# Public-Key Steganography with Active Attacks

Michael Backes and Christian Cachin

IBM Zurich Research Laboratory
CH-8803 Rüschlikon, Switzerland
{mbc,cca}@zurich.ibm.com

**Abstract.** A complexity-theoretic model for public-key steganography with active attacks is introduced. The notion of *steganographic security against adaptive chosen-covertext attacks (SS-CCA)* and a relaxation called *steganographic security against publicly-detectable replayable adaptive chosen-covertext attacks (SS-PDR-CCA)* are formalized. These notions are closely related to *CCA-security* and *PDR-CCA-security* for public-key cryptosystems. In particular, it is shown that any SS-(PDR-)CCA stegosystem is a (PDR-)CCA-secure public-key cryptosystem and that an SS-PDR-CCA stegosystem for any covertext distribution with sufficiently large min-entropy can be realized from any PDR-CCA-secure public-key cryptosystem with pseudorandom ciphertexts.

## 1 Introduction

Steganography is the art and science of hiding information by embedding messages within other, seemingly harmless messages. As the goal of steganography is to hide the *presence* of a message, it can be seen as the complement of cryptography, whose goal is to hide the *content* of a message.

Consider two parties linked by a public communications channel which is under the control of an adversary. The parties are allowed to exchange messages as long as they are not adding a hidden meaning to their conversation. A genuine communication message is called *covertext*; but if the sender of a message has embedded hidden information in a message, it is called *stegotext*. The adversary, who also knows the distribution of the covertext, tries to detect whether a given message is covertext or stegotext.

Steganography has a long history as surveyed by Anderson and Petitcolas [2], but formal models for steganography have only recently been introduced. Several information-theoretic formalizations [4, 24, 15] and one complexity-theoretic model [12] have addressed *private-key* steganography, where the participants share a common secret key. These models are all limited to a passive adversary, however, who can only read messages on the channel.

Von Ahn and Hopper [22] have recently formalized *public-key* steganography with a passive adversary and, in a restricted model, also with an active adversary. Their notion offers security against "attacker-specific" chosen-stegotext attacks, where the recipient must know the identity of the sender, however; this is a limitation of the model compared to the bare public-key scenario.

In this paper, we introduce a complexity-theoretic model for public-key steganography with active attacks, where the participants a priori do not need shared secret information and the adversary may write to the channel and mount a so-called *adaptive chosen-covertext attack*. This attack seems to be the most general attack conceivable against a public-key stegosystem. It allows the adversary to send an arbitrary sequence of adaptively chosen covertext messages to a receiver and to learn the interpretation of every message, i.e., if the receiver considers a message to be covertext or stegotext, plus the decoding of the embedded message in the latter case. (Note that here and in the sequel, a message on the channel is sometimes also called a "covertext" when we do not want to distinguish between stegotext and covertext in the proper sense.)

We do not address denial-of-service attacks in this work, where the adversary tries to disrupt the hidden communication among the participants. Although they also qualify as "active" attacks and are very important in practice, we think that protection against them can be addressed orthogonally to the methods presented here.

Our model is based on the intuition that a public-key stegosystem essentially is a public-key cryptosystem with the additional requirement that its output conforms to a given covertext distribution. As in previous formalizations of steganography [4, 12, 9, 22], the covertext distribution is publicly known in the sense that it is accessible through an oracle that samples the distribution. We introduce the notions of *steganographic security against adaptive chosen-covertext attacks (SS-CCA)* and *steganographic security against publicly-detectable replayable adaptive chosen-covertext attacks (SS-PDR-CCA)* and show that they are closely linked to the analogous notions for public-key cryptosystems, called *security against adaptive chosen-ciphertext attacks* (or *CCA-security*) [16] and *security against publicly-detectable replayable adaptive chosen-ciphertext attacks* [5] (or *PDR-CCA-security*), respectively. (PDR-CCA-security is the same as *benign malleability* [19] and *generalized CCA-security* [1].)

In particular, we show that stegosystems are related to public-key cryptosystems in the following ways:

**Theorem 1 (informal statement).** *Any SS-(PDR-)CCA stegosystem is a (PDR-)CCA-secure public-key cryptosystem.*

**Theorem 2 (informal statement).** *An SS-PDR-CCA stegosystem for covertext distributions with sufficiently large min-entropy can be constructed from any PDR-CCA-secure public-key cryptosystem whose ciphertexts are pseudorandom (i.e., computationally indistinguishable from a random bit string).*

A corollary of Theorem 2 is that SS-PDR-CCA stegosystems exist in the standard model under the Decisional Diffie-Hellman (DDH) assumption and in the random oracle model under the assumption of trapdoor one-way permutations. The stegosystem constructed in the proof of Theorem 2 uses the "rejection sampler" construction found in essentially all previous work in the area [12, 9, 22], which is already described by Anderson and Petitcolas [2]. However, our system embeds more hidden bits per stegotext than any previous system. This follows from an improved analysis of the rejection sampler. It is not known if a result analogous to Theorem 2 holds for CCA-security; finding an SS-CCA stegosystem that works for an arbitrary covertext distribution with sufficiently large min-entropy remains an interesting open problem.

Our model for public-key steganography is introduced in Section 2, where also the relation to previous models for steganography is discussed in detail. Section 3 recalls the definitions of CCA- and PDR-CCA-security for public-key cryptosystems, states our results formally, and presents the proof of Theorem 1. Section 4 gives the construction of an SS-PDR-CCA stegosystem and proves Theorem 2.

## 2 Definitions

### 2.1 Notation

A function $f\colon \mathbb{N} \to \mathbb{R}_{\geq 0}$ is called *negligible* if for every constant $c \geq 0$ there exists $k_c \in \mathbb{N}$ such that $f(k) < \frac{1}{k^c}$ for all $k > k_c$. Given some set $\mathcal{S}$, a subset of *almost all* elements contains all but a negligible fraction of elements from $\mathcal{S}$. A (randomized) algorithm is called *efficient* if its running time is bounded by a polynomial except with negligible probability (over the coin tosses of the algorithm).

Let $x \leftarrow y$ denote the algorithm that assigns a value $y$ to $x$. If $\mathsf{A}(\cdot)$ is a (randomized) algorithm, the notation $x \leftarrow \mathsf{A}(y)$ denotes the algorithm that assigns to $x$ a randomly selected value according to the probability distribution induced by $\mathsf{A}(\cdot)$ with input $y$ over the set of its outputs.

If $\mathcal{S}$ is a probability distribution, then the notation $x \xleftarrow{R} \mathcal{S}$ denotes any algorithm which assigns to $x$ an element randomly selected according to $\mathcal{S}$. If $S$ is a finite set, then the notation $x \xleftarrow{R} S$ denotes the algorithm which assigns to $x$ an element selected at random from $S$ with uniform distribution over $S$.

If $p(\cdot, \cdot, \cdots)$ is a predicate, the notation

$$\Pr[x \xleftarrow{R} S; y \xleftarrow{R} T; \cdots : p(x, y, \cdots)]$$

denotes the probability that $p(x, y, \cdots)$ will be true after the ordered execution of the algorithms $x \xleftarrow{R} S$, $y \xleftarrow{R} T, \cdots$. If $X$ is a (randomized) algorithm, a distribution, or a set, then $\Pr_X[x]$ is short for $\Pr_{x \xleftarrow{R} X}[x]$, which is short for $\Pr[s \xleftarrow{R} X : s = x]$.

The *statistical distance* between two distributions $\mathcal{X}$ and $\mathcal{Y}$ over the same set $X$ is defined as $\|\mathcal{X} - \mathcal{Y}\| = \max_{X_0 \subseteq X} \left|\sum_{x \in X_0} \Pr_{\mathcal{X}}(x) - \Pr_{\mathcal{Y}}(x)\right|$. The *min-entropy* of a distribution $\mathcal{X}$ over an alphabet $X$ is defined as $H_\infty(\mathcal{X}) = -\log \max_{x \in X} \Pr_{\mathcal{X}}[x]$. (All logarithms are to the base 2.)

### 2.2 Public-key Stegosystems

We define a public-key stegosystem as a triple of algorithms for key generation, message encoding, and message decoding, respectively. The notion corresponds to a public-key cryptosystem in which the ciphertext should conform to a target covertext distribution.

For the scope of this work, the covertext is modeled by a distribution $\mathcal{C}$ over a given set $C$. The distribution is only available via an oracle; it samples $\mathcal{C}$ upon request, with each sample being independent. In other words, it outputs a sequence of independent and identically distributed covertexts. W.l.o.g., $\Pr_{\mathcal{C}}[c] > 0$ for all $c \in C$.

The restriction to independent repetitions is made here only to simplify the notation and to focus on the contribution of this work. All our definitions and results can be extended in the canonical way to the very general model of a covertext *channel* as introduced by Hopper et al. [12]. They model a channel as an unbounded sequence of values drawn from a set $C$ whose distribution may depend in arbitrary ways on past outputs; access to the channel is given only by an oracle that samples from the channel.

Such a channel underlies only one restriction: The sampling oracle must allow random access to the channel distribution, i.e., the oracle can be queried with an arbitrary prefix of a possible channel output and will return the next symbol according to the channel distribution. In other words, the channel sampler cannot only be rewound to an earlier state of its execution but also restarted from a given state. (Hence it may be difficult to use an email conversation among humans for a covertext channel since that cannot easily be restarted.)

The sampling oracle for the covertext distribution is available to all users and to the adversary. In order to avoid technical complications, assume w.l.o.g. that the sampling oracle is implemented by a probabilistic polynomial-time algorithm and therefore does not help an adversary beyond its own capabilities (for example, with solving a computationally hard problem).

**Definition 1.** *[Public-Key Stegosystem] Let $\mathcal{C}$ be a distribution on a set $C$ of* covertexts. *A* public-key stegosystem *is a triple of probabilistic polynomial-time algorithms* $(\mathsf{SK}, \mathsf{SE}, \mathsf{SD})$ *with the following properties.*

- *The* key generation algorithm $\mathsf{SK}$ *takes as input the security parameter $k$ and outputs a pair of bit strings $(spk, ssk)$, called the* [stego] public key *and the* [stego] secret key. *W.l.o.g.* $\mathsf{SK}$ *induces the uniform distribution over the set of possible key pairs for security parameter $k$.*
- *The* steganographic encoding algorithm $\mathsf{SE}$ *takes as inputs the security parameter $k$, a public key $spk$ and a* message $m \in \{0,1\}^{l(k)}$, *where $l(k)$ is an arbitrary polynomial, and outputs a* covertext $c \in C$. *The plaintext $m$ is often called the* embedded message.
- *The* steganographic decoding algorithm $\mathsf{SD}$ *takes as inputs the security parameter $k$, a secret key $ssk$, and a covertext $c \in C$ and outputs either a message $m \in \{0,1\}^{l(k)}$ or a special symbol $\perp$. An output value of $\perp$ indicates a decoding error, for example, when* $\mathsf{SD}$ *has determined that no message is embedded in $c$.*

*We require that for almost all $(spk, ssk)$ output by $\mathsf{SK}(1^k)$ and all $m \in \{0,1\}^{l(k)}$, the probability that $\mathsf{SD}(1^k, ssk, \mathsf{SE}(1^k, spk, m)) \neq m$ is negligible in $k$.*

Note that except for the presence of the covertext distribution, this definition is equivalent to that of a public-key cryptosystem. Although all algorithms have oracle access to $\mathcal{C}$, only $\mathsf{SE}$ needs it in the stegosystems considered in this paper. For ease of notation, the security parameter will be omitted henceforth.

The probability that the decoding algorithm outputs the correct embedded message is referred to as the *reliability* of the stegosystem. Although one might also allow a non-negligible decoding error in the definition of a stegosystem (as done in previous work [12]), we require that the decoding error probability is negligible in order to maintain the analogy between a stegosystem and a cryptosystem.

*Security definition.* Coming up with the “right” security definition for a cryptographic primitive has always been a challenging task because the sufficiency of a security property cannot be demonstrated by running the cryptosystem. Only its insufficiency can be shown by pointing out a specific attack, but finding an attack is usually hard. Often, security definitions had to be strengthened when a primitive was used as part of a larger system. Probably the most typical example is the security of public-key cryptosystems: the original notion of semantic security [11], which considers only a passive or eavesdropping adversary, was later augmented to security against adaptive chosen-ciphertext attacks or non-malleability, which allows also for active attacks [16, 10, 3].

We introduce here the notion of *steganographic security against adaptive chosen-covertext attacks*, abbreviated *SS-CCA*, and its slightly relaxed variant *steganographic security against publicly-detectable replayable chosen-covertext attacks*, abbreviated *SS-PDR-CCA*. Both notions are based on the intuition that a stegosystem is essentially a cryptosystem with a prescribed ciphertext distribution. We first recall the definition of *compatible [publicly computable] relations*, adopted from public-key cryptosystem to stegosystems, on which the definition of SS-PDR-CCA is based.

**Definition 2.** *[Compatible Relation [19]] Let $\Sigma = (\mathsf{SK}, \mathsf{SE}, \mathsf{SD})$ be a stegosystem. A family of binary relations $\equiv_{spk}$ (indexed by the public keys of $\Sigma$) on covertext pairs is called a* compatible *relation family for $\Sigma$ if for almost all key pairs $(spk, ssk)$ we have:*

- *For any two covertexts $c$ and $c'$, if $c \equiv_{spk} c'$ then $\mathsf{SD}(ssk, c) = \mathsf{SD}(ssk, c')$, except with negligible probability over the random choices of the algorithm $\mathsf{SD}$.*
- *For any two covertexts $c$ and $c'$, it can be determined except with negligible probability whether $c \equiv_{spk} c'$ using a probabilistic polynomial-time algorithm taking inputs $spk$, $c$, and $c'$.*

SS-CCA and SS-PDR-CCA are defined by the following experiment. Let an arbitrary distribution $\mathcal{C}$ on a set $C$ be given and consider a (stego-)adversary, defined by two arbitrary probabilistic polynomial-time algorithms $SA_1$ and $SA_2$. For the SS-PDR-CCA experiment, let also an arbitrary compatible relation family $\equiv_{spk}$ be given. The experiment consists of five stages, where both notions only differ in the fourth stage.

**Key generation:** A key pair $(spk, ssk)$ is generated by the key generation algorithm $\mathsf{SK}$.

**First decoding stage:** Algorithm $SA_1$ is run with the public key $spk$ as input and has access to the sampling oracle for $\mathcal{C}$ and to a decoding oracle $SO_1$. The decoding oracle knows the secret key $ssk$. Whenever it receives a covertext $c$, it runs $\mathsf{SD}(ssk, c)$ and returns the result to $SA_1$.

When $SA_1$ finishes its execution, it outputs a tuple $(m^*, s)$, where $m^* \in \{0, 1\}^l$ is a message and $s$ is some additional information which the algorithm wants to preserve.

**Challenge:** A bit $b$ is chosen at random and a *challenge covertext* $c^*$ is determined depending on it: If $b = 0$ then $c^* \leftarrow \mathsf{SE}(spk, m^*)$ else $c^* \overset{R}{\leftarrow} \mathcal{C}$. $c^*$ is given to algorithm $SA_2$, who should guess the value of $b$, i.e., determine whether the message $m^*$ has been embedded in $c^*$ or whether $c^*$ has simply been chosen according to $\mathcal{C}$.

**Second decoding stage:** $SA_2$ is run on input $c^*$, and $s$, i.e., it knows the challenge covertext and the state provided by $SA_1$.
For SS-CCA, $SA_2$ may access a decoding oracle $SO_2^{cca}$, which is analogous to $SO_1$ except that upon receiving query $c^*$, oracle $SO_2^{cca}$ returns $\perp$.
For SS-PDR-CCA, $SA_2$ has access to a decoding oracle $SO_2^{pdr\text{-}cca,\equiv_{spk}}$, which is identical to $SO_2^{cca}$ except that it does not allow any query that is equivalent to $c^*$ under $\equiv_{spk}$. In particular, upon receiving query $c$, $SO_2^{pdr\text{-}cca,\equiv_{spk}}$ returns $\perp$ if $c \equiv_{spk} c^*$; otherwise, it returns $\mathsf{SD}(ssk, c)$.

**Guessing stage:** When $SA_2$ finishes its execution, it outputs a bit $b'$.

The stego-adversary succeeds in distinguishing stegotext from covertext if $b' = b$ in the above experiment. We require that for a secure stegosystem, no efficient adversary can distinguish stegotext from covertext except with negligible probability over random guessing.

**Definition 3.** *[Steganographic Security against Active Attacks] Let $\mathcal{C}$ be a distribution on a covertext set $C$ and let $\Sigma = (\mathsf{SK}, \mathsf{SE}, \mathsf{SD})$ be a stegosystem. We say that $\Sigma$ is* steganographically secure against adaptive chosen-covertext attacks (SS-CCA) *with respect to $\mathcal{C}$ if for all probabilistic polynomial-time adversaries $(SA_1, SA_2)$, there exists a negligible function $\epsilon$ such that*

$$\Pr\Big[(spk, ssk) \leftarrow \mathsf{SK};\ (m^*, s) \leftarrow SA_1^{SO_1}(spk);\ b \xleftarrow{R} \{0,1\};$$
$$\textbf{\textit{if}}\ b = 0\ \textbf{\textit{then}}\ c^* \leftarrow \mathsf{SE}(spk, m^*)\ \textbf{\textit{else}}\ c^* \xleftarrow{R} \mathcal{C}:$$
$$SA_2^{SO_2^{cca}}(spk, m^*, c^*, s) = b\Big] \;=\; \frac{1}{2} + \epsilon(k).$$

*Similarly, we say that $\Sigma$ is* steganographically secure against publicly-detectable replayable adaptive chosen-covertext attacks (SS-PDR-CCA) *with respect to $\mathcal{C}$ if there exists a compatible relation family $\equiv_{spk}$ such that for all probabilistic polynomial-time adversaries $(SA_1, SA_2)$, there exists a negligible function $\epsilon$ such that the above equation holds with $SO_2^{cca}$ replaced by $SO_2^{pdr\text{-}cca,\equiv_{spk}}$.*

Note that this leaves the adversary free to query the decoding oracle with any element of the covertext space *before* the challenge is issued. By definition, an SS-CCA stegosystem is also SS-PDR-CCA.

### 2.3 Discussion

*The relation to public-key cryptosystems.* A stegosystem should enable two parties to communicate over a public channel in such a way that the presence of a message in the conversation cannot be detected by an adversary. It seems natural to conclude from this that the adversary must not learn any useful information about an embedded message, should there be one. The latter property is the subject of cryptography: hiding the content of a message transmitted over a public channel. This motivates the approach of von Ahn and Hopper [22] and of this paper that models a public-key stegosystem after a public-key cryptosystem in which the ciphertext conforms to a particular covertext distribution.

The most widely accepted formal notion of a public-key cryptosystem secure against an active adversary is *indistinguishability of encryptions against an adaptive chosen-ciphertext attack* (CCA-security) [16] and is equivalent to *non-malleability of ciphertexts* in the same attack model [10, 3]. CCA-security is defined by an experiment with almost the same stages as above, except that the first part of the adversary outputs *two* messages $m_0$ and $m_1$, of which one is chosen at random and then encrypted. The resulting value $c^*$, also called the *target ciphertext,* is returned to the adversary and the adversary has to guess what has been encrypted. In the second query stage, the adversary is allowed to obtain decryptions of *any* ciphertext except for $c^*$.

This appears to be the minimal requirement to make the definition of a cryptosystem meaningful, but it has turned out to be overly restrictive in some cases. For example, consider a CCA-secure cryptosystem where a useless bit is appended to each ciphertext during encryption and that is ignored during decryption. Although this clearly does not affect the security of the cryptosystem, the modified scheme is no longer CCA-secure.

Several authors have relaxed CCA-security to allow for such “benign” modifications [19, 1, 5]. The corresponding relaxed security notion has been called *publicly-detectable replayable CCA-security* or *PDR-CCA-security* by Canetti et al. [5] because the modifications are apparent without knowledge of the secret key. The difference to CCA-security is that in the second query stage, the adversary is more restricted and does not allow any query that is equivalent to the target ciphertext under some compatible relation that can be derived from the public key. The intuition is that such a cryptosystem allows anyone to modify a ciphertext into an equivalent one if this is apparent from the public key, and therefore to “replay” the target ciphertext.

Our notion of an SS-CCA stegosystem is analogous to a CCA-secure cryptosystem, in that it only excludes the target covertext from the queries to the second decoding oracle. Likewise, our notion of an SS-PDR-CCA stegosystem contains a restriction that is reminiscent of a PDR-CCA-secure cryptosystem, by not allowing queries that are publicly identifiable transformations of the challenge covertext. These similarities are no coincidence: We show in Section 3 that any SS-CCA stegosystem is a CCA-secure public-key cryptosystem, and similarly for their replayable counterparts.

Canetti et al. [5] also propose a further relaxation of CCA-security called *replayable CCA-security* (or *R-CCA-security*), where anyone can generate new ciphertexts that decrypt to the same value as a given ciphertext, but the equivalence may not be publicly detectable. We note that it is possible to formulate the corresponding notion of *steganographic security against replayable chosen-ciphertext attacks* (*SS-R-CCA*) by suitably modifying Definition 3. Our results of Sections 3 and 4 can be adapted analogously.

*Related work on steganography.* The first published model of a steganographic system is the “Prisoners’ Problem” by Simmons [21]. This work addresses the particular situation of message authentication among two communicating parties, where a so-called *subliminal channel* might be used to transport a hidden message in the view of an adversary who tries to detect the presence of a hidden message. Although a subliminal channel in that sense is only made possible by the existence of message authentication in the model, it can be seen as the first formulation of a general model for steganography.

Cachin [4] presented an information-theoretic model for steganography, which was the first to explicitly require that the stegotext distribution is indistinguishable from the covertext distribution to an adversary. Since the model is unconditional, a statistical information measure is used.

Hopper et al. [12] give the first complexity-theoretic model for private-key steganography with passive attacks; they point out that a stegosystem is similar to a cryptosystem whose ciphertext is indistinguishable from a given covertext. In Section 3 we establish such an equivalence formally for public-key systems with active attacks.

Dedić et al. [9] study the efficiency of stegosystems that have black-box access to the covertext distribution and provide lower bounds on their efficiency.

Recently, von Ahn and Hopper [22] have formalized public-key steganography with a passive adversary, i.e., one who can mount a chosen-message attack. The resulting notion is the analogue of a cryptosystem with security against chosen-plaintext attacks (i.e., a cryptosystem with semantic security). They also formalize the notion of a stegosystem that offers security against "attacker-specific" chosen-stegotext attacks; this means that the decoder must know the identity of the encoder, however, and restricts the usefulness of their notion compared to SS-CCA and SS-PDR-CCA.

No satisfying formal model for public-key steganography with active attacks has been published so far, although the subject was discussed by several authors, and some systems with heuristic security have been proposed [8, 2]. A crucial element that seems to make our formalizations useful is the restriction of the stage-two decoding oracle depending on the challenge covertext.

## 3 Results

This section investigates the relation between SS-(PDR-)CCA stegosystems and (PDR-)CCA-secure public-key cryptosystems. Two results are presented:

1. Any SS-CCA stegosystem is a CCA-secure public-key cryptosystem and, similarly, any SS-PDR-CCA stegosystem is a PDR-CCA-secure public-key cryptosystem.
2. An SS-PDR-CCA stegosystem for covertext distributions with sufficiently large min-entropy can be constructed from any PDR-CCA-secure public-key cryptosystem whose ciphertexts are pseudorandom.

We first recall the formal definitions for public-key encryption with CCA- and PDR-CCA-security, respectively. A *public-key cryptosystem* is a triple $(\mathsf{K}, \mathsf{E}, \mathsf{D})$ of probabilistic polynomial-time algorithms. Algorithm $\mathsf{K}$, on input the security parameter $k$, generates a pair of keys $(pk, sk)$. The encryption and decryption algorithms, $\mathsf{E}$ and $\mathsf{D}$, have the property that for almost all pairs $(pk, sk)$ generated by $\mathsf{K}$ and for any plaintext message $m \in \{0, 1\}^{l(k)}$ where $l$ is an arbitrary polynomial in $k$, the probability that $\mathsf{D}(1^k, sk, \mathsf{E}(1^k, pk, m)) \neq m$ is negligible in $k$. (The security parameter is omitted henceforth.)

CCA-security and PDR-CCA-security for a public-key encryption scheme are defined by the following experiment. Consider an adversary defined by two arbitrary polynomial-time algorithms $A_1$ and $A_2$. First, a key pair $(pk, sk)$ is generated by $\mathsf{K}$. Next, $A_1$ is run on input the public key $pk$ and may access a decryption oracle $O_1$. Oracle $O_1$ knows the secret key $sk$, and whenever it receives a ciphertext $c$, it applies $\mathsf{D}$ with key $sk$ to $c$ and returns the result to $A_1$. When $A_1$ finishes its execution, it outputs a triple $(m_0, m_1, s)$, where $m_0, m_1 \in \{0,1\}^l$ are two arbitrary messages and $s$ is some additional state information. Now a bit $b$ is chosen at random and $m_b$ is encrypted using $\mathsf{E}$ under key $pk$, resulting in a ciphertext $c^*$. Algorithm $A_2$ is given $m_0$ and $m_1$, ciphertext $c^*$, and state $s$, and has to guess the value of $b$, i.e., whether $m_0$ or $m_1$ has been encrypted. For CCA-security, $A_2$ may access a decryption oracle $O_2^{cca}$, which is analogous to $O_1$ and knows $sk$, but returns $\perp$ upon receiving query $c^*$. For PDR-CCA-security, the cryptosystem also specifies a compatible relation family $\equiv pk$ according to Definition 2 with the stegosystem being replaced by the cryptosystem. $A_2$ may access a decryption oracle $O_2^{pdr\text{-}cca,\equiv_{pk}}$, which is identical to $O_1^{cca}$ except that it answers $\perp$ for any query $c$ with $c \equiv_{pk} c^*$. Finally, $A_2$ outputs a bit $b'$ as its guess for $b$.

A secure cryptosystem requires that no efficient adversary can distinguish an encryption of $m_0$ from an encryption of $m_1$ except with negligible probability.

**Definition 4.** *[(PDR-)CCA-Security for Public-Key Cryptosystems [3,5]] Let $\Omega = (\mathsf{K}, \mathsf{E}, \mathsf{D})$ be a public-key cryptosystem. We say that $\Omega$ is* CCA-secure *if for all probabilistic polynomial-time adversaries $A = (A_1, A_2)$, there exists a negligible function $\epsilon$ such that*

$$\Pr\Big[(pk, sk) \leftarrow \mathsf{K};\ (m_0, m_1, s) \leftarrow A_1^{O_1}(pk);\ b \xleftarrow{R} \{0,1\};$$
$$c^* \leftarrow \mathsf{E}(pk, m_b);\ A_2^{O_2^{cca}}(pk, m_0, m_1, c^*, s) = b\Big] \ = \ \frac{1}{2} + \epsilon(k).$$

*We say that $\Omega$ is* PDR-CCA-secure *if there exists a compatible relation family $\equiv_{pk}$ such that the above condition holds with $O_2^{cca}$ replaced by $O_2^{pdr\text{-}cca,\equiv_{pk}}$.*

The following is our first main result.

**Theorem 1.** *Let $\Sigma = (\mathsf{SK}, \mathsf{SE}, \mathsf{SD})$ be a public-key stegosystem. If $\Sigma$ is SS-CCA (SS-PDR-CCA) with respect to some distribution $\mathcal{C}$, then $\Sigma$ is a CCA-secure (PDR-CCA-secure) public-key cryptosystem.*

*Proof.* Note first that $\Sigma$ satisfies the definition of a public-key cryptosystem. We prove that $\Sigma$ is (PDR-)CCA-secure by a reduction argument. Assume that $\Sigma$ is not a (PDR-)CCA-secure cryptosystem and hence there exists an (encryption-)adversary $(A_1, A_2)$ that breaks the (PDR-)CCA-security of $\Sigma$, i.e., it wins in the experiment of Definition 4 with probability $\frac{1}{2} + \delta(k)$ for some non-negligible function $\delta$. Let $\equiv_{pk}$ denote a compatible relation family for $\Sigma$ in the case of PDR-CCA security. We construct a (stego-)adversary $(SA_1, SA_2)$ against $\Sigma$ as a stegosystem with respect to $\mathcal{C}$ that has black-box access to $(A_1, A_2)$ as follows.

**Key generation:** When $SA_1$ receives a public-key $pk$, it invokes $A_1$ with this key.

**First decoding stage:** Whenever $A_1$ queries its decryption oracle $O_1$ with a ciphertext $c$, $SA_1$ passes $c$ on to its decoding oracle $SO_1$, waits for the response and forwards the response to $A_1$.

When $A_1$ halts and outputs $(m_0, m_1, s)$, the stego-adversary $SA_1$ chooses a random bit $b'$, and outputs $(m_{b'}, (m_0, m_1, b', s))$.

**Challenge:** A challenge covertext $c^*$ is computed according to the definition of a stegosystem and given to $SA_2$.

**Second decoding stage:** $SA_2$ receives inputs $m_{b'}$, $c^*$, and $(m_0, m_1, b', s)$ and invokes $A_2$ on inputs $m_0$, $m_1$, $c^*$, and $s$. Otherwise, $SA_2$ behaves in the same way as $SA_1$ during the first decoding stage, forwarding the decryption requests that $A_2$ makes to $O_2$ to the respective decoding oracle $SO_2^{cca}$ or $SO_2^{pdr\text{-}cca,\equiv_{pk}}$ and the responses back to $A_2$. If the distinction between $SO_2^{cca}$ and $SO_2^{pdr\text{-}cca,\equiv_{pk}}$ is irrelevant, we simply write $SO_2$, similarly for the decryption oracle $O_2$.

**Guessing stage:** When $A_2$ outputs a bit $b^*$, the stego-adversary $SA_2$ tests if $b^* = b'$ and outputs 0 if true, and 1 otherwise.

We now analyze the environment simulated by the stego-adversary $(SA_1, SA_2)$ to the encryption-adversary $(A_1, A_2)$, and the probability that the stego-adversary can distinguish stegotext from covertext.

Clearly, key generation and the first decoding stage perfectly simulate the decryption oracle to adversary $A_1$. During the challenge, a random bit $b$ is chosen and a challenge covertext is computed as $c^* \leftarrow \mathsf{SE}(pk, m_{b'})$ in case $b = 0$ and as $c^* \overset{R}{\leftarrow} \mathcal{C}$ in case $b = 1$.

Note that when $b = 1$, algorithm $A_2$ and its final output $b^*$ are independent of $b'$. Hence, we have $\Pr[b' = b^* | b = 1] = \frac{1}{2}$ and the stego-adversary has no advantage over randomly guessing $b'$ in that case. When $b = 0$, we show that during the second decoding phase, $SA_2$ correctly simulates the decryption oracle $O_2$ to $A_2$. For SS-CCA, correct simulation for queries $c \neq c^*$ is clear by definition. For a query $c = c^*$, the decoding oracle $SO_2^{cca}$ will output $\perp$, and so will the decryption oracle $O_2^{cca}$, which gives a correct simulation again. For SS-PDR-CCA, correct simulation for queries $c \not\equiv_{pk} c^*$ is again clear by definition. For queries $c$ with $c \equiv_{pk} c^*$, the decoding oracle $SO_2^{pdr\text{-}cca,\equiv_{pk}}$ will output $\perp$, and so will the decryption oracle $O_2^{pdr\text{-}cca,\equiv_{pk}}$.

Since the encryption-adversary $A_2$ by assumption breaks the (PDR-)CCA-security of the cryptosystem, and $A_2$ is independent of $b'$ when $b = 1$ as argued above, it obtains all its advantage in the case $b = 0$ and we have $\Pr[b' = b^* | b = 0] = \frac{1}{2} + \delta(k)$. By the definition of $SA_2$, this is also the probability that the stego-adversary guesses $b$ correctly when $b = 0$. Hence, the overall probability that $SA_2$ guesses $b$ correctly is $\frac{1}{2} + \frac{\delta(k)}{2}$, which exceeds $\frac{1}{2}$ by a non-negligible quantity and shows that $\Sigma$ is not SS-(PDR-)CCA with respect to any $\mathcal{C}$.

Theorem 1 shows that an SS-CCA stegosystem is a special case of a CCA-secure public-key cryptosystem, and similarly for their replayable variants. In the converse direction, we show now that some PDR-CCA-secure public-key cryptosystems, namely those with "pseudorandom ciphertexts," can also be used to construct SS-PDR-CCA stegosystems. Constructing an SS-CCA stegosystem from a CCA-secure public-key cryptosystem — or from other assumptions, for that matter — for an arbitrary covertext distribution with sufficiently large min-entropy remains an open problem.

In a cryptosystem with pseudorandom ciphertexts, the encryption algorithm outputs a bit string that is indistinguishable from a random string of the same length for any efficient distinguisher that has knowledge of the public key. We make the assumption that the encryption of a plaintext of length $l(k)$ always results in a ciphertext of length $n(k)$, for some polynomial $n$ in $k$.

**Definition 5.** *[Public-key Cryptosystem with Pseudorandom Ciphertexts [22]] A public-key cryptosystem $(\mathsf{K}, \mathsf{E}, \mathsf{D})$ is said to have* pseudorandom ciphertexts *if for all probabilistic polynomial-time adversaries $A = (A_1, A_2)$, there exists a negligible function $\epsilon$ such that*

$$\begin{aligned}\Pr\Big[&(pk, sk) \leftarrow \mathsf{K};\ (m, s) \leftarrow A_1(pk);\ c_0 \leftarrow \mathsf{E}(pk, m);\ c_1 \overset{R}{\leftarrow} \{0,1\}^{n(k)};\\ &\qquad\qquad b \overset{R}{\leftarrow} \{0,1\};\ A_2(pk, m, c_b, s) = b\Big] \;=\; \frac{1}{2} + \epsilon(k).\end{aligned}$$

It seems difficult to construct SS-(PDR-)CCA stegosystems for *any* covertext distribution. We show that it is possible for covertexts whose distribution conforms to a sequence of independently repeated experiments and has sufficiently large min-entropy. (According to the remark in Section 2.2, this result generalizes to an arbitrary covertext *channel*.) Given a covertext distribution $\mathcal{C}$ and positive $t$, let $\mathcal{C}^t$ denote the probability distribution consisting of a sequence of $t$ independent repetitions of $\mathcal{C}$.

The next theorem is our second main result. Its proof is the subject of Section 4.

**Theorem 2.** *SS-PDR-CCA stegosystems with respect to a covertext distribution $\mathcal{C}^t$ for any $\mathcal{C}$ with sufficiently large min-entropy can be efficiently constructed from any PDR-CCA-secure cryptosystem with pseudorandom ciphertexts.*

Theorem 2 leaves us with the task of finding a PDR-CCA-secure cryptosystem with pseudorandom ciphertexts. Such cryptosystems exist under a variety of standard assumptions if one asks for security against a *passive* adversary only, i.e., security against *chosen-plaintext attacks (CPA)*. For example, von Ahn and Hopper [22] demonstrate a scheme that is as secure as RSA and one that is secure under the Decisional Diffie-Hellman (DDH) assumption. It is also straightforward to verify that the generic method of encrypting a single bit by xoring it with the hard-core predicate of a trapdoor one-way permutation has pseudorandom ciphertexts.

But any PDR-CCA-secure cryptosystem can be turned into one with pseudorandom ciphertexts using the following method, suggested by Lindell [13]: Take the ciphertext output by the PDR-CCA-secure encryption algorithm and encrypt it again, using a second cryptosystem with pseudorandom ciphertexts, which is secure against chosen-plaintext attacks. Decryption proceeds analogously, by first applying the decryption operation of the second cryptosystem and then the decryption operation of the PDR-CCA-secure cryptosystem. It can be verified that the composed cryptosystem retains PDR-CCA-security because the stage-two decryption oracle knows both secret keys. This method yields SS-PDR-CCA stegosystems in three different models as follows.

By applying the above generic CPA-secure cryptosystem with pseudorandom ciphertexts to a generic non-malleable cryptosystem [10, 18], we obtain an SS-PDR-CCA stegosystem under general assumptions.

**Corollary 1.** *Provided that trapdoor one-way permutations exist, there is an SS-PDR-CCA stegosystem in the common random string model.*

Using the above DDH-based cryptosystem with pseudorandom ciphertexts combined with the Cramer-Shoup cryptosystem [7], we obtain also an efficient SS-PDR-CCA stegosystem in the standard model.

**Corollary 2.** *Under the Decisional Diffie-Hellman assumption, there is an SS-PDR-CCA stegosystem.*

A more practical cryptosystem with pseudorandom ciphertexts exists also in the random oracle model: the OAEP+ scheme of Shoup [20]. OAEP+ is a CCA-secure cryptosystem based on an arbitrary trapdoor one-way permutation.

**Corollary 3.** *Provided that trapdoor one-way permutations exist, there is an SS-PDR-CCA stegosystem in the random oracle model.*

## 4 An SS-PDR-CCA Stegosystem

In this section, we propose a stegosystem that is steganographically secure against publicly-detectable replayable adaptive chosen-covertext attacks.

This stegosystem works for any covertext distribution that consists of a sequence of independent repetitions of a base-covertext distribution. Deviating from the notation of Section 2, we denote the base-covertext distribution by $\mathcal{C}$ and the covertext distribution used by the stegosystem by $\mathcal{C}^t = \Pi_{i=1}^t \mathcal{C}$. As noted in Section 2.2, through the introduction of a history, our construction also generalizes to arbitrary covertext channels.

Let $(\mathsf{K}, \mathsf{E}, \mathsf{D})$ be a PDR-CCA-secure public-key cryptosystem with pseudorandom ciphertexts and compatible relation $\equiv_{pk}$. Suppose its cleartexts are $l$-bit strings and its ciphertexts are $n$-bit strings.

A class $G$ of functions $X \to Y$ is called *strongly* $2$*-universal* [23] if, for all distinct $x_1, x_2 \in X$ and all (not necessarily distinct) $y_1, y_2 \in Y$, exactly $|G|/|Y|^2$ functions from $G$ take $x_1$ to $y_1$ and $x_2$ to $y_2$. Such a function family is sometimes simply called a *strongly 2-universal hash function* for brevity.

### 4.1 Description

The SS-PDR-CCA stegosystem consists of a triple of algorithms $(\mathsf{keygen}, \mathsf{encode}, \mathsf{decode})$. The idea behind it is to encrypt a message using the public-key cryptosystem first and to embed the resulting ciphertext into a covertext sequence, as shown in Figure 1.

The encoding method is based on the following algorithm $\mathsf{sample}$, which has oracle access to $\mathcal{C}$ and samples a base-covertext according to $\mathcal{C}$ such that a given $f$-bit string $b$ is embedded in it. This algorithm is the well-known rejection sampler [2, 12, 17, 9], generalized to embed multi-bit messages instead of only single-bit messages.

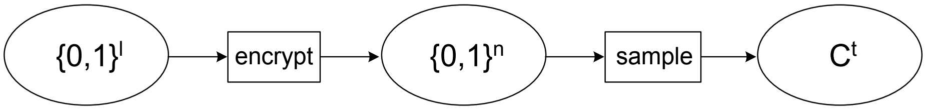

**Fig. 1.** The encoding process of the stegosystem: a message is first encrypted and then embedded using Algorithm sample. The decoding process works analogously in the reverse direction.

**Algorithm** sample$^{\mathcal{C}}$

```
Input: security parameter $k$, a function $g : C \to \{0,1\}^f$, and a value $b \in \{0,1\}^f$
Output: a covertext $x$
$j \leftarrow 0$
repeat
    $x \overset{R}{\leftarrow} \mathcal{C}$
    $j \leftarrow j + 1$
until $g(x) = b$ or $j = k$
return $x$
```

Intuitively, algorithm sample returns a covertext chosen from distribution $\mathcal{C}$, but restricted to that subset of $C$ which is mapped to the given $b$ by $g$. sample may also fail and return a covertext $c$ with $g(c) \neq b$, but this happens only with negligible probability in $k$. As will be shown in Section 4.2, when $b$ is a random $f$-bit string, $g$ is chosen randomly from a 2-universal hash function, and $\mathcal{C}$ has sufficient min-entropy, then the output distribution of sample is statistically close to $\mathcal{C}$.

We now turn to the description of the stegosystem. Let $f \leq \gamma \log k$ for a positive constant $\gamma < 1$ and let $G : C \to \{0,1\}^f$ denote a strongly 2-universal hash function.

Algorithm keygen chooses a random $g \overset{R}{\leftarrow} G$ and computes a tuple $(pk, sk) \leftarrow \mathsf{K}$, by running the key generation algorithm of the cryptosystem. The output of keygen is the tuple $(spk, ssk) = ((pk, g), sk)$.

Algorithm encode first encrypts an input message $m$ using the given encryption algorithm $\mathsf{E}$, which outputs in a ciphertext $y$. Assuming w.l.o.g. that $y$ is an $n$-bit string (bounded by a polynomial in $k$) and $n = tf$, encode then repeatedly invokes sample to embed $y$ in pieces of $f$ bits a time into a sequence of $t$ covertext symbols. Formally:

**Algorithm** encode

```
Input: security parameter $k$, a public key $spk = (pk, g)$, and a message $m \in \{0,1\}^l$
    to encode
Output: a covertext $(c_1, \ldots, c_t)$
$y \leftarrow \mathsf{E}(pk, m)$
parse $y$ as $y_1 \| \cdots \| y_t$, where $y_i \in \{0,1\}^f$
for $i = 1$ to $t$ do
    $c_i \leftarrow \mathsf{sample}^{\mathcal{C}}(k, g, y_i)$
return $(c_1, \ldots, c_t)$
```

Algorithm decode proceeds analogously. From each of the $t$ symbols in the covertext, a string of $f$ bits is extracted by $g$; then the concatenation of these bit strings is decrypted by D, and the resulting value is returned (this is either an $l$-bit message or the symbol $\perp$):

```
Algorithm decode
Input: security parameter $k$, a secret key $ssk = (sk, g)$, and a covertext $(c_1, \dots, c_t) \in$
    $C^t$ to decode
Output: a decoded $l$-bit message or $\perp$
 1: for $i = 1$ to $t$ do
 2:     $y_i \leftarrow g(c_i)$
 3: $y \leftarrow y_1 \| \cdots \| y_t$
 4: $x \leftarrow \mathsf{D}(sk, y)$
 5: return $x$
```

The compatible relation $\equiv_{spk}$ of the stegosystem is computed as follows: given a pair of covertexts $(c_1, \dots, c_t)$ and $(c'_1, \dots, c'_t)$, they are first mapped to a pair of ciphertexts $y$ and $y'$, respectively, by running lines 1–3 of Algorithm decode. Then the relation is determined according to $y \equiv_{pk} y'$.

### 4.2 Analysis

This section is devoted to an analysis of the above stegosystem. Theorems 3 and 4 below together imply Theorem 2.

**Theorem 3.** *(keygen, encode, decode) is a valid stegosystem for covertext distributions with sufficiently large min-entropy.*

*Proof (Sketch).* According to Definition 1, the only non-trivial steps are to show that the algorithms are efficient and that the stegosystem is reliable, i.e., that

$$\mathsf{decode}(ssk, \mathsf{encode}(spk, m)) \;=\; m$$

for almost all pairs $(spk, ssk)$ and all $m \in \{0,1\}^l$ except with negligible probability.

Efficiency follows immediately from the construction, the assumption $f \leq \gamma \log k$, and the efficiency of the public-key cryptosystem.

For reliability, it suffices to analyze the output of encode because the decoding operation is deterministic.

Consider iteration $i$ in Algorithm encode, in which Algorithm sample tries to find a covertext $x$ that is mapped to $y_i$ by $g$. Because $g$ is chosen from a strongly 2-universal class of hash functions, the entropy smoothing theorem [14] implies that over the random choices of $g$ and $c \overset{R}{\leftarrow} \mathcal{C}$, the random variable $(g, g(c))$ is exponentially close to the uniform distribution over $f$-bit strings, provided $\mathcal{C}$ has enough min-entropy. Hence, there exists a negligible quantity $\epsilon(k) \ll 2^{-f}$ such that for almost all $g$, the distance of $g(c)$ from the uniform distribution is at most $\epsilon(k)$ over the choice $c \overset{R}{\leftarrow} \mathcal{C}$. Thus, the probability that in any particular iteration of sample, an $x$ is chosen with $g(x) \neq y_i$, is at most $1 - 2^{-f} + \epsilon(k)$.

For any such $g$, since the $k$ iterations and choices of $\mathcal{C}$ in $\mathsf{sample}$ are independent, the algorithm returns $c$ with $g(c) \neq y_i$ only with some negligible probability $\epsilon'(k)$ for $f \leq \gamma \log k$. Hence, by the union bound, the probability that any iteration of Algorithm $\mathsf{encode}$ fails to embed the correct value is at most $t\epsilon'(k)$, which is negligible.

The proof of security is based on the following result. It shows that the joint distribution of the output from Algorithm $\mathsf{sample}$ and $\mathcal{G}$ is statistically close to the joint distribution of $\mathcal{C}$ and $\mathcal{G}$, where $\mathcal{G}$ denotes the distribution of choosing $g$ uniformly from $G$, and where $\mathsf{sample}$ is run with a uniformly chosen $b$. The proof of Proposition 1 is given in the full version of the paper.

**Proposition 1.** *If the min-entropy of the covertext distribution $\mathcal{C}$ is large enough compared to $f$, then the statistical distance between $(\mathcal{S}(k), \mathcal{G})$ and $(\mathcal{C}, \mathcal{G})$ is negligible.*

**Theorem 4.** *For a covertext distribution $\mathcal{C}^t$ such that $\mathcal{C}$ has sufficiently large min-entropy and provided that $(K, E, D)$ is a PDR-CCA-secure public-key cryptosystem with pseudorandom ciphertexts, the stegosystem $(\mathsf{keygen}, \mathsf{encode}, \mathsf{decode})$ is SS-PDR-CCA.*

*Proof (Sketch).* We prove that the stegosystem $(\mathsf{keygen}, \mathsf{encode}, \mathsf{decode})$ is SS-PDR-CCA by a reduction argument. Assume that it is not SS-PDR-CCA and and hence there exists a (stego-)adversary $(SA_1, SA_2)$ that succeeds in the experiment of Definition 3 with probability $\frac{1}{2} + \delta(k)$ for some non-negligible function $\delta$. We construct an (encryption-)adversary $(A_1, A_2)$ that has black-box access to $(SA_1, SA_2)$ and breaks the PDR-CCA-security of $(\mathsf{K}, \mathsf{E}, \mathsf{D})$ as follows.

**Key generation:** When $A_1$ receives a public-key $pk$ generated by $\mathsf{K}$, it chooses $g \overset{R}{\leftarrow} G$, computes $spk \leftarrow (pk, g)$, and invokes $SA_1$ with $spk$.

**First decryption stage:** When $SA_1$ sends a query $(c_1, \ldots, c_t)$ to its decoding oracle $SO_1$, then $A_1$ computes $y \leftarrow y_1 \| \cdots \| y_t$ for $y_i \leftarrow g(c_i)$, gives $y$ to its decryption oracle $O_1$, waits for the response and forwards the response to $SA_1$.

**Challenge:** When $SA_1$ halts and outputs $(m^*, s)$, the encryption-adversary $A_1$ chooses an arbitrary plaintext message $m' \in \{0,1\}^l$, different from $m^*$, and outputs a triple $(m^*, m', g)$. According to the definition of a public-key cryptosystem, a challenge ciphertext $y^*$ is computed. Now $A_2$ is invoked with inputs $pk$, $m^*$, $m'$, $y^*$, and $g$. It parses $y^*$ as a sequence $y_1^* \| \cdots \| y_t^*$ of $f$-bit strings, computes $c_i^* \leftarrow \mathsf{sample}^{\mathcal{C}}(k, g, y_i^*)$ for $i = 1, \ldots, t$, and invokes $SA_2$ with inputs $(pk, g)$, $m^*$, $(c_1^*, \ldots, c_t^*)$, and $s$.

**Second decryption stage:** $A_2$ behaves in the same way as $A_1$ during first decryption stage: It computes a ciphertext $y$ from any decoding request that $SA_2$ makes as above, submits $y$ to the decryption oracle $O_2$, and returns the answer to $SA_2$.

**Guessing stage:** When $SA_2$ outputs a bit $b^*$, indicating its guess as to whether message $m^*$ is contained in the challenge covertext $(c_1^*, \ldots, c_t^*)$, the encryption-adversary $A_2$ returns $b^*$ as its own guess of whether $m^*$ or $m'$ is encrypted in $y^*$.

We now analyze the environment simulated by the encryption-adversary $(A_1, A_2)$ to the stego-adversary $(SA_1, SA_2)$ and the probability that the encryption-adversary can distinguish the encrypted messages.

Clearly, during key generation and the first decoding stage, the simulation for the stego-adversary $SA_1$ is perfect. During the encoding stage, a random bit $b$ is chosen according to Definition 4 and the challenge ciphertext is computed as $y^* \leftarrow \mathsf{E}(pk, m^*)$ if $b = 0$ and $y^* \leftarrow \mathsf{E}(pk, m')$ if $b = 1$.

When $b = 0$, then, according to the definition of $A_1$, the challenge covertext $c^*$ is computed in the same way as expected by the stego-adversary in the experiment of Definition 3 and the simulation is perfect.

When $b = 1$, however, $SA_2$ expects $(c_1^*, \ldots, c_t^*)$ to be a random covertext drawn according to $\mathcal{C}^t$, but receives $c_i^* = \mathsf{sample}^{\mathcal{C}}(k, g, y_i^*)$ for $i = 1, \ldots, t$ instead, where the concatenation of the $y_i^*$ is an encryption of $m'$ under key $pk$ with $\mathsf{E}$.

Proposition 1 implies that for every $i \in \{1, \ldots, t\}$, the statistical distance between $\mathcal{C}$ and the distribution of $c_i^*$ as computed by Algorithm $\mathsf{sample}$ when run with input a *uniformly chosen* $f$-bit string is bounded by a negligible quantity $\epsilon_1^*(k)$. Furthermore, since the cryptosystem $(\mathsf{K}, \mathsf{E}, \mathsf{D})$ has pseudorandom ciphertexts, for every distinguisher $SA_2$ there exists a negligible quantity $\epsilon_2^*(k)$ such that its advantage (over guessing randomly) in distinguishing between $y^*$ as used by $A_2$ and the uniform distribution on $n$-bit strings is at most $\epsilon_2^*(k)$.

By combining these two facts, it follows that the behavior of the stego-adversary $SA_2$ who observes $(c_1^*, \ldots, c_t^*)$ in the simulation when $b = 1$ does not differ from its behavior in experiment of Definition 3, where it observes covertext $\mathcal{C}^t$, with more than probability $\epsilon^*(k) = t\epsilon_1^*(k) + \epsilon_2^*(k)$.

By definition, the output of the encryption-adversary $A_2$ is the same as that of the stego-adversary $SA_2$. Since $SA_2$ succeeds with probability $\frac{1}{2} + \delta(k)$ in attacking the stegosystem and since the simulated view of $SA_2$ is correct except with probability $\epsilon^*(k)$ when $b = 1$, the probability that $SA_2$ breaks PDR-CCA-security is $\frac{1}{2} + \delta(k) - \frac{\epsilon^*(k)}{2}$, which exceeds $\frac{1}{2}$ by a non-negligible quantity and establishes the theorem.

## References

1. J. H. An, Y. Dodis, and T. Rabin, "On the security of joint signatures and encryption," in *Advances in Cryptology: EUROCRYPT 2002* (L. Knudsen, ed.), vol. 2332 of *Lecture Notes in Computer Science*, Springer, 2002.
2. R. J. Anderson and F. A. Petitcolas, "On the limits of steganography," *IEEE Journal on Selected Areas in Communications*, vol. 16, May 1998.
3. M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway, "Relations among notions of security for public-key encryption schemes," in *Advances in Cryptology: CRYPTO '98* (H. Krawczyk, ed.), vol. 1462 of *Lecture Notes in Computer Science*, Springer, 1998.
4. C. Cachin, "An information-theoretic model for steganography," *Information and Computation*, vol. 192, pp. 41–56, July 2004. Parts of this paper appeared in Proc. 2nd Workshop on Information Hiding, Springer, 1998.
5. R. Canetti, H. Krawczyk, and J. Nielsen, "Relaxing chosen-ciphertext security," in *Advances in Cryptology: CRYPTO 2003* (D. Boneh, ed.), vol. 2729 of *Lecture Notes in Computer Science*, Springer, 2003.
6. T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley, 1991.
7. R. Cramer and V. Shoup, "A practical public-key cryptosystem provably secure against adaptive chosen-ciphertext attack," in *Advances in Cryptology: CRYPTO '98* (H. Krawczyk, ed.), vol. 1462 of *Lecture Notes in Computer Science*, Springer, 1998.

8. S. Craver, "On public-key steganography in the presence of an active warden," in *Information Hiding, 2nd International Workshop* (D. Aucsmith, ed.), vol. 1525 of *Lecture Notes in Computer Science*, pp. 355–368, Springer, 1998.
9. N. Dedić, G. Itkis, L. Reyzin, and S. Russell, "Upper and lower bounds on black-box steganography," in *Proc. 2nd Theory of Cryptography Conference (TCC)* (J. Kilian, ed.), Lecture Notes in Computer Science, Springer, 2005.
10. D. Dolev, C. Dwork, and M. Naor, "Non-malleable cryptography," *SIAM Journal on Computing*, vol. 30, no. 2, pp. 391–437, 2000.
11. S. Goldwasser and S. Micali, "Probabilistic encryption," *Journal of Computer and System Sciences*, vol. 28, pp. 270–299, 1984.
12. N. J. Hopper, J. Langford, and L. von Ahn, "Provably secure steganography," in *Advances in Cryptology: CRYPTO 2002* (M. Yung, ed.), vol. 2442 of *Lecture Notes in Computer Science*, Springer, 2002.
13. Y. Lindell. Personal communication, Jan. 2004.
14. M. Luby, *Pseudorandomness and Cryptographic Applications*. Princeton University Press, 1996.
15. T. Mittelholzer, "An information-theoretic approach to steganography and watermarking," in *Information Hiding, 3rd International Workshop, IH'99* (A. Pfitzmann, ed.), vol. 1768 of *Lecture Notes in Computer Science*, pp. 1–16, Springer, 1999.
16. C. Rackoff and D. R. Simon, "Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack," in *Advances in Cryptology: CRYPTO '91* (J. Feigenbaum, ed.), vol. 576 of *Lecture Notes in Computer Science*, pp. 433–444, Springer, 1992.
17. L. Reyzin and S. Russell, "Simple stateless steganography." Cryptology ePrint Archive, Report 2003/093, 2003. `http://eprint.iacr.org/`.
18. A. Sahai, "Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security," in *Proc. 40th IEEE Symposium on Foundations of Computer Science (FOCS)*, pp. 543–553, 1999.
19. V. Shoup, "A proposal for an ISO standard for public key encryption." Cryptology ePrint Archive, Report 2001/112, 2001. `http://eprint.iacr.org/`.
20. V. Shoup, "OAEP reconsidered," *Journal of Cryptology*, vol. 15, no. 4, pp. 223–249, 2002.
21. G. J. Simmons, "The prisoners' problem and the subliminal channel," in *Advances in Cryptology: Proceedings of Crypto 83* (D. Chaum, ed.), pp. 51–67, Plenum Press, 1984.
22. L. von Ahn and N. J. Hopper, "Public-key steganography," in *Advances in Cryptology: Eurocrypt 2004* (C. Cachin and J. Camenisch, eds.), vol. 3027 of *Lecture Notes in Computer Science*, pp. 322–339, Springer, 2004.
23. M. N. Wegman and J. L. Carter, "New hash functions and their use in authentication and set equality," *Journal of Computer and System Sciences*, vol. 22, pp. 265–279, 1981.
24. J. Zöllner, H. Federrath, H. Klimant, A. Pfitzmann, R. Piotraschke, A. Westfeld, G. Wicke, and G. Wolf, "Modeling the security of steganographic systems," in *Information Hiding, 2nd International Workshop* (D. Aucsmith, ed.), vol. 1525 of *Lecture Notes in Computer Science*, pp. 344–354, Springer, 1998.