisher D, the following function is negligible:*

$$\mathsf{Adv}^{\mathsf{sm}}_{\mathsf{EHPS},D}(k) := \Pr\left[D(X) = 1 \mid X \leftarrow \mathcal{X}_\rho \setminus \mathcal{L}_\rho\right] - \Pr\left[D(X) = 1 \mid X \leftarrow \mathcal{L}_\rho\right]$$

where in both probabilities $\rho \leftarrow \mathsf{SysGen}(1^k)$.

Definition 7 (EHPS). *An* extended hash proof system *(short:* EHPS*)* EHPS *for a subset membership problem* SMP *associates with each* $\rho \leftarrow \mathsf{SysGen}(1^k)$ *an efficiently recognizable set of keys* $\mathcal{K}_\rho$ *and an efficiently recognizable set of tags* $\mathcal{T}_\rho$*, and consists of the following PPT algorithms:*

Individual key generation. $\mathsf{HashGen}(\rho)$ *outputs a public key* hpk *and a secret key* hsk*. We assume that* hpk *and* hsk *both contain* ρ*.*

Secret evaluation. $\mathsf{SEval}(hsk, X, t)$ *computes a key* $K \in \mathcal{K}_\rho$*. We also write* $K = hsk(X, t)$*.*

Public evaluation (with witness). $\mathsf{PEval}(hpk, X, W, t)$ *computes a key* $K \in \mathcal{K}_\rho$*. We require correctness in the sense of* $\mathsf{PEval}(hpk, X, W, t) = \mathsf{SEval}(hsk, X, t)$ *for all* $\rho \leftarrow \mathsf{SysGen}(1^k)$*,* $(hpk, hsk) \leftarrow \mathsf{HashGen}(\rho)$*,* $X \leftarrow \mathsf{SampleL}(\mathcal{L}_\rho; W)$*, and all* $t \in \mathcal{T}_\rho$*.*

By definition, in an EHPS the public key hpk uniquely determines the action of SEval for ciphertexts $X \in \mathcal{L}_\rho$. An EHPS typically becomes interesting/useful when on the other hand the action of SEval for ciphtertexts $X \in \mathcal{X}_\rho \setminus \mathcal{L}_\rho$ is "very undetermined". We capture this as follows.

Definition 8 (2-universal). *An EHPS (for* SMP*) is* 2-universal *iff for all possible* $\rho \leftarrow \mathsf{SysGen}(1^k)$*, all* hpk *in the range of* $\mathsf{HashGen}(\rho)$*, and all distinct* $(X_1, t_1), (X_2, t_2)$ *in* $(\mathcal{X}_\rho \setminus \mathcal{L}_\rho) \times \mathcal{T}$*,*

$$\Pr\big[hsk(X_2, t_2) = K_2 \,\big|\, hsk(X_1, t_1) = K_1\big] = \frac{1}{|\mathcal{K}_\rho|},$$

where the probability is over possible hsk *with* $(hpk, hsk) \leftarrow \mathsf{HashGen}(\rho)$*.*

In addition to the above (standard) properties, we will also need the following non-standard requirements.

Definition 9 (Sparseness of the language). *An subset membership problem* SMP *has a* sparse language *if for* $\rho \leftarrow \mathsf{SysGen}(1^k)$ *and* $X \leftarrow \mathcal{X}_\rho$*, the probability that* $X \in \mathcal{L}_\rho$ *is negligible.*

Definition 10 (Explainable ciphertexts and keys). *We say that a subset membership problem* SMP *has* explainable ciphertexts *if the set* $\mathcal{X}_\rho$ *is efficiently samplable and explainable in the sense of Definition 5. Similarly, an extended hash proof system* EHPS *has* explainable keys *if the set* $\mathcal{K}_\rho$ *is efficiently samplable and explainable.*

We point out that explainable keys can actually be assumed without loss of generality, because $\mathcal{K}_\rho$ can always be efficiently mapped into $\mathcal{K}'_\rho = \{0,1\}^m$ by means of a suitable (almost) balanced function, such that uniform distribution in $\mathcal{K}_\rho$ induces (almost) uniform distribution in $\mathcal{K}'_\rho$, and where m is linear in $\log(|\mathcal{K}_\rho|)$. The requirement on the ciphertexts to be explainable, on the other hand, is a real restriction on the SMP; nevertheless, several suitable SMPs do satisfy this requirement and have a sparse language, as we will outline next.

Examples of suitable SMPs. The DDH-based SMPs from Cramer and Shoup [14] satisfy all our requirements, assuming that the platform group $\mathbb{G}$ is efficiently samplable and explainable in the sense of Definition 5. One popular such group in which DDH is assumed to be hard is the unique q-order subgroup of $\mathbb{Z}_p^*$, where $p = 2q + 1$ is a safe prime. Another one is the elliptic curve $\mathbb{G}_1$ from [7, Section 5.1]. The Paillier-based SMP from [14] fulfils our requirements as well. Finally, the SMP from [14] based on quadratic residuosity satisfies all our requirements *except* for a sparse language (Definition 9). However, the SMP that consists of, say, k parallel copies of the QR SMP from [14] (and where the EHPS key is the product of the individual keys) *has* a sparse language and satisfies our remaining requirements.

5 Cross-Authentication Codes

We introduce here a new information-theoretic authentication technique, which will play an important role in our construction of a SO-CCA-secure encryption scheme. However, the technique may also be useful in other contexts. Cross-authentication, as we call our technique, allows to compute an authentication tag T for a *list* $K_1, \ldots, K_L$ of keys, with the following two properties. The tag T can be verified by any single key K_i from the list, and without knowledge of K_i it is information-theoretically hard to forge a tag T' that is correctly verified by K_i, even when given a correctly computed tag T and all the other keys $K_{\neq i} = (K_j)_{j \neq i}$.

Below is the formal definition followed by an efficient example construction.

Definition 11 (L-Cross-authentication code). *For $L \in \mathbb{N}$, an* L-cross-authentication code *(short: L-XAC)* XAC *consists of a key space $\mathcal{XK}$ and a tag space $\mathcal{XT}$ and of three PPT algorithms* XGen, XAuth *and* XVer. $\mathsf{XGen}(1^k)$ *produces a uniformly random key $K \in \mathcal{XK}$,* $\mathsf{XAuth}(K_1, \ldots, K_L)$ *outputs a tag $T \in \mathcal{XT}$, and* $\mathsf{XVer}(K, i, T)$ *outputs a decision bit. The following is required:*

Correctness. *For all $i \in [L]$, the probability*

$$\mathsf{fail}_{\mathsf{XAC}}(k) := \Pr\left[\mathsf{XVer}(K_i, i, \mathsf{XAuth}(K_1, \ldots, K_L)) \neq 1\right],$$

is negligible, where $K_1, \ldots, K_L \leftarrow \mathsf{XGen}(1^k)$ in the probability.

Security against impersonation and substitution attacks. $\mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k)$ *and* $\mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k)$ *as defined below are both negligible:*

$$\mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k) := \max_{i,T'} \Pr\left[\mathsf{XVer}(K, i, T') = 1 \mid K \leftarrow \mathsf{XGen}(1^k)\right]$$

where the max *is over all $i \in [L]$ and $T' \in \mathcal{XT}$, and*

$$\mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k) := \max_{i,K_{\neq i},F} \Pr\left[\begin{array}{l|l} T' \neq T \wedge & K_i \leftarrow \mathsf{XGen}(1^k), \\ \mathsf{XVer}(K_i, i, T') = 1 & T := \mathsf{XAuth}(K_1, \ldots, K_L), \\ & T' \leftarrow F(T) \end{array}\right].$$

where the max *is over all $i \in [L]$, all $K_{\neq i} = (K_j)_{j \neq i} \in \mathcal{XK}^{L-1}$ and all (possibly randomized) functions $F : \mathcal{XT} \to \mathcal{XT}$.*

Note that by taking $\mathcal{R}_{\mathsf{XGen}}$ as key space, instead of $\mathcal{XK}$, we may without loss of generality assume that $\mathcal{XK}$ is of the form $\mathcal{XK} = \{0,1\}^r$ (and XGen simply outputs its randomness).

Example of a L-XAC. Let $\mathbb{F}$ be a finite field of size q, where q depends on k (e.g. $q = 2^k$). Set $\mathcal{XK} = \mathbb{F}^2$ and $\mathcal{XT} = \mathbb{F}^L \cup \{\perp\}$, and let XGen produce a random key in $\mathcal{XK} = \mathbb{F}^2$. For $K_1 = (a_1, b_1), \ldots, K_L = (a_L, b_L) \in \mathcal{XK}$, the authentication tag $T = \mathsf{XAuth}(K_1, \ldots, K_L)$ is given by the unique vector $T = (T_0, \ldots, T_{L-1}) \in \mathbb{F}^L$ such that $p_T(a_i) = b_i$ for $i = 1, \ldots, L$, where $p_T(x) = T_0 + T_1 x + \cdots + T_{L-1} x^{L-1} \in \mathbb{F}[x]$. T can be computed efficiently by solving the linear equation system $\mathbf{A}T = B$, where $\mathbf{A} \in \mathbb{F}^{L \times L}$ is the Vandermonde matrix whose i-th row is given by $1, a_i, a_i^2 \ldots, a_i^{L-1}$, and where $B \in \mathbb{F}^L$ is the column vector with entries $b_1, \ldots, b_L$. If $\mathbf{A}T = B$ admits more than one or no solution, then T is set to $\perp$ instead. For any $T \in \mathcal{XT}$, $K = (a, b) \in \mathcal{XK}$ and $i \in [L]$, the verification $\mathsf{XVer}(K, i, T)$ outputs 1 if and only if $T \neq \perp$ and $p_T(a) = b$.

Lemma 2. *The above L-XAC* XAC *satisfies:*

$$\mathsf{fail}_{\mathsf{XAC}}(k) \leq \frac{L(L-1)}{2q}\,, \quad \mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k) \leq \frac{1}{q} \quad \textit{and} \quad \mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k) \leq 2 \cdot \frac{L-1}{q}\,.$$

Proof. Correctness: By construction, $\mathsf{XVer}(K_i, i, \mathsf{XAuth}(K_1, \ldots, K_L)) = 1$ except if the Vandermonde matrix $\mathbf{A}$ is singular. The Vandermonde determinant $\det(\mathbf{A})$ is well known to be non-zero unless $a_i = a_j$ for some $i \neq j$, where the latter happens with probability at most $\frac{1}{2}L(L-1)/|\mathbb{F}|$.

Security against impersonation attack: Consider an arbitrary but fixed $T' \in \mathcal{XT}$. If $T' = \perp$ then $\mathsf{XVer}(K, i, T) = 0$ for any choice of K and i. Else, if $T' \in \mathbb{F}^L$, then the probability (over the uniformly random choice of b) that $p_{T'}(a) = b$ is $1/|\mathbb{F}|$.

Security against substitution attack: Consider an arbitrary $i \in [L]$. For concreteness, but without loss of generality, we may assume $i = L$. We fix arbitrary values for $K_1 = (a_1, b_1), \ldots, K_{L-1} = (a_{L-1}, b_{L-1})$. We may assume those a_i's to be pairwise distinct, since otherwise T will be $\perp$ for any choice of K_L and then the probability of finding T' that is accepted by K_L is upper bounded by $\mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k)$. We first slightly modify the computation of T as follows. Instead of setting T to $\perp$ as soon as $\det(\mathbf{A}) = 0$, we distinguish between the case where $\mathbf{A}T = B$ has no solution and where it has multiple solutions for T. In the former case, T is still set to $\perp$, but in the latter, T is chosen uniformly at random from all the solutions. Note that this modification makes the computation of T randomized (at least in general), but the definition of $\mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k)$ still makes sense. This modification changes the value of $\mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k)$ by at most $\varepsilon_{\text{multi}} = \Pr[\mathbf{A}T = B \text{ has multiple solutions}]$, where the probability is over the choice of K_L.

In the following argument, we consider the above modified version of XAC. The probability $\mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k)$ is upper bounded by the corresponding probability conditioned on $T \neq \perp$ plus the probability that $T = \perp$. Since the latter probability equals $\varepsilon_{\text{no}} = \Pr[\mathbf{A}T = B \text{ has no solution}]$, we can focus on the former while book-keeping the "error" accumulated so far: $\varepsilon_{\text{multi}} + \varepsilon_{\text{no}} = \Pr[\det(\mathbf{A}) = 0] = \Pr[a_L \in \{a_1, \ldots, a_{L-1}\}] = (L-1)/|\mathbb{F}|$. In the following argument, we consider an arbitrary $T \neq \perp$, and we consider the corresponding (conditional) probability distribution of K_L. It holds that

$K_L = (a_L, b_L)$ is uniformly distributed in $\mathbb{F}^2$ subject to $p_T(a_L) = b_L$. This in particular implies that a_L on its own is uniformly distributed. Consider now an arbitrary choice for $T' \in \mathcal{XT}$ (computed from $K_1, \ldots, K_{L-1}$ and T). T' is required to be different from T, and we may assume that $T' \neq \perp$, since otherwise $\mathsf{XVer}(K, L, T) = 0$ holds with certainty. By linearity, $p_{T'}(a_L) = b_L$ holds exactly if $p_{T'-T}(a_L) = 0$. However, by the Schwartz-Zippel lemma, $p_{T'-T}(a_L) = 0$ holds with probability at most $\deg(p_{T'-T}(x))/|\mathbb{F}| \leq (L-1)/|\mathbb{F}|$ for a uniformly random $a_L \in \mathbb{F}$. Taking into account $\varepsilon_{\text{multi}}$ and ε_{no} from further up, this proves the claim.

6 Our NC-CCA Secure Scheme

Ingredients. For our encryption scheme with message space $\{0,1\}^L$, we need the following.

1. A hard subset membership problem SMP with sparse language $\mathcal{L}_\rho$ and explainable ciphertexts $\mathcal{X}_\rho$.
2. A 2-universal extended hash proof system EHPS for SMP with tags $\mathcal{T}_\rho$ and explainable keys $\mathcal{K}_\rho$.
3. A collision-resistant hash function $\mathcal{H}$ with domain $(\mathcal{X}_\rho)^L$ and range $\mathcal{T}_\rho$.
4. An L-cross-authentication code XAC with key space $\mathcal{XK} = \mathcal{K}_\rho$ and tag space $\mathcal{XT}$.

From the remarks after Definition 10 and 11 it follows that the efficient samplability and explainability of $\mathcal{K}_\rho$ and the requirement on $\mathcal{XK}$ to coincide with $\mathcal{K}_\rho$ pose no real restriction. In fact, all of these ingredients exist under standard number-theoretic assumptions such as decisional Diffie-Hellman (DDH), decisional composite residuosity (DCR), and quadratic residuosity (QR).

The scheme. We define our encryption scheme $\mathsf{NCCCA} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ as follows:

$\mathsf{Gen}(1^k)$**.** Run $\rho \leftarrow \mathsf{SysGen}(1^k)$, $(hpk, hsk) \leftarrow \mathsf{HashGen}(\rho)$ and $\mathsf{H} \leftarrow \mathcal{H}$. Return public key $pk = (hpk, \mathsf{H})$ and secret key $sk = (hsk, \mathsf{H})$.

$\mathsf{Enc}(pk, M; R)$**.** Parse $pk = (hpk, \mathsf{H})$, $M = (M_1, \ldots, M_L) \in \{0,1\}^L$, and $R = (W_i, R_i^X, R_i^K)_{i \in [L]} \in (\mathcal{R}_{\mathsf{SampleL}} \times \mathcal{R}_{\mathsf{Sample}} \times \mathcal{R}_{\mathsf{Sample}})^L$. For $i \in [L]$, set

$$X_i := \begin{cases} \mathsf{Sample}(\mathcal{X}_\rho; R_i^X) \in \mathcal{X}_\rho & \text{if } M_i = 0 \\ \mathsf{SampleL}(\mathcal{L}_\rho; W_i) \in \mathcal{L}_\rho & \text{if } M_i = 1 \end{cases},$$

and compute $t := \mathsf{H}(X_1, \ldots, X_L)$. Then, for $i \in [L]$, set the keys

$$K_i := \begin{cases} \mathsf{Sample}(\mathcal{K}_\rho; R_i^K) & \text{if } M_i = 0 \\ \mathsf{PEval}(hpk, X_i, W_i, t) & \text{if } M_i = 1 \end{cases}$$

and compute the tag $T := \mathsf{XAuth}(K_1, \ldots, K_L)$. Return $C = (X_1, \ldots, X_L, T)$.

$\mathsf{Dec}(sk, C)$**.** Parse $sk = (hsk, \mathsf{H})$ and $C = (X_1, \ldots, X_L, T) \in \mathcal{X}_\rho^L \times \mathcal{XT}$. Set $t := \mathsf{H}(X_1, \ldots, X_L)$. For $i \in [L]$, let $\overline{K}_i := hsk(X_i, t)$, and $M_i := \mathsf{XVer}(\overline{K}_i, i, T)$. Return $M := (M_1, \ldots, M_L)$.

Lemma 3 (Correctness of NCCCA). *For any pk in the range of* Gen*, any M, and any $C \leftarrow \mathsf{Enc}(pk, M)$, we have $\mathsf{Dec}(sk, C) = M$ except with probability at most $L \cdot \max\{\mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k), \mathsf{fail}_{\mathsf{XAC}}(k)\}$.*

Proof. If $M_i = 1$, then $\overline{K}_i = hsk(X_i, t) = \mathsf{PEval}(hpk, X_i, W_i, t) = K_i$ by completeness of EHPS, and so $\mathsf{XVer}(\overline{K}_i, i, T) = 1$ except with probability $\mathsf{fail}_{\mathsf{XAC}}(k)$ by correctness of XAC. On the other hand, for $M_i = 0$, EHPS's universality implies that $\overline{K}_i = hsk(X^f, t)$ is uniformly random, even given pk, C, and M. Hence, the probability that $\mathsf{XVer}(\overline{K}_i, i, T) = 1$ is at most $\mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(\mathrm{k})$. The statement follows by a union bound over $i \in [L]$.

Equivocable ciphertexts. As with our earlier scheme NCCPA, NCCCA has the property that 1-encryptions are equivocable. Specifically, we can construct a NC-CCA simulator S that proceeds as follows:

- On input $(\mathtt{sim}, pk, 1^L)$ where pk is the public key (hpk, H), it generates an *equivocable* ciphertext of the form

$$C = (X'_1, \ldots, X'_L, T) = (\mathsf{SampleL}(\mathcal{L}_\rho; W'_1), \ldots, \mathsf{SampleL}(\mathcal{L}_\rho; W'_L), T) \quad (4)$$

  for uniformly chosen $W'_i \in \mathcal{R}_{\mathsf{SampleL}}$ and $T := \mathsf{XAuth}(K'_1, \ldots, K'_L)$ with $K'_i := \mathsf{PEval}(hpk, X_i, W_i, t)$.
- On input $(\mathtt{open}, M)$ for an arbitrary $M \in \{0,1\}^L$, such a C can be explained as an encryption of M by releasing $R = \left(W_i, R^X_i, R^K_i\right)_{i \in [L]}$ with $W_i = W'_i$ if $M_i = 1$, and $(R^X_i, R^K_i) = (\mathsf{Explain}(\mathcal{X}_\rho, X'_i), \mathsf{Explain}(\mathcal{K}_\rho, K'_i))$ if $M_i = 0$.

Our security proof shows that equivocated ciphertexts and their openings are indistinguishable from authentic ones, even given a decryption oracle.

7 Security Analysis

Theorem 3 (NCCCA is NC-CCA secure). *There exists a simulator S such that for every adversary A there exists a subset membership distinguisher D and an adversary B on* H*'s collision resistance property such that $\mathsf{time}_D, \mathsf{time}_B \approx \mathsf{time}_A$ and*

$$\left|\mathsf{Adv}^{\mathsf{cca\text{-}nc}}_{\mathsf{NCCCA},A,S}(k)\right| \leq L \cdot \left|\mathsf{Adv}^{\mathsf{sm}}_{\mathsf{EHPS},D}(k)\right| + 2L^2 q \cdot \mathsf{Adv}^{\mathsf{xac}}_{\mathsf{XAC}}(k) + \mathsf{Adv}^{\mathsf{cr}}_{\mathsf{H},B}(k) + \frac{L(L-1)}{|\mathcal{L}_\rho|}, \quad (5)$$

where $\mathsf{Adv}^{\mathsf{xac}}_{\mathsf{XAC}}(k) = \max\left\{\mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k), \mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k)\right\}$ and q is an upper bound on the number of decryption queries A performs.

Before going into the formal proof below, we briefly give a high-level description of the reasoning. The goal is to replace the challenge ciphertext by an equivocable ciphertext. We replace the challenge ciphertexts as follows, one-by-one for every X^*_m (that is not already in $\mathcal{L}_\rho$) within every challenge ciphertext C^*. First, instead of choosing the corresponding key K^*_m at random whenever $M_m = 0$, K^*_m is always computed as HPS key $K^*_m = hsk(X^*_m, t^*)$. Next, $X^*_m \notin \mathcal{L}_\rho$ is replaced by $X^*_m \in \mathcal{L}_\rho$, yielding an equivocable ciphertext.

We now briefly argue why these modifications do not (significantly) alter the adversary A's view. In order to argue that the modification to the choice of K_m^* does not change A's view, it is crucial that A has no information on the HPS secret key hsk beyond the public key hpk. In order to guarantee this, we first slightly modify the decryption procedure Dec used to answer the decryption queries so that Dec does not make any use of hsk: rather than verifying the XAC tag T_i, the decrypted message bit M_i is directly set to 0 whenever $X_i \notin \mathcal{L}_\rho$. By universality of the hash proof system and the security of XAC against impersonation attacks, it follows that this modification does not significantly change A's view. Note that with this modified decryption procedure, the resulting game is not efficient anymore, but this fine for arguing that choosing K_m^* as HPS key instead of random does not change A's view, since this is an information-theoretic argument. However, this step would be a problem for justifying the switch from $X_m^* \notin \mathcal{L}_\rho$ to $X_m^* \in \mathcal{L}_\rho$. Therefore, before doing the latter switch, the modified decryption procedure is replaced again by the original procedure Dec. Again, this change to the decryption procedure can be argued to have little effect on A's view by universality of the hash proof system and security of XAC. However, in this case things are slightly more subtle because if $X_i = X_m^*$ and $t = t^*$, then A now knows an XAC tag that is verified by the HPS key $K_i = hsk(X_i, t)$, namely T^*. But if indeed $t = t^*$ then the collision resistance of H ensures that A has to submit a *different* XAC tag. Hence security against substitution attacks of XAC ensures that the tag will be rejected. Thus both decryption processes decrypt to the same message bit and are hence indistinguishable.

Proof. We proceed in a series of games. Generally, we will denote the output of Game i by out_i.

Game -2 is the original real experiment $\mathsf{Exp}^{\mathsf{cca\text{-}nc\text{-}real}}_{\mathsf{NCCCA},A}$. By definition,

$$\Pr[out_{-2} = 1] = \mathsf{Exp}^{\mathsf{cca\text{-}nc\text{-}real}}_{\mathsf{NCCCA},A}(k). \tag{6}$$

Let $M^* = (M_1^*, \ldots, M_\ell^*)$ denote the message chosen by A; C^* be the challenge ciphertext handed to A; and C^j be A's j-th decryption query. Write $C^* = (X_1^*, \ldots, X_L^*, T^*)$, $C^j = (X_1^j, \ldots, X_L^j, T^j)$, and similarly for the variables t^*, K_i^j, etc. Without loss of generality, we assume that A always makes $q = q(k)$ decryption queries.

In **Game** -1, we abort the experiment (with output 1) as soon as $X_i^* = X_{i'}^*$ for some distinct $i, i' \in [L]$. A counting argument and a union bound show

$$|\Pr[out_{-1} = 1] - \Pr[out_{-2} = 1]| \leq \frac{L(L-1)}{|\mathcal{L}_\rho|}. \tag{7}$$

In **Game** 0, we abort the experiment (with output 1) as soon as A submits a decryption query C^j with

$$t^j = \mathsf{H}(X_1^j, \ldots, X_L^j) = (X_1^*, \ldots, X_L^*) = t^*$$

for some ℓ. A straightforward reduction shows that

$$\Pr[out_0 = 1] - \Pr[out_{-1} = 1] = \mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H},B}(k) \tag{8}$$

for a suitable B that simulates Game 0.[6]

[6] If H is only *target* collision resistant, a reduction with a multiplicative loss of q can be conducted.

From **Game** 0 **up to Game** L, we will stepwise replace the challenge ciphertext C^* with an equivocable ciphertext of the form (4). Specifically, **Game** m with $0 \leq m \leq L$ coincides with **Game** 0 except that X_i^* and K_i^* with $i \leq m$ are computed as $X_i^* := \mathsf{SampleL}(\mathcal{L}_\rho; W_i^*) \in \mathcal{L}_\rho$ and $K_i^* := \mathsf{PEval}(hpk, X_i^*, W_i^*, t)$, no matter what M_i^* is, and X_i^* is opened suitably to M_i^* as explained at the end of Section 6. Looking ahead, we point out that the final **Game** L, in which C^* is equivocable, is identical to the ideal experiment $\mathsf{Exp}^{\mathsf{cca\text{-}nc\text{-}ideal}}_{\mathsf{NCCCA},S}$ for the simulator S described earlier. We now show indistinguishability between **Games** m **and** $m+1$ for any $0 \leq m \leq L-1$. We do this in several steps.

Game $m.1$ is identical to **Game** m above.

In **Game** $m.2$, we slightly modify the decryption oracle. Recall that from each EHPS ciphertext X_i of a decryption query C, a key $\overline{K}_i = hsk(X_i, t)$ is computed and $M_i := \mathsf{XVer}(\overline{K}_i, i, T)$ is returned. We change this to $M_i := 0$ iff X_i is inconsistent in the sense of $X_i \notin \mathcal{L}_\rho$. (Note that this makes Game $m.2$ inefficient.)

Let $\mathsf{bad}_{m.i.1}$ denote the event that in Game $m.1$, there is a EHPS ciphertext X_i in some C^j that is inconsistent in the sense $X_i \notin \mathcal{L}_\rho$, but $\mathsf{XVer}(\overline{K}_i, i, T) = 1$. Let $\mathsf{bad}_{m.2}$ be the corresponding event in Game $m.2$. By construction, it holds that Game $m.1$ and Game $m.2$ are identical as long as the respective events $\mathsf{bad}_{m.1}$ and $\mathsf{bad}_{m.2}$ do not occur, and $\Pr[\mathsf{bad}_{m.1}] = \Pr[\mathsf{bad}_{m.2}]$. We postpone the proof of the following claim:

Lemma 4. $\Pr[\mathsf{bad}_{m.2}] \leq Lq \cdot \mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k)$.

It follows that

$$|\Pr[out_{m.2} = 1] - \Pr[out_{m.1} = 1]| \leq \Pr[\mathsf{bad}_{m.2}] \leq Lq \cdot \mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k)\,. \qquad (9)$$

Note that the adversary's view in Game $m.2$ depends only on hpk. Namely, while the experiment uses hsk to decrypt consistent EHPS ciphertexts efficiently, by completeness of EHPS, this does not release any information on hsk beyond hpk.

In **Game** $m.3$, instead of choosing $K_m^* \in \mathcal{K}_\rho$ uniformly, using Sample, if $M_m^* = 0$, we compute

$$K_m^* := hsk(X_m^*, t^*)$$

as in a hypothetical decryption of C^*. (Later, if C^* is to be opened, K_m^* is explained as being randomly through $\mathsf{Sample}(\mathcal{K}_\rho)$, using coins $\mathsf{Explain}(\mathcal{K}_\rho, K_m^*)$.) Since the only information about hsk *beyond* hpk is released while computing K_m^*, the universality of EHPS guarantees that K_m^* looks uniform. Concretely,

$$\Pr[out_{m.3} = 1] = \Pr[out_{m.2} = 1]\,. \qquad (10)$$

In **Game** $m.4$, we reverse the changes from Game $m.2$. That is, decryption does not set $M_i := 1$ iff $X_i \in \mathcal{L}_\rho$, but again computes $M_i := \mathsf{XVer}(\overline{K}_i, i, T)$. Note that this makes Game $m.4$ efficient again.

Let $\mathsf{bad}_{m.3}$ denote the event that in Game $m.3$, there is a EHPS ciphertext X_i in some C^j that is inconsistent in the sense $X_i \notin \mathcal{L}_\rho$, but $\mathsf{XVer}(\overline{K}_i, i, T) = 1$. Let $\mathsf{bad}_{m.4}$ be the corresponding event in Game $m.4$. Similar to above, it holds that Game $m.3$ and Game $m.4$ are identical as long as the respective events $\mathsf{bad}_{m.3}$ and $\mathsf{bad}_{m.4}$ do not occur, and $\Pr[\mathsf{bad}_{m.3}] = \Pr[\mathsf{bad}_{m.4}]$. We postpone the proof of the following claim:

Lemma 5. $\Pr[\mathsf{bad}_{m.3}] \leq Lq \cdot \max\{\mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k), \mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k)\}$.

Writing $\mathsf{Adv}^{\mathsf{xac}}_{\mathsf{XAC}}(k) := \max\{\mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k), \mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k)\}$, it follows that

$$|\Pr[out_{m.4} = 1] - \Pr[out_{m.3} = 1]| \leq \Pr[\mathsf{bad}_{m.3}] \leq Lq \cdot \mathsf{Adv}^{\mathsf{xac}}_{\mathsf{XAC}}(k). \tag{11}$$

In **Game** $m.5$, we do not sample a random $X^*_m \leftarrow \mathcal{X}_\rho$ if $M^*_m = 0$, but instead a consistent $X^*_m \in \mathcal{L}_\rho$. Concretely, the experiment always runs $X^*_m \leftarrow \mathsf{SampleL}(\mathcal{L}_\rho; W^*_m)$. (Later, if C^* is to be opened, X^*_m is explained as being randomly through $\mathsf{Sample}(\mathcal{X}_\rho)$, using random coins $\mathsf{Explain}(\mathcal{X}_\rho, X^*_m)$.) Since Game $m.4$ is again efficient , we can use the subset membership assumption to obtain

$$\frac{1}{L} \sum_{m \in [L]} (\Pr[out_{m.5} = 1] - \Pr[out_{m.4} = 1]) = \mathsf{Adv}^{\mathsf{sm}}_{\mathsf{EHPS},D}(k) \tag{12}$$

for a suitable D that guesses m uniformly and simulates Game $m.4$, resp. Game $m.5$, (implicitly) depending on its challenge.

Because $K^*_m = hsk(X^*_m, t^*) = \mathsf{PEval}(hpk, C^*_m, W^*_m, t^*)$ in Game $m.5$, Game $m.5$ is nothing but a reformulation of Game $m+1$. Hence, summing up (9,10,11,12) over $m \in [L]$ yields

$$|\Pr[out_L = 1] - \Pr[out_0 = 1]| \leq L \cdot \left|\mathsf{Adv}^{\mathsf{sm}}_{\mathsf{EHPS},D}(k)\right| + 2L^2 q \cdot \mathsf{Adv}^{\mathsf{xac}}_{\mathsf{XAC}}(k). \tag{13}$$

It is left to observe that in **Game** L, the experiment is exactly that of $\mathsf{Exp}^{\mathsf{so\text{-}ideal}}_{\mathsf{NCCCA},S}$ for the NC-CCA simulator S described earlier. Therefore,

$$\Pr\left[\mathsf{Exp}^{\mathsf{so\text{-}ideal}}_{\mathsf{NCCCA},S}(k) = 1\right] = \Pr[out_L = 1]. \tag{14}$$

Combining (6,7,8,13,14) finishes the proof.

We catch up with the proofs of the two technical lemmas:

Proof (of Lemma 4). Let $\mathsf{bad}_{m.2.j.i}$ denote the event that in Game $m.2$, the EHPS ciphertext X_i in some C^j is inconsistent in the sense $X_i \notin \mathcal{L}_\rho$, but $\mathsf{XVer}(\overline{K}^j_i, i, T^j) = 1$. Note $\mathsf{bad}_{m.2} = \bigvee_{(j,i) \in [q] \times [L]} \mathsf{bad}_{m.2.j.i}$.

Fix $(j,i) \in [q] \times [L]$. If $X^j_i \notin \mathcal{L}_\rho$, universality of EHPS implies that $\overline{K}^j_i = hsk(X^j_i, t^j)$ is uniformly random and independent of A's view. (Recall that A's view in Game $m.2$ depends only on hpk.) Hence

$$\Pr[\mathsf{bad}_{m.2.j.i}] \leq \mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k),$$

and a union bound over j and i shows the claim.

Proof (of Lemma 5). Let $\mathsf{bad}_{m.3.j.i}$ denote the event that in Game $m.3$, the EHPS ciphertext X_i in some C^j is inconsistent in the sense $X_i \notin \mathcal{L}_\rho$, but $\mathsf{XVer}(\overline{K}^j_i, i, T^j) = 1$. Note $\mathsf{bad}_{m.3} = \bigvee_{(j,i) \in [q] \times [L]} \mathsf{bad}_{m.3.j.i}$.

Fix $(j,i) \in [q] \times [L]$. We may assume that $X^j_i \notin \mathcal{L}_\rho$ (as necessary for $\mathsf{bad}_{m.3.j.i}$). Suppose first that $(X^j_i, t^j) \neq (X^*_m, t^*)$. Recall that A's information on hsk in Game $m.3$ is restricted to hpk and $K^*_m = hsk(X^*_m, t^*)$. Thus, EHPS's 2-universality implies

that $\overline{K}_i^j = hsk(X_i^j, t^j)$ is uniformly random and independent of A's view. By XAC's security against impersonation attacks,

$$\Pr\left[\mathsf{bad}_{m.3.j.i} \mid (X_i^j, t^j) \neq (X_m^*, t^*)\right] \leq \mathsf{Adv}^{\mathsf{imp}}_{\mathsf{XAC}}(k). \tag{15}$$

Now suppose $(X_i^j, t^j) = (X_m^*, t^*)$. By our changes in Games 0 and 1, we may assume that $(X_{i'}^j)_{i' \in [L]} = (X_{i'}^*)_{i' \in [L]}$, so that necessarily $m = i$ and $\overline{K}_i^j = K_i^* = hsk(X_i^*, t^*)$. Furthermore, for the decryption query to be valid, $T^j \neq T^*$ has to hold. EHPS's universality implies that $K_i^* = hsk(X_i^*, t^*)$ is uniformly distributed and the only information A has on K_i^* is T^*. By XAC's security against substitution attacks,

$$\Pr\left[\mathsf{bad}_{m.3.j.m} \mid (X_i^j, t^j) = (X_m^*, t^*)\right] \leq \mathsf{Adv}^{\mathsf{sub}}_{\mathsf{XAC}}(k). \tag{16}$$

A union bound and summing up (15,16) shows the inequality part of the lemma. The equality part follows by noting $\Pr[\mathsf{bad}_{m.3}] = \Pr[\mathsf{bad}_{m.4}]$, as in the proof of Lemma 4.

Acknowledgements. This work was initiated at the Securing Cyberspace Reunion Conference in 2009; we thank IPAM and the conference organizers for their hospitality. We are also grateful to the anonymous referees for helpful comments.

Note: The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the US Government, the UK Ministry of Defense, or the UK Government.

References

[1] Beaver, D., Haber, S.: Cryptographic protocols provably secure against dynamic adversaries. In: Rueppel, R.A. (ed.) EUROCRYPT 1992. LNCS, vol. 658, pp. 307–323. Springer, Heidelberg (1992)

[2] Bellare, M., Desai, A., Pointcheval, D., Rogaway, P.: Relations among notions of security for public-key encryption schemes. In: Krawczyk, H. (ed.) CRYPTO 1998. LNCS, vol. 1462, pp. 26–45. Springer, Heidelberg (1998)

[3] Bellare, M., Hofheinz, D., Yilek, S.: Possibility and impossibility results for encryption and commitment secure under selective opening. In: EUROCRYPT 2009. LNCS, vol. 5479, pp. 1–35. Springer, Heidelberg (2009)

[4] Bellare, M., Waters, B., Yilek, S.: Identity-based encryption secure under selective opening attack (manuscript, 2010)

[5] Bellare, M., Yilek, S.: Encryption Schemes Secure under Selective Opening Attack. Cryptology ePrint Archive, Report 2009/101 (2009)

[6] Blum, M., Micali, S.: How to generate cryptographically strong sequences of pseudorandom bits. SIAM Journal on Computing 13(4), 850–864 (1984)

[7] Boneh, D., Franklin, M.K.: Identity-based encryption from the Weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001)

[8] Boneh, D., Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity-based encryption. SIAM Journal on Computing 36(5), 915–942 (2006)

[9] Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: 42nd FOCS, October 2001, pp. 136–145. IEEE Computer Society Press, Los Alamitos (2001)

[10] Canetti, R., Feige, U., Goldreich, O., Naor, M.: Adaptively secure multi-party computation. In: 28th ACM STOC, pp. 639–648. ACM Press, New York (1996)

[11] Canetti, R., Dwork, C., Naor, M., Ostrovsky, R.: Deniable encryption. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 90–104. Springer, Heidelberg (1997)
[12] Canetti, R., Krawczyk, H., Nielsen, J.B.: Relaxing chosen-ciphertext security. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 565–582. Springer, Heidelberg (2003)
[13] Cramer, R., Shoup, V.: A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: Krawczyk, H. (ed.) CRYPTO 1998. LNCS, vol. 1462, pp. 13–25. Springer, Heidelberg (1998)
[14] Cramer, R., Shoup, V.: Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 45–64. Springer, Heidelberg (2002)
[15] Damgård, I., Nielsen, J.B.: Improved non-committing encryption schemes based on general complexity assumptions. In: Bellare, M. (ed.) CRYPTO 2000. LNCS, vol. 1880, pp. 432–450. Springer, Heidelberg (2000)
[16] Dolev, D., Dwork, C., Naor, M.: Nonmalleable cryptography. SIAM Journal on Computing 30(2), 391–437 (2000)
[17] Dwork, C., Naor, M., Reingold, O., Stockmeyer, L.: Magic functions. Journal of the ACM 50(6), 852–921 (2003)
[18] Goldreich, O.: Foundations of Cryptography: Basic Applications, vol. 2. Cambridge University Press, Cambridge (2004)
[19] Goldwasser, S., Micali, S.: Probabilistic encryption. Journal of Computer and System Sciences 28(2), 270–299 (1984)
[20] Hemenway, B., Libert, B., Ostrovsky, R., Vergnaud, D.: Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. Cryptology ePrint Archive, Report 2009/088 (2009)
[21] Hofheinz, D.: Possibility and impossibility results for selective decommitments. IACR ePrint Archive (2008)
[22] Hofheinz, D., Kiltz, E.: Secure hybrid encryption from weakened key encapsulation. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 553–571. Springer, Heidelberg (2007)
[23] Hofheinz, D., Kiltz, E.: Practical chosen ciphertext secure encryption from factoring. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 313–332. Springer, Heidelberg (2009)
[24] Katz, J., Ostrovsky, R.: Round-optimal secure two-party computation. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 335–354. Springer, Heidelberg (2004)
[25] Kiltz, E., Pietrzak, K., Stam, M., Yung, M.: A new randomness extraction paradigm for hybrid encryption. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 590–609. Springer, Heidelberg (2009)
[26] Kurosawa, K., Desmedt, Y.: A new paradigm of hybrid encryption scheme. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 426–442. Springer, Heidelberg (2004)
[27] Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: 22nd STOC. ACM Press, New York (1990)
[28] Nielsen, J.B.: Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 111–126. Springer, Heidelberg (2002)
[29] Peikert, C., Waters, B.: Lossy trapdoor functions and their applications. In: 40th STOC, pp. 187–196. ACM Press, New York (2008)
[30] Rackoff, C., Simon, D.R.: Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 433–444. Springer, Heidelberg (1992)