Key-Privacy in Public-Key Encryption

M. Bellare[1], A. Boldyreva[1], A. Desai[2], and D. Pointcheval[3]

[1] Dept of Computer Science & Engineering, University of California, San Diego
La Jolla, California 92093, USA.
{mihir,aboldyre}@cs.ucsd.edu
http://www-cse.ucsd.edu/users/{mihir,aboldyre}

[2] NTT Multimedia Communications Laboratories
Palo Alto, California 94306, USA.
desai@nttmcl.com
http://www-cse.ucsd.edu/users/adesai

[3] Dépt d'Informatique, ENS - CNRS
45 rue d'Ulm, 75230 Paris Cedex 05, France.
David.Pointcheval@ens.fr
http://www.di.ens.fr/users/pointche

Abstract. We consider a novel security requirement of encryption schemes that we call "key-privacy" or "anonymity". It asks that an eavesdropper in possession of a ciphertext not be able to tell which specific key, out of a set of known public keys, is the one under which the ciphertext was created, meaning the receiver is anonymous from the point of view of the adversary. We investigate the anonymity of known encryption schemes. We prove that the El Gamal scheme provides anonymity under chosen-plaintext attack assuming the Decision Diffie-Hellman problem is hard and that the Cramer-Shoup scheme provides anonymity under chosen-ciphertext attack under the same assumption. We also consider anonymity for trapdoor permutations. Known attacks indicate that the RSA trapdoor permutation is not anonymous and neither are the standard encryption schemes based on it. We provide a variant of RSA-OAEP that provides anonymity in the random oracle model assuming RSA is one-way. We also give constructions of anonymous trapdoor permutations, assuming RSA is one-way, which yield anonymous encryption schemes in the standard model.

1 Introduction

The classical security requirement of an encryption scheme is that it provide privacy of the encrypted data. Popular formalizations— such as indistinguishability (semantic security) [22] or non-malleability [15], under either chosen-plaintext or various kinds of chosen-ciphertext attacks [27, 29]— are directed at capturing various data-privacy requirements. (See [5] for a comprehensive treatment).

In this paper we consider a different (additional) security requirement of an encryption scheme which we call *key-privacy* or *anonymity*. It asks that the

encryption provide (in addition to privacy of the data being encrypted) privacy of the key under which the encryption was performed.

This might sound odd, especially in the public-key setting which is our main focus: here the key under which encryption is performed is the public key of the receiver and being public there might not seem to be anything to keep private about it. The privacy refers to the information conveyed to the adversary regarding which specific key, out of a set of known public keys, is the one under which a given ciphertext was created. We call this anonymity because it means that the receiver is anonymous from the point of view of the adversary.

Anonymity of encryption has surfaced in various different places in the past, and found several applications, as we detail later. However, it lacks a comprehensive treatment. Our goal is to provide definitions, and then systematically study popular asymmetric encryption schemes with regard to their meeting these definitions. Below we discuss our contributions and then discuss related work.

1.1 Definitions

We suggest a notion we call "indistinguishability of keys" to formalize the property of key-privacy. In the formalization, the adversary knows two public keys pk_0, pk_1, corresponding to two different entities, and gets a ciphertext C formed by encrypting some data under one of these keys. Possession of C should not give the adversary an advantage in determining under which of the two keys C was created. This can be considered under either chosen-plaintext attack or chosen-ciphertext attack, yielding two notions of security, IK-CPA and IK-CCA.

We also introduce the notion of an anonymous trapdoor permutation, which will serve as tool in some of the designs.

1.2 The search for anonymous asymmetric encryption schemes

In a heterogenous public-key environment, encryption will probably fail to be anonymous for trivial reasons. For example, different users might be using different cryptosytems, or, if the same cryptosystem, have keys of different lengths. (If one possible recipient has a RSA public key with a 1024 bit modulus and the other a RSA public key with a 512 bit modulus, the length of the RSA ciphertext will immediately enable an eavesdropper to know for which recipient the ciphertext is intended.) We can however hope for anonymity in a context where all users use the same security parameter or global parameters. We will look at specific systems with this restriction in mind.

Ideally, we would like to be able to prove that popular, existing and practical encryption schemes have the anonymity property (rather than having to design new schemes.) This would be convenient because then existing encryption-using protocols or software would not have to be altered in order for them to have the anonymity guarantees conferred by those of the encryption scheme. Accordingly, we begin by examining existing schemes. We will consider discrete log based schemes such as El Gamal and Cramer-Shoup, and also RSA-based schemes such as RSA-OAEP.

It is easy to see that an encryption scheme could meet even the strongest notion of data-privacy— namely indistinguishability under chosen-ciphertext attack— yet not provide key-privacy. (The ciphertext could contain the public key.) Accordingly, existing results about data-privacy of asymmetric encryption schemes are not directly applicable. Existing schemes must be re-analyzed with regard to key-privacy.

In approaching this problem, we had no a priori way to predict whether or not a given asymmetric scheme would have the key-privacy property, and, if it did, whether the proof would be a simple modification of the known data privacy proof, or require new techniques. It is only by doing the work that one can tell what is involved.

We found that the above-mentioned discrete log based schemes did have the key-privacy property, and, moreover, that it was possible to prove this, under the same assumptions as used to prove data-privacy, by following the outline of the proofs of data-privacy with appropriate modifications. This perhaps unexpected strength of the discrete log based world (meaning not only the presence of the added security property in the popular schemes, but the fact that the existing techniques are strong enough to lead to a proof) seems important to highlight. In contrast, folklore attacks already rule out key-privacy for standard RSA-based schemes. Accordingly, we provide variants that have the property. Let us now look at these results in more detail.

1.3 Discrete log based schemes

The El Gamal cryptosystem over a group of prime order provably provides data-privacy under chosen-plaintext attack assuming the DDH (Decision Diffie-Hellman) problem is hard in the group [25, 12, 33, 3]. Let us now consider a system of users all of which work over the same group. (To be concrete, let q be a prime such that $2q+1$ is also prime, let G_q be the order q subgroup of quadratic residues of Z^*_{2q+1} and let $g \in G_q$ be a generator of G_q. Then q, g are system wide parameters based on which all users choose keys.) In this setting we prove that the El Gamal scheme meets the notion of IK-CPA under the same assumption used to establish data-privacy, namely the hardness of the DDH problem in the group. Thus the El Gamal scheme provably provides anonymity. Our proof exploits self-reducibility properties of the DDH problem together with ideas from the proof of data-privacy.

The Cramer-Shoup scheme [12] is proven to provide data-privacy under chosen-ciphertext attack, under the assumption that the DDH problem is hard in the group underlying the scheme. Let us again consider a system of users, all of which work over the same group, and for concreteness let it be the group G_q that we considered above. In this setting we prove that the Cramer-Shoup scheme meets the notion of IK-CCA assuming the DDH problem is hard in G_q. Our proof exploits ideas in [12, 3].

1.4 RSA-based schemes

A simple observation that seems to be folkore is that standard RSA encryption does not provide anonymity, even when all modulii in the system have the same length. In all popular schemes, the ciphertext is (or contains) an element $y = x^e \bmod N$ where x is a random member of Z_N^*. Suppose an adversary knows that the ciphertext is created under one of two keys N_0, e_0 or N_1, e_1, and suppose $N_0 \leq N_1$. If $y \geq N_0$ then the adversary bets it was created under N_1, e_1, else it bets it was created under N_0, e_0. It is not hard to see that this attack has non-negligible advantage.

One approach to anonymizing RSA, suggested by Desmedt [14], is to add random multiples of the modulus N to the ciphertext. This seems to overcome the above attack, at least when the data encrypted is random, but results in a doubling of the length of the ciphertext. We look at a few other approaches.

We consider an RSA-based encryption scheme popular in current practice, namely RSA-OAEP [8]. (It is the PKCS v2.0 standard [28], proved secure against chosen-ciphertext attack in the random oracle model [18].) We suggest a variant which we can prove is anonymous. Recall that OAEP is a randomized (invertible) transform that on input a message M picks a random string r and, using some public hash functions, produces a point $x = \mathsf{OAEP}(r, M) \in Z_N^*$ where N, e is the public key of the receiver. The ciphertext is then $y = x^e \bmod N$. Our variant simply repeats the ciphertext computation, each time using new coins, until the ciphertext y satisfies $1 \leq y \leq 2^{k-2}$, where k is the length of N. We prove that this scheme meets the notion of IK-CCA in the random oracle model assuming RSA is a one-way function. (Data-privacy under chosen-ciphertext attack must be re-proved, but this can be done, under the same assumption, following [18].) The expected number of exponentiations for encryption being two, encryption in our variant is about twice as expensive as for RSA-OAEP itself, but this may be tolerable when the encryption exponent is small. The cost of decryption is the same as for RSA-OAEP itself, namely one exponentiation with the decryption exponent. As compared to Desmedt's scheme, the size of the ciphertext increases by only one bit rather than doubling. Our proof exploits the framework and techniques of [18, 8].

1.5 Trapdoor permutation based schemes

We then ask a more theoretical, or foundational, question, namely whether there exists an encryption scheme that can be proven to provide key-privacy based only on the assumption that RSA is one-way, meaning without making use of the random oracle model. To answer this we return to the classical techniques based on hardcore bits. We define a notion of anonymity for trapdoor permutations. We note that the above attack implies that RSA is not an anonymous trapdoor permutation, but we then design some trapdoor permutations which are anonymous and one-way as long as RSA is one-way. Appealing to known results about hardcore bits then yields an encryption scheme whose anonymity is proven based solely on the one-wayness of RSA. The computational costs of this approach, however, prohibit its being useful in practice.

1.6 Applications and Related work

In recent years, anonymous encryption has arisen in the context of mobile communications. Consider a mobile user A, communicating over a wireless network with some entity B. The latter is sending A ciphertexts encrypted under A's public key. A common case is that B is a base station. A wants to keep her identity private from an eavesdropping adversary. In this case A will be a member of some set of users whose identities and public keys are possibly known to the adversary. The adversary will also be able to see the ciphertexts sent by B to A. If the scheme is anonymous, however, the adversary will be unable to determine A's identity. A particular case of this is anonymous authenticated key exchange, where the communication between roaming user A and base station B is for the purpose of authentication and distribution of a session key based on the parties public keys, but the identity of A should remain unknown to an eavesdropper. Anonymity is targeted in authenticated key exchange protocols such as SKEME [23]. The author notes that a requirement for SKEME to provide anonymous authenticated key exchange is that the public-key encryption scheme used to encrypt under A's public key must have the key-privacy property.

In independent and concurrent work, Camenisch and Lysyanskaya [10] consider anonymous credential systems. Such a sytem enables users to control the dissemination of information about themselves. It is required that it be infeasible to correlate transactions carried out by the same user. The solution to this given in [10] makes use of a *verifiable circular* encryption scheme that needs to have the key-privacy property. They provide a notion similar to ours, but in the context of verifiable encryption. They observe that their variant of the El Gamal scheme is anonymous under chosen-plaintext attack.

Sako [30] considers the problem of achieving bid secrecy and verifiability in auction protocols. Their approach is to express each bid as an encryption of a known message, with the *key* to encrypt it corresponding to the value of the bid. Thus, what needs to be hidden is not the message that is encrypted, but the key used to encrypt it. The bid itself can be identified by finding the corresponding decrypting key that successfully decrypts to the given message. Unlike the previous examples, where the key-privacy property was needed to protect identities, this application shows how that property can be exploited to satisfy a secrecy requirement. Sako also considered a notion similar to ours and gave a variant of the El Gamal scheme that was expected to be secure in that sense.

Formal notions of key-privacy have appeared in the context of symmetric encryption [1, 13, 17]. Abadi and Rogaway [1] show that popular modes of operation of block ciphers, such as CBC, provide key-privacy if the block cipher is a pseudorandom permutation.

The notion given by Desai [13], like ours, is concerned with the privacy of keys. However, the goal, model and setting in which it is considered differs from ours— the goal there is to capture a security property for block cipher based encryption schemes that implies that exhaustive key-search on them is slowed down proportional to the size of the ciphertext. There is, however, a similarity

between our definitions (suitably adapted to the symmetric setting) and those of Abadi and Rogaway [1] and Fischlin [17]. Although the exact formalizations differ, it is not hard to see that there is an equivalence between the three for chosen-plaintext attack.

Chosen-ciphertext attacks do not seem to have been considered before in the context of key-privacy. In fact, Fischlin [17] observes that giving decryption oracles to the adversary in their setting makes its task trivial. However, in our formalization chosen-ciphertext attacks can be modeled by giving decryption oracles and then putting an appropriate restriction on their use. The restriction is the most natural and is anyway in effect for modeling semantic security against chosen-ciphertext attack. This allows us to make a distinction between those encryption schemes that are anonymous under chosen-ciphertext attack, such as Cramer-Shoup, and those that are not, such as El Gamal— just as there are schemes that are semantically secure under chosen-plaintext attack but not under chosen-ciphertext attack.

2 Notions of Key-Privacy

The notions of security typically considered for encryption schemes are "indistinguishability of encryptions under chosen-plaintext attack" [22] and "indistinguishability of encryptions under adaptive chosen-ciphertext attack" [29]. The former is usually denoted IND-CPA, but is denoted IE-CCA in this paper to emphasize that it is about encryptions, not keys. Similarly, the latter notion is usually denoted IND-CCA (or IND-CCA2), but is denoted IE-CCA in this paper. It is well-known that these capture strong data-privacy properties. However, they do not guarantee that some partial information about the underlying *key* is not leaked. Indeed, in a public-key encryption scheme, the entire public-key could be made an explicit part of the ciphertext and yet the scheme could meet the above-mentioned data-privacy notions. We want to make a distinction between such schemes and those that do not leak information about the underlying key. As noted earlier, schemes of the latter kind are necessary if the anonymity of receivers is a concern.

We are interested in formalizing the inability of an adversary, given a challenge ciphertext, to learn any information about the underlying plaintext or key. It is not hard to see that the goals of data-privacy and key-privacy are orthogonal. We recognize that existing encryption schemes are likely to have already been investigated with respect to their data-privacy security properties. Hence it is useful, from a practical point of view, to isolate the key-privacy requirements from the data-privacy ones. We do this in the form of two notions: "indistinguishability of keys under chosen-plaintext attack" (IK-CPA) and "indistinguishability of keys under adaptive chosen-ciphertext attack" (IK-CCA). We begin with a syntax for public-key encryption schemes, divorcing syntax from formal notions of security.

2.1 Syntax

The syntax of an encryption scheme specifies what algorithms make it up. We augment the usual formalization in order to better model practice, where users may share some fixed "global" information.

A *public-key encryption scheme* $\mathcal{PE} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ consists of four algorithms. The *common-key generation* algorithm $\mathcal{G}$ takes as input some security parameter k and returns some common key I. (Here I may be just a security parameter k, or include some additional information. For example in a Diffie-Hellman based scheme, I might include, in addition to k, a global prime number and generator of a group which all parties use to create their keys.) The *key generation* algorithm $\mathcal{K}$ is a randomized algorithm that takes as input the common key I and returns a pair (pk, sk) of keys, the public key and a matching secret key, respectively; we write $(pk, sk) \stackrel{R}{\leftarrow} \mathcal{K}(I)$. The *encryption* algorithm $\mathcal{E}$ is a randomized algorithm that takes the public key pk and a *plaintext* x to return a *ciphertext* y; we write $y \leftarrow \mathcal{E}_{pk}(x)$. The *decryption* algorithm $\mathcal{D}$ is a deterministic algorithm that takes the secret key sk and a ciphertext y to return the corresponding plaintext x or a special symbol $\perp$ to indicate that the ciphertext was invalid; we write $x \leftarrow \mathcal{D}_{sk}(y)$ when y is valid and $\perp \leftarrow \mathcal{D}_{sk}(y)$ otherwise. Associated to each public key pk is a *message space* $\mathrm{MsgSp}(pk)$ from which x is allowed to be drawn. We require that $\mathcal{D}_{sk}(\mathcal{E}_{pk}(x)) = x$ for all $x \in \mathrm{MsgSp}(pk)$.

2.2 Indistinguishability of Keys

We give notions of key-privacy under chosen-plaintext and chosen-ciphertext attacks. We think of an adversary running in two stages. In the find stage it takes two public keys pk_0 and pk_1 (corresponding to secret keys sk_0 and sk_1, respectively) and outputs a message x together with some state information s. In the guess stage it gets a challenge ciphertext y formed by encrypting at random the messages under one of the two keys, and must say which key was chosen. In the case of a chosen-ciphertext attack the adversary gets oracles for $\mathcal{D}_{sk_0}(\cdot)$ and $\mathcal{D}_{sk_1}(\cdot)$ and is allowed to invoke them on any point with the restriction (on both oracles) of not querying y during the guess stage.

Definition 1. [IK-CPA, IK-CCA] *Let $\mathcal{PE} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. Let $b \in \{0,1\}$ and $k \in \mathsf{N}$. Let $A_{\mathrm{cpa}}, A_{\mathrm{cca}}$ be adversaries that run in two stages and where A_{cca} has access to the oracles $\mathcal{D}_{sk_0}(\cdot)$ and $\mathcal{D}_{sk_1}(\cdot)$. Now, we consider the following experiments:*

Experiment $\mathbf{Exp}^{\text{ik-cpa-}b}_{\mathcal{PE},A_{\mathrm{cpa}}}(k)$
- $I \stackrel{R}{\leftarrow} \mathcal{G}(k)$
- $(pk_0, sk_0) \stackrel{R}{\leftarrow} \mathcal{K}(I)$; $(pk_1, sk_1) \stackrel{R}{\leftarrow} \mathcal{K}(I)$
- $(x, s) \leftarrow A_{\mathrm{cpa}}(\mathsf{find}, pk_0, pk_1)$
- $y \leftarrow \mathcal{E}_{pk_b}(x)$
- $d \leftarrow A_{\mathrm{cpa}}(\mathsf{guess}, y, s)$
- *Return* d

Experiment $\mathbf{Exp}^{\text{ik-cca-}b}_{\mathcal{PE},A_{\mathrm{cca}}}(k)$
- $I \stackrel{R}{\leftarrow} \mathcal{G}(k)$
- $(pk_0, sk_0) \stackrel{R}{\leftarrow} \mathcal{K}(I)$; $(pk_1, sk_1) \stackrel{R}{\leftarrow} \mathcal{K}(I)$
- $(x, s) \leftarrow A_{\mathrm{cca}}^{\mathcal{D}_{sk_0}(\cdot),\mathcal{D}_{sk_1}(\cdot)}(\mathsf{find}, pk_0, pk_1)$
- $y \leftarrow \mathcal{E}_{pk_b}(x)$
- $d \leftarrow A_{\mathrm{cca}}^{\mathcal{D}_{sk_0}(\cdot),\mathcal{D}_{sk_1}(\cdot)}(\mathsf{guess}, y, s)$
- *Return* d

Above it is mandated that A_{cca} *never queries* $\mathcal{D}_{sk_0}(\cdot)$ *or* $\mathcal{D}_{sk_1}(\cdot)$ *on the challenge ciphertext* y*. For* $\mathrm{atk} \in \{\mathrm{cpa}, \mathrm{cca}\}$ *we define the advantages of the adversaries via*

$$\mathbf{Adv}^{\text{ik-atk}}_{\mathcal{PE},A_{\mathrm{atk}}}(k) = \Pr[\,\mathbf{Exp}^{\text{ik-atk-1}}_{\mathcal{PE},A_{\mathrm{atk}}}(k) = 1\,] - \Pr[\,\mathbf{Exp}^{\text{ik-atk-0}}_{\mathcal{PE},A_{\mathrm{atk}}}(k) = 1\,]\,.$$

The scheme $\mathcal{PE}$ *is said to be IK-CPA secure (respectively IK-CCA secure) if the function* $\mathbf{Adv}^{\text{ik-cpa}}_{\mathcal{PE},A}(\cdot)$ *(resp.* $\mathbf{Adv}^{\text{ik-cca}}_{\mathcal{PE},A}(\cdot)$*) is negligible for any adversary* A *whose time complexity is polynomial in* k*.* ∎

The "time-complexity" is the worst case execution time of the experiment plus the size of the code of the adversary, in some fixed RAM model of computation. (Note that the execution time refers to the entire experiment, not just the adversary. In particular, it includes the time for key generation, challenge generation, and computation of responses to oracle queries if any.) The same convention is used for all other definitions in this paper and will not be explicitly mentioned again.

2.3 Anonymous one-way functions

A *family of functions* $F = (K, S, E)$ is specified by three algorithms. The randomized *key-generation* algorithm K takes input the security parameter $k \in \mathsf{N}$ and returns a pair (pk, sk) where pk is a public key, and sk is an associated secret key. (In cases where the family is not trapdoor, the secret key is simply the empty string.) The randomized *sampling* algorithm S takes input pk and returns a random point in a set that we call the domain of pk and denote $\mathrm{Dom}_F(pk)$. We usually omit explicit mention of the sampling algorithm and just write $x \xleftarrow{R} \mathrm{Dom}_F(pk)$. The deterministic *evaluation* algorithm E takes input pk and a point $x \in \mathrm{Dom}_F(pk)$ and returns an output we denote by $E_{pk}(x)$. We let $\mathrm{Rng}_F(pk) = \{\, E_{pk}(x) : x \in \mathrm{Dom}_F(pk) \,\}$ denote the range of the function $E_{pk}(\cdot)$. We say that F is a family of *trapdoor* functions if there exists a deterministic *inversion* algorithm I that takes input sk and a point $y \in \mathrm{Rng}_F(pk)$ and returns a point $x \in \mathrm{Dom}_F(pk)$ such that $E_{pk}(x) = y$. We say that F is a family of *permutations* if $\mathrm{Dom}_F(pk) = \mathrm{Rng}_F(pk)$ and E_{pk} is a permutation on this set.

Definition 2. *Let* $F = (K, S, E)$ *be a family of functions. Let* $b \in \{0, 1\}$ *and* $k \in \mathsf{N}$ *be a security parameter. Let* $0 < \theta \leq 1$ *be a constant. Let* A, B *be adversaries. Now, we consider the following experiments:*

Experiment $\mathbf{Exp}^{\theta\text{-pow-fnc}}_{F,B}(k)$
- $(pk, sk) \xleftarrow{R} K(k)$
- $x_1 \| x_2 \xleftarrow{R} \mathrm{Dom}_F(pk)$ *where* $|x_1| = \lceil \theta \cdot |(x_1 \| x_2)| \rceil$
- $y \leftarrow E_{pk}(x_1 \| x_2)$
- $x_1' \leftarrow B(pk, y)$ *where* $|x_1'| = |x_1|$
- *For any* x_2' *if* $E_{pk}(x_1' \| x_2') = y$ *then return 1*
- *Else return 0*

Experiment $\mathbf{Exp}^{\text{ik-fnc-}b}_{F,A}(k)$
- $(pk_0, sk_0) \xleftarrow{R} K(k)$
- $(pk_1, sk_1) \xleftarrow{R} K(k)$
- $x \xleftarrow{R} \mathrm{Dom}_F(pk_b)$
- $y \leftarrow E_{pk_b}(x)$
- $d \leftarrow A(pk_0, pk_1, y)$
- *Return* d

We define the advantages of the adversaries via

$$\mathbf{Adv}^{\theta\text{-pow-fnc}}_{F,B}(k) = \Pr[\,\mathbf{Exp}^{\theta\text{-pow-fnc}}_{F,B}(k) = 1\,]$$
$$\mathbf{Adv}^{\text{ik-fnc}}_{F,A}(k) = \Pr[\,\mathbf{Exp}^{\text{ik-fnc-1}}_{F,A}(k) = 1\,] - \Pr[\,\mathbf{Exp}^{\text{ik-fnc-0}}_{F,A}(k) = 1\,]\,.$$

The family F is said to be θ-partial one-way if the function $\mathbf{Adv}^{\theta\text{-pow-fnc}}_{F,B}(\cdot)$ is negligible for any adversary B whose time complexity is polynomial in k. The family F is said to be anonymous if the function $\mathbf{Adv}^{\text{ik-fnc}}_{F,A}(\cdot)$ is negligible for any adversary A whose time complexity is polynomial in k. The family F is said to be perfectly anonymous if $\mathbf{Adv}^{\text{ik-fnc}}_{F,A}(k) = 0$ for every k and every adversary A. ∎

Note that when $\theta = 1$ the notion of θ-partial one-wayness coincides with the standard notion of one-wayness. As the above indicates, we expect that information-theoretic anonymity is possible for one-way functions, even though not for encryption schemes.

3 Anonymity of DDH-based schemes

The DDH-based schemes we consider work over a group of prime order. This could be a subgroup of order q of Z_p^* where p, q are primes such that q divides $p-1$. It could also be an elliptic curve group of prime order. For concreteness our description is for the first case. Specifically if q is a prime such that $2q+1$ is also prime we let G_q be the subgroup of quadratic residues of Z_p^*. It has order q. A *prime-order-group generator* is a probabilistic algorithm that on input the security parameter k returns a pair (q, g) satisfying the following conditions: q is a prime with $2^{k-1} < q < 2^k$; $2q+1$ is a prime; and g is a generator of G_q. (There are numerous possible specific prime-order-group generators.) We will relate the anonymity of the El Gamal and Cramer-Shoup schemes to the hardness of the DDH problem for appropriate prime-order-group generators. Accordingly we next summarize definitions for the latter.

Definition 3. [DDH] *Let $\mathcal{G}$ be a prime-order-group generator. Let D be an adversary that on input q, g and three elements $X, Y, T \in G_q$ returns a bit. We consider the following experiments*

Experiment $\mathbf{Exp}^{\text{ddh-real}}_{\mathcal{G},D}(k)$	*Experiment* $\mathbf{Exp}^{\text{ddh-rand}}_{\mathcal{G},D}(k)$
$(q, g) \xleftarrow{R} \mathcal{G}(k)$	$(q, g) \xleftarrow{R} \mathcal{G}(k)$
$x \xleftarrow{R} Z_q\,;\ X \leftarrow g^x$	$x \xleftarrow{R} Z_q\,;\ X \leftarrow g^x$
$y \xleftarrow{R} Z_q\,;\ Y \leftarrow g^y$	$y \xleftarrow{R} Z_q\,;\ Y \leftarrow g^y$
$T \leftarrow g^{xy}$	$T \xleftarrow{R} G_q$
$d \leftarrow D(q, g, X, Y, T)$	$d \leftarrow D(q, g, X, Y, T)$
Return d	*Return* d

The advantage of D in solving the Decisional Diffie-Hellman (DDH) problem for $\mathcal{G}$ is the function of the security parameter defined by

$$\mathbf{Adv}^{\text{ddh}}_{\mathcal{G},D}(k) \;=\; \Pr[\,\mathbf{Exp}^{\text{ddh-real}}_{\mathcal{G},D}(k) = 1\,] - \Pr[\,\mathbf{Exp}^{\text{ddh-rand}}_{\mathcal{G},D}(k) = 1\,]\,.$$

We say that the DDH problem is hard for $\mathcal{G}$ if the function $\mathbf{Adv}^{\mathrm{ddh}}_{\mathcal{G},D}(\cdot)$ is negligible for every algorithm D whose time-complexity is polynomial in k. ∎

3.1 El Gamal

The El Gamal scheme in a group of prime order is known to meet the notion of indistinguishability under chosen-plaintext attack under the assumption that the decision Diffie-Hellman (DDH) problem is hard. (This is noted in [25, 12] and fully treated in [33]). We want to look at the anonymity of the El Gamal encryption scheme under chosen-plaintext attack.

Let $\mathcal{G}$ be a prime-order-group generator. This is the common key generation algorithm of the associated scheme $\mathcal{EG} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, the rest of whose algorithms are as follows:

Algorithm $\mathcal{K}(q, g)$	Algorithm $\mathcal{E}_{pk}(M)$	Algorithm $\mathcal{D}_{sk}(Y, W)$
$x \xleftarrow{R} Z_q$	$y \xleftarrow{R} Z_q$	$T \leftarrow Y^x$
$X \leftarrow g^x$	$Y \leftarrow g^y$	$M \leftarrow WT^{-1}$
$pk \leftarrow (q, g, X)$	$T \leftarrow X^y$	Return M
$sk \leftarrow (q, g, x)$	$W \leftarrow TM$	
Return (pk, sk)	Return (Y, W)	

The message space associated to a public key (q, g, X) is the group G_q itself, with the understanding that all messages from G_q are properly encoded as strings of some common length whenever appropriate. Note that a generator g is the output of the common key generation algorithm, which means we fix g for all keys. We do it only for a simplicity reason and will show that all our results hold also for a case when each key uses a random generator g.

We now analyze the anonymity of the El Gamal scheme under chosen-plaintext attack.

Theorem 1. *Let $\mathcal{G}$ be a prime-order-group generator. If the DDH problem is hard for $\mathcal{G}$ then the associated El Gamal scheme $\mathcal{EG}$ is IK-CPA secure. Concretely, for any adversary A there exists a distinguisher D such that for any k*

$$\mathbf{Adv}^{\text{ik-cpa}}_{\mathcal{EG},A}(k) \;\leq\; 2\mathbf{Adv}^{\mathrm{ddh}}_{\mathcal{G},D}(k) + \frac{1}{2^{k-2}}$$

and the running time of D is that of A plus $O(k^3)$. ∎

The proof of the above is in the full version of this paper [2].

3.2 Cramer-Shoup

The El Gamal scheme provides data privacy and anonymity against chosen-plaintext attack. We now consider the Cramer-Shoup scheme [12] in order to obtain the same security properties under chosen-ciphertext attack. We will use collision-resistant hash functions so we begin by recalling what we need.

A family of hash functions $\mathcal{H} = (\mathcal{GH}, \mathcal{EH})$ is defined by a probabilistic generator algorithm $\mathcal{GH}$ —which takes as input the security parameter k and returns a key K— and a deterministic evaluation algorithm $\mathcal{EH}$ —which takes as input the key K and a string $M \in \{0,1\}^*$ and returns a string $\mathcal{EH}_K(M) \in \{0,1\}^{k-1}$.

Definition 4. *Let $\mathcal{H} = (\mathcal{GH}, \mathcal{EH})$ be a family of hash functions and let C be an adversary that on input a key K returns two strings. Now, we consider the following experiment:*

Experiment $\mathbf{Exp}^{\mathrm{cr}}_{\mathcal{H},C}(k)$
$K \xleftarrow{R} \mathcal{GH}(k)\,;\ (x_0, x_1) \leftarrow C(K)$
If $(x_0 \neq x_1)$ *and* $\mathcal{EH}_K(x_0) = \mathcal{EH}_K(x_1)$ *then return 1 else return 0*

We define the advantage of adversary C via

$$\mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H},C}(k) = \Pr[\,\mathbf{Exp}^{\mathrm{cr}}_{\mathcal{H},C}(k) = 1\,]\,.$$

We say that the family of hash functions $\mathcal{H}$ is collision-resistant if $\mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H},C}(\cdot)$ is negligible for every algorithm C whose time-complexity is polynomial in k. ∎

Let $\overline{\mathcal{G}}$ be a prime-order-group generator. The common key generation algorithm of the associated Cramer-Shoup scheme $\mathcal{CS} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ is:

$$\text{Algorithm } \mathcal{G}(k):\ (q, g_1) \xleftarrow{R} \overline{\mathcal{G}};\ g_2 \xleftarrow{R} G_q;\ K \xleftarrow{R} \mathcal{GH}(k);\ \ \text{Return } (q, g_1, g_2, K).$$

The rest of algorithms are specified as follows:

Algorithm $\mathcal{K}(q, g_1, g_2, K)$	Algorithm $\mathcal{E}_{pk}(M)$	Algorithm $\mathcal{D}_{sk}(u_1, u_2, e, v)$
$g_1 \leftarrow g$	$r \xleftarrow{R} Z_q$	$\alpha \leftarrow \mathcal{EH}_K(u_1, u_2, e)$
$x_1, x_2, y_1, y_2, z \xleftarrow{R} Z_q$	$u_1 \leftarrow g_1^r\ ;\ u_2 \leftarrow g_2^r$	If ${u_1}^{x_1+y_1\alpha}{u_2}^{x_2+y_2\alpha} = v$
$c \leftarrow g_1^{x_1} g_2^{x_2}\ ;\ d \leftarrow g_1^{y_1} g_2^{y_2}$	$e \leftarrow h^r M$	then $M \leftarrow e/{u_1}^z$
$h \leftarrow g_1^z$	$\alpha \leftarrow \mathcal{EH}_K(u_1, u_2, e)$	else $M \leftarrow \perp$
$pk \leftarrow (g_1, g_2, c, d, h, K)$	$v \leftarrow c^r d^{r\alpha}$	Return M
$sk \leftarrow (x_1, x_2, y_1, y_2, z)$	Return (u_1, u_2, e, v)	
Return (pk, sk)		

The message space is the group G_q. Note that the range of the hash function $\mathcal{EH}_K$ is $\{0,1\}^{k-1}$ which we identify with $\{0, \ldots, 2^{k-1}\}$. Since $q > 2^{k-1}$ this is a subset of Z_q. Again for simplicity we assume that g_1, g_2 are fixed for all keys but we will show that our results hold even if g_1, g_2 are chosen at random for all keys.

We now analyze the anonymity of $\mathcal{CS}$ under chosen-ciphertext attack.

Theorem 2. *Let $\overline{\mathcal{G}}$ be a prime-order-group generator and let $\mathcal{CS}$ be the associated Cramer-Shoup scheme. If the DDH problem is hard for $\overline{\mathcal{G}}$ then $\mathcal{CS}$ is anonymous in the sense of IK-CCA. Concretely, for any adversary A attacking the anonymity of $\mathcal{CS}$ under a chosen-ciphertext attack and making in total $q_{\mathrm{dec}}(\cdot)$ decryption oracle queries, there exists a distinguisher D for DDH and an adversary C attacking the collision-resistance of $\mathcal{H}$ such that*

$$\mathbf{Adv}^{\text{ik-cca}}_{\mathcal{CS},A}(k) \ \leq\ 2\mathbf{Adv}^{\mathrm{ddh}}_{\overline{\mathcal{G}},D}(k) + 2\mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H},C}(k) + \frac{q_{\mathrm{dec}}(k)+2}{2^{k-3}}.$$

and the running time of D and C is that of A plus $O(k^3)$. ∎

The proof of the above is in the full version of this paper [2]. Note that security of the Cramer-Shoup scheme in the IE-CCA sense has been proven in [12] using a weaker assumption on the hash function $\mathcal{H}$ than the one we have here. They do not require that $\mathcal{H}$ be collision-resistant, as we do, but only that it be a universal one-way family of hash functions (UOWHF) [26]. We have at this time not determined if the scheme can also be proven secure in the IK-CCA sense assuming $\mathcal{H}$ to be a UOWHF.

4 Anonymity of RSA-based schemes

The attack on RSA mentioned in Section 1 implies that the RSA family of trapdoor permutations is not anonymous. This means that all traditional RSA-based encryption schemes are not anonymous. We provide several ways to implement anonymous RSA-based encryption. First we take a direct approach, specifying an anonymous RSA-OAEP variant based on repetition and proving it secure in the random oracle model. Then we show how to construct anonymous trapdoor permutation families based on RSA and derive anonymous RSA-based encryption schemes from them. In particular, the latter leads to anonymous encryption schemes whose proofs of security are in the standard rather than the random oracle model. We begin with a description of the RSA family of trapdoor permutations we will use in this section. See Section 2 for notions of security for families of trapdoor permutations.

Example 1. The specifications of the *standard RSA family* of trapdoor permutations $\mathsf{RSA} = (K, S, E)$ are as follows. The key generation algorithm takes as input a security parameter k and picks random, distinct primes p, q in the range $2^{k/2-1} < p, q < 2^{k/2}$. (If k is odd, increment it by 1 before picking the primes.) It sets $N = pq$. It picks $e, d \in Z^*_{\varphi(N)}$ such that $ed \equiv 1 \pmod{\varphi(N)}$ where $\varphi(N) = (p-1)(q-1)$. The public key is N, e and the secret key is N, d. The sets $\mathrm{Dom}_{\mathsf{RSA}}(N, e)$ and $\mathrm{Rng}_{\mathsf{RSA}}(N, e)$ are both equal to Z^*_N. The evaluation algorithm is $E_{N,e}(x) = x^e \bmod N$ and the inversion algorithm is $I_{N,d}(y) = y^d \bmod N$. The sampling algorithm returns a random point in Z^*_N. ∎

The anonymity attack on RSA carries over to most encryption schemes based on it, including the most popular one, RSA-OAEP. We next describe a variant of RSA-OAEP that preserves its data-privacy properties but is in addition anonymous.

4.1 Anonymous variant of RSA-OAEP

The original scheme and our variant are described in the random-oracle (RO) model [7]. All the notions of security, defined earlier, can be "lifted" to the RO setting in a straightforward manner. To modify the definitions, begin the experiment defining advantage by choosing random functions G and H, each from the set of all functions from some appropriate domain to appropriate range.

Then provide a G-oracle and H-oracle to the adversaries, and allow that $\mathcal{E}_{pk}$ and $\mathcal{D}_{sk}$ may depend on G and H (which we write as $\mathcal{E}_{pk}^{G,H}$ and $\mathcal{D}_{sk}^{G,H}$).

The idea behind our variant is to repeat the standard encryption procedure under RSA-OAEP, until the ciphertext falls in some "safe" range. We refer to our scheme as RSA-RAEP (for *repeated* asymmetric encryption with padding). More concretely, for $\mathsf{RSA} = (K, S, E)$, our scheme RSA-RAEP $= (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ is as follows. The common key generator algorithm $\mathcal{G}$ takes a security parameter k and returns parameters k, k_0 and k_1 such that $k_0(k) + k_1(k) < k$ for all $k > 1$. This defines an associated plaintext-length function $n(k) = k - k_0(k) - k_1(k)$. The key generation algorithm $\mathcal{K}$ takes k, k_0, k_1 and runs the key-generation algorithm of the RSA family, namely K on k to get a public key (N, e) and secret key (N, d) (see Example 1). The public key for the scheme pk is $(N, e), k, k_0, k_1$ and the secret key sk is $(N, d), k, k_0, k_1$. The other algorithms are depicted below. The oracles G and H which $\mathcal{E}_{pk}$ and $\mathcal{D}_{sk}$ reference below map bit strings as follows: $G: \{0,1\}^{k_0} \mapsto \{0,1\}^{n+k_1}$ and $H: \{0,1\}^{n+k_1} \mapsto \{0,1\}^{k_0}$.

Algorithm $\mathcal{E}_{pk}^{G,H}(x)$
- $ctr = -1$
- Repeat
  - $ctr \leftarrow ctr + 1$
  - $r \xleftarrow{R} \{0,1\}^{k_0}$
  - $s \leftarrow (x \| 0^{k_1}) \oplus G(r)$
  - $t \leftarrow r \oplus H(s)$
  - $v \leftarrow (s\|t)^e \bmod N$
- Until $(v < 2^{k-2}) \vee (ctr = k_1)$
- If $ctr = k_1$ then $y \leftarrow 1 \| 0^{k_0+k_1} \| x$
- Else $y \leftarrow 0 \| v$
- Return y

Algorithm $\mathcal{D}_{sk}^{G,H}(y)$
- Parse y as $b\|v$ where b is a bit
- If $b = 1$ then parse v as $w\|x$ where $|x| = n$
  - If $w = 0^{k_0+k_1}$ then $z \leftarrow x$
  - Else (if $w \neq 0^{k_0+k_1}$) $z \leftarrow \perp$
- Else (if $b = 0$)
  - $(s\|t) \leftarrow v^d \bmod N$ where:
    - $|s| = k_1 + n$ and $|t| = k_0$
  - $r \leftarrow t \oplus H(s)$
  - $(x\|p) \leftarrow s \oplus G(r)$ where:
    - $|x| = n$ and $|p| = k_1$
  - If $p = 0^{k_1}$ then $z \leftarrow x$
  - Else $z \leftarrow \perp$
- Return z

Note that the valid ciphertexts under RSA-OAEP are (uniformly) distributed in $\mathrm{Rng}_{\mathsf{RSA}}(N, e)$, which is Z_N^*. Under RSA-RAEP, valid ciphertexts take the form $0\|v$ where $v \in (Z_N^* \cap [1, 2^{k-2}])$. The expected running time of this scheme is approximately twice that of RSA-OAEP (and k_1 times more, in the worst case). The ciphertext is longer by one bit. However, unlike RSA-OAEP, this scheme turns out to be IK-CCA secure. The (data-privacy) security of RSA-OAEP under CCA has already been established [18]. It is not hard to see that this result holds for RSA-RAEP as well. We omit the (simple) proof of this, noting only that the security (relative to RSA-OAEP) degrades roughly by the probability that after k_1 repetitions, the ciphertext was still not in the desired range (and consequently, the plaintext had to be sent in the clear). Given this, we turn to determining its security in the IK-CCA sense. We show that if the RSA family of trapdoor permutations is *partial* one-way then RSA-RAEP is anonymous.

Theorem 3. *If the* RSA *family of trapdoor permutations is partial one-way then* Π *= RSA-RAEP is anonymous. Concretely, for any adversary A attacking the*

anonymity of Π under a chosen-ciphertext attack, and making at most $q_{\rm dec}$ decryption oracle queries, $q_{\rm gen}$ G-oracle queries and $q_{\rm hash}$ H-oracle queries, there exists a θ-partial inverting adversary M_A for the RSA *family, such that for any $k, k_0(k), k_1(k)$ and $\theta = \frac{k-k_0(k)}{k}$,*

$$\mathbf{Adv}^{\text{ik-cca}}_{\Pi,A}(k) \leq 32q_{\rm hash} \cdot ((1-\epsilon_1)\cdot(1-\epsilon_2)\cdot(1-\epsilon_3))^{-1} \cdot \mathbf{Adv}^{\theta\text{-pow-fnc}}_{\mathsf{RSA},M_A}(k) + q_{\rm gen}\cdot(1-\epsilon_3)^{-1}\cdot 2^{-k+2}$$

where

$$\epsilon_1 = 4\cdot\left(\frac{3}{4}\right)^{k/2-1}; \qquad \epsilon_2 = \frac{1}{2^{k/2-3}-1};$$
$$\epsilon_3 = \frac{2q_{\rm gen}+q_{\rm dec}+2q_{\rm gen}q_{\rm dec}}{2^{k_0}} + \frac{2q_{\rm dec}}{2^{k_1}} + \frac{2q_{\rm hash}}{2^{k-k_0}},$$

and the running time of M_A is that of A plus $q_{\rm gen}\cdot q_{\rm hash}\cdot O(k^3)$. ∎

The proof of the above is in the full version of this paper [2]. Note that for typical parameters $k_0(k), k_1(k)$, and number of allowed queries $q_{\rm gen}, q_{\rm hash}$ and $q_{\rm dec}$, the values of ϵ_1, ϵ_2 and ϵ_3 are very small. This means that if there exists an adversary that is successful in breaking RSA-RAEP in the IK-CCA sense, then there exists a partial inverting adversary for the RSA family of trapdoor permutations that has a comparable advantage and running time.

The θ-partial one-wayness of RSA has been shown to be equivalent to the one-wayness of RSA, for $\theta > 0.5$ [18]. In RSA-RAEP (as also in RSA-OAEP) this is usually the case. (In general, the equivalence holds if any constant fraction of the most significant bits of the pre-image can be recovered, but the reduction is proportionately weaker [18].) Using this and Theorem 3 we are able to prove the security of RSA-RAEP in the IK-CCA sense assuming RSA to be one-way. A theorem to this effect, with concrete bounds, can be found in the full version of this paper [2].

4.2 Encryption with anonymous trapdoor permutations

Given that the standard RSA family is not anonymous, we seek families that are. We describe some simple RSA-derived anonymous families.

Construction 1 We define a family $F = (K, S, E)$ as follows. The key generation algorithm is the same as in the standard RSA family of Example 1. Let (N, e) be a public key and k the corresponding security parameter. We set $\text{Dom}_F(N,e) = \text{Rng}_F(N,e) = \{0,1\}^k$. Viewing Z_N^* as a subset of $\{0,1\}^k$ we define

$$E_{N,e}(x) = \begin{cases} x^e \bmod N & \text{if } x \in Z_N^* \\ x & \text{otherwise} \end{cases}$$

for any $x \in \{0,1\}^k$. This is a permutation on $\{0,1\}^k$. The sampling algorithm S on input N, e simply returns a random k-bit string. It is easy to see that this family is trapdoor. ∎

As we will see, the family F is perfectly anonymous. But it is not one-way. However, it is weakly one-way. (Meaning, for every polynomial-time adversary B, there is a polynomial $\beta(\cdot)$ such that $\mathbf{Adv}^{\text{1-pow-fnc}}_{F,B}(k) \leq 1 - 1/\beta(k)$ for all sufficiently large k.) Thus, standard transformations of weak to strong one-way functions (cf. [19, Section 2.3]) can be applied. Most of these preserve anonymity. To be concrete, let us use one.

Construction 2 Let $\overline{F} = (K, \overline{S}, \overline{E})$ be obtained from F of Construction 1 by Yao's cross-product construction [34]. In detail, the key-generation algorithm is unchanged and for any key N, e we set $\text{Dom}_{\overline{F}}(N, e) = \text{Rng}_{\overline{F}}(N, e) = \{0,1\}^{k^2}$. Parsing a point from this domain as a sequence of k-bit strings we set $\overline{E}_{N,e}(x_1, \ldots, x_k) = (E_{N,e}(x_1), \ldots, E_{N,e}(x_k))$. The sampling algorithm is obvious and it is easy to see the family is trapdoor. ∎

Proposition 1. *The family $\overline{F}$ of Construction 2 is a perfectly anonymous family of trapdoor, one-way permutations, under the assumption that the standard RSA family is one-way.* ∎

The proof of one-wayness is a direct consequence of the known results on the security of the cross-product construction. (A proof of Yao's result can be found in [19, Section 2.3].) The anonymity is easy to see. Regardless of the key, the adversary simply gets a random string of length k^2, and can have no advantage in determining the key based on it.

The drawback of the construction is that the cross product construction is costly, increasing both the computational and the space requirements. There are alternative amplification methods that are better and in particular do not increase space requirements, but we know of none that do not increase the computational cost.

Standard methods of trapdoor permutation based encryption yield anonymous schemes provided the underlying trapdoor permutation is anonymous. This means any encryption method based on hardcore bits [21].

These methods lead to appreciable losses of concrete security, which is why we do not state concrete security versions of the results.

Acknowledgements

The UCSD authors are supported in part by Bellare's 1996 Packard Foundation Fellowship in Science and Engineering.

References

1. M. Abadi and P. Rogaway, "Reconciling two views of cryptography (The computational soundness of formal encryption)," *Proceedings of the First IFIP International Conference on Theoretical Computer Science*, LNCS Vol. 1872, Springer-Verlag, 2000.

2. M. Bellare, A. Boldyreva, A. Desai and D. Pointcheval, "Key-privacy in public-key encryption," Full version of this paper, available via `http://www-cse.ucsd.edu/users/mihir/`.
3. M. Bellare, A. Boldyreva and S. Micali, "Public-key encryption in a multi-user setting: security proofs and improvements," *Advances in Cryptology – EUROCRYPT '00*, LNCS Vol. 1807, B. Preneel ed., Springer-Verlag, 2000.
4. M. Bellare, A. Desai, E. Jokipii and P. Rogaway, "A concrete security treatment of symmetric encryption ," *Proceedings of the 38th Symposium on Foundations of Computer Science*, IEEE, 1997.
5. M. Bellare, A. Desai, D. Pointcheval and P. Rogaway, "Relations among notions of security for public-key encryption schemes," *Advances in Cryptology – CRYPTO '98*, LNCS Vol. 1462, H. Krawczyk ed., Springer-Verlag, 1998.
6. M. Bellare, J. Kilian and P. Rogaway, "The security of the cipher block chaining message authentication code," *Advances in Cryptology – CRYPTO '94*, LNCS Vol. 839, Y. Desmedt ed., Springer-Verlag, 1994.
7. M. Bellare and P. Rogaway, Random oracles are practical: a paradigm for designing efficient protocols. *First ACM Conference on Computer and Communications Security*, ACM, 1993.
8. M. Bellare and P. Rogaway, "Optimal asymmetric encryption – How to encrypt with RSA," *Advances in Cryptology – EUROCRYPT '95*, LNCS Vol. 921, L. Guillou and J. Quisquater ed., Springer-Verlag, 1995.
9. M. Blum and S. Goldwasser, "An efficient probabilistic public-key encryption scheme which hides all partial information," *Advances in Cryptology – CRYPTO '84*, LNCS Vol. 196, R. Blakely ed., Springer-Verlag, 1984.
10. J. Camenisch and A. Lysyanskaya, "Efficient non-transferable anonymous multi-show credential system with optional anonymity revocation," *Advances in Cryptology – EUROCRYPT '01*, LNCS Vol. 2045, B. Pfitzmann ed., Springer-Verlag, 2001.
11. D. Coppersmith, "Finding a small root of a bivariate integer equation; factoring with high bits known," *Advances in Cryptology – EUROCRYPT '96*, LNCS Vol. 1070, U. Maurer ed., Springer-Verlag, 1996.
12. R. Cramer and V. Shoup, "A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack," *Advances in Cryptology – CRYPTO '98*, LNCS Vol. 1462, H. Krawczyk ed., Springer-Verlag, 1998.
13. A. Desai, "The security of all-or-nothing encryption: protecting against exhaustive key search," *Advances in Cryptology – CRYPTO '00*, LNCS Vol. 1880, M. Bellare ed., Springer-Verlag, 2000.
14. Y. Desmedt, "Securing traceability of ciphertexts: Towards a secure software escrow scheme," *Advances in Cryptology – EUROCRYPT '95*, LNCS Vol. 921, L. Guillou and J. Quisquater ed., Springer-Verlag, 1995.
15. D. Dolev, C. Dwork and M. Naor, "Non-malleable cryptography," *SIAM J. on Computing*, Vol. 30, No. 2, 2000, pp. 391–437.
16. T. ElGamal, "A public key cryptosystem and signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol 31, 1985, pp. 469–472.
17. M. Fischlin, "Pseudorandom Function Tribe Ensembles based on one-way permutations: Improvements and applications," *Advances in Cryptology – EUROCRYPT '99*, LNCS Vol. 1592, J. Stern ed., Springer-Verlag, 1999.
18. E. Fujisaki, T. Okamoto, D. Pointcheval and J. Stern, "RSA-OAEP is Secure under the RSA Assumption," *Advances in Cryptology – CRYPTO '01*, LNCS Vol. 2139, J. Kilian ed., Springer-Verlag, 2001.

19. O. Goldreich, "Foundations of Cryptography, Basic Tools," Cambridge University Press, 2001.
20. O. Goldreich, S. Goldwasser and S. Micali, "How to construct random functions," *Journal of the ACM,* Vol. 33, No. 4, 1986, pp. 210–217.
21. O. Goldreich and L. Levin, "A hard-core predicate for all one-way functions," *Proceedings of the 21st Annual Symposium on the Theory of Computing*, ACM, 1989.
22. S. Goldwasser and S. Micali, "Probabilistic encryption," *J. of Computer and System Sciences*, Vol. 28, April 1984, pp. 270–299.
23. H. Krawczyk, "SKEME: A Versatile Secure Key Exchange Mechanism for Internet," *Proceedings of the 1996 Internet Society Symposium on Network and Distributed System Security*, 1996.
24. National Bureau of Standards, NBS FIPS PUB 81, "DES modes of operation," U.S Department of Commerce, 1980.
25. M. Naor and O. Reingold, "Number-theoretic constructions of efficient pseudo-random functions," *Proceedings of the 38th Symposium on Foundations of Computer Science*, IEEE, 1997.
26. M. Naor and M. Yung, "Universal one-way hash functions and their cryptographic applications," *Proceedings of the 21st Annual Symposium on the Theory of Computing*, ACM, 1989.
27. M. Naor and M. Yung, "Public-key cryptosystems provably secure against chosen ciphertext attacks," *Proceedings of the 22nd Annual Symposium on the Theory of Computing*, ACM, 1990.
28. RSA Labs, "PKCS-1," `http://www.rsasecurity.com/rsalabs/pkcs/pkcs-1/`.
29. C. Rackoff and D. Simon, "Non-interactive zero-knowledge proof of knowledge and chosen-ciphertext attack," *Advances in Cryptology – CRYPTO '91*, LNCS Vol. 576, J. Feigenbaum ed., Springer-Verlag, 1991.
30. K. Sako, "An auction protocol which hides bids of losers," *Proceedings of the Third International workshop on practice and theory in Public Key Cryptography (PKC 2000)*, LNCS Vol. 1751, H. Imai and Y. Zheng eds., Springer-Verlag, 2000.
31. V. Shoup, "On formal models for secure key exchange, " Technical report. Theory of Cryptography Library: 1999 Records.
32. M. Stadler, "Publicly verifiable secret sharing," *Advances in Cryptology – EUROCRYPT '96*, LNCS Vol. 1070, U. Maurer ed., Springer-Verlag, 1996.
33. Y. Tsiounis and M. Yung, "On the security of El Gamal based encryption," *Proceedings of the First International workshop on practice and theory in Public Key Cryptography (PKC'98)*, LNCS Vol. 1431, H. Imai and Y. Zheng eds., Springer-Verlag, 1998.
34. A. Yao, "Theory and applications of trapdoor functions, " *Proceedings of the 23rd Symposium on Foundations of Computer Science*, IEEE, 1982.