Randomness Re-use in Multi-recipient Encryption Schemeas

Mihir Bellare[1], Alexandra Boldyreva[1], and Jessica Staddon[2]

[1] Dept. of Computer Science & Engineering
University of California at San Diego
9500 Gilman Drive, La Jolla, California 92093, USA
mihir@cs.ucsd.edu
http://www-cse.ucsd.edu/users/mihir
[2] Dept. of Computer Science & Engineering
University of California at San Diego
9500 Gilman Drive, La Jolla, California 92093, USA
aboldyre@cs.ucsd.edu
http://www-cse.ucsd.edu/users/aboldyre

Abstract. Kurosawa showed how one could design multi-receiver encryption schemes achieving savings in bandwidth and computation relative to the naive methods. We broaden the investigation. We identify new types of attacks possible in multi-recipient settings, which were overlooked by the previously suggested models, and specify an appropriate model to incorporate these types of attacks. We then identify a general paradigm that underlies his schemes and also others, namely the re-use of randomness: ciphertexts sent to different receivers by a single sender are computed using the same underlying coins. In order to avoid case by case analysis of encryption schemes to see whether they permit secure randomness re-use, we provide a condition, or test, that when applied to an encryption scheme shows whether or not the associated randomness re-using version of the scheme is secure. As a consequence, our test shows that randomness re-use is secure in the strong sense for asymmetric encryption schemes such as El Gamal, Cramer-Shoup, DHIES, and Boneh and Franklin's escrow El Gamal.

Keywords: Encryption, randomness, provable security, broadcast encryption.

1 Introduction

The standard setting for encryption is that a sender, in possession of the encryption key K of a receiver and a message M that it wants to send privately to this receiver, computes a ciphertext C by applying an encryption algorithm to K and M, and sends C to the receiver. We are interested in a setting where there is one sender but multiple receivers. The sender is in possession of encryption keys $K_1, \ldots, K_n$ of receivers $1, \ldots, n$ respectively, and of message $M_1, \ldots, M_n$ that it wants to send privately to receivers $1, \ldots, n$ respectively. A *multi-recipient*

Y.G. Desmedt (Ed.): PKC 2003, LNCS 2567, pp. 85–99, 2003.

encryption scheme is just like an encryption scheme except that the encryption algorithm is replaced by a (randomized) *multi-recipient encryption algorithm* which, given $K_1, \ldots, K_n$ and $M_1, \ldots, M_n$, outputs ciphertexts $C_1, \ldots, C_n$, with C_i an encryption of M_i under K_i.

There is of course a naive, or obvious way to build a multi-receiver encryption scheme: for each i simply encrypt M_i under K_i using the encryption algorithm of a standard scheme. However, viewing the task of producing multiple ciphertexts as being done by a single process allows one to explore reductions in cost that might arise from batching. In particular it enables different encryptions to be based on the same coins.

In this paper we introduce and define a subclass of multi-recipient encryption sche-mes that we call randomness-reusing multi-recipient encryption schemes. Let $\mathcal{E}$ denote the encryption algorithm of some standard encryption scheme. In the associated randomness-reusing multi-recipient encryption scheme, one picks at random coins r for $\mathcal{E}$, and then, for each i, computes $C_i = \mathcal{E}(K_i, M_i)$. In other words, the different ciphertexts are computed using the same coins.

Motivating examples. The definition of randomness-reusing multi-recipient encryption schemes was motivated by the work of Kurosawa [22]. Here is an example from his paper. Suppose a sender wants to send message M_i to receiver i encrypted under the latter's El Gamal public key g^{x_i} $(1 \leq i \leq n)$. The naive procedure would be to separately encrypt each message with new coins, meaning pick $r_1, \ldots, r_n$ at random, let $C_i = (g^{r_i}, g^{x_i r_i} \cdot M_i)$, and send C_i to i for $1 \leq i \leq n$. Kurosawa [22] considers picking just one r at random and setting $C_i = (g^r, g^{x_i r} \cdot M_i)$ instead. Kurosawa's main motivation was to reduce bandwidth in the case that the ciphertexts were being broadcast or multi-cast by the sender, since in that case, the transmission would now be $\mathbf{C} = (g^r, g^{x_1 r} \cdot M_1, \ldots, g^{x_n r} \cdot M_n)$, which is about half as many bits as required to transmit the ciphertexts computed by the naive method. However, he also points out that his suggested scheme halves the computational cost (number of exponentiations), a more broadly applicable and perhaps more useful savings than the one in bandwidth. Kurosawa notes similar savings in using the Cramer-Shoup encryption scheme [13].

We note that the technique underlying Kurosawa's scheme is randomness re-use, specifically re-use of r as coins for El Gamal encryption of different messages under different public keys. Accordingly, we are considering randomness re-use at a more general level.

Security issues and model. Before we can meaningfully address the security of specific multi-recipient encryption schemes such as randomness-reusing ones, we need a model of security. We seek notions of security for multi-recipient encryption schemes, specifically appropriate definitions of IND-CPA and IND-CCA in this context.

Kurosawa [22] proposed such definitions based on a fairly direct adaptation of the definitions of encryption security in the multi-user setting [3, 1]. However, although the latter do explicitly consider the presence of many recipients, they assume all encryptions are produced under independent coins, which is not true for multi-recipient schemes. In particular, we show that Kurosawa's model and definitions fail to cover several practical attacks, and thus security proved under his definitions may not suffice for applications. We remedy this by providing a model that takes the new attacks into account. In Section 4 we specify the model and also provide examples of schemes that can be proven secure in the model of [22] but fall to practical attacks and can be (correctly) shown to be insecure in our model. Let us now highlight some of the new security issues for multi-recipient schemes that we consider.

First are rogue-key attacks. The framework is well-known, and consists of an adversary registering public keys created as a function of public keys of other, legitimate users. This can be particularly damaging in the context of random-string re-use, as we illustrate in Section 4 with a rogue-key attack on Kurosawa's El Gamal based scheme. It is important to be aware of this attack, but it is for such reasons that certification authorities require (or should require) that a user registering a public key prove knowledge of the corresponding secret key. (In that case, this attack fails.) The assumption we make in this paper is that the adversary cannot register a public key without knowing the corresponding secret key. The assumption is built into our formal model by requiring the adversary, at the time it corrupts a user, to supply not only a public key for that user, but also a corresponding secret key.

Second are insider attacks. An adversary who is one of the legitimate recipients can decrypt a received ciphertext, and might then obtain the coins r underlying the encryption. This is not a concern if, as in [1, 3], encryptions to other recipients use independent coins, and thus these works do not consider insider attacks. But in a multi-recipient scheme, the ciphertext sent to another recipient might be based on the same coins r, and thus the adversary might obtain information about the plaintext underlying this ciphertext too. Our model takes this into account, by allowing the adversary to corrupt some fraction of the users and choose secret and public keys for them. We present a variant of Kurosawa's El Gamal based randomness re-use scheme that is provably secure in his model but insecure under our model due to insider attacks. The attack is a practical one, highlighting the value of the enhanced model in capturing real attacks.

Reproducibility property and theorem. Not all encryption schemes can securely re-use randomness. An example of a class of schemes that cannot are RSA embedding schemes such as PKCS#1: we illustrate in Section 3 how Håstad's attacks [19] can be exploited to break these schemes if randomness is re-used in generation of ciphertexts for three different receivers. Thus, an important issue is, given an encryption scheme, determine whether or not it permits secure randomness re-use.

Looking at the description of the existing encryption scheme it is easy to decide whether re-use of randomness will allow computational or bandwidth savings. However, it is not clear how to check whether this can be done securely. Case by case analysis of the many existing encryption schemes, e.g. following the proof techniques of [22] although possible, would be prohibitive. One of the main contributions of this paper is a way to establish that an encryption scheme permits secure randomness re-use based on existing security results about the scheme. It takes two parts: definition of a property of encryption schemes called *reproducibility*, and a theorem, called the *reproducibility theorem*. The latter says that if an encryption scheme is reproducible and is IND-CPA (resp. IND-CCA) in the standard, single-receiver setting, then the corresponding randomness re-using multi-recipient scheme is also IND-CPA (resp. IND-CCA) with respect to our notions of security for such schemes. It is usually easy to check whether a given encryption scheme is reproducible, and the test and theorem are valid for several asymmetric schemes, so numerous applications follow.

Reproducibility itself is quite simply explained. Focusing on the asymmetric case, let pk_1, pk_2 be public encryption keys, and let $C_1 = \mathcal{E}_{pk_1}(M_1, r)$ be a ciphertext of a message M_1 created under key pk_1 based on random string r. We say that the encryption scheme is *reproducible* if, given pk_1, pk_2, C_1, any message M_2, and the secret decryption key sk_2 corresponding to pk_2, there is a polynomial time *reproduction algorithm* that returns the ciphertext $C_2 = \mathcal{E}_{pk_2}(M_2, r)$. It might seem at first as a counter-intuitive property, exploiting some weakness of the encryption scheme. We show, however, that reproducibility itself does not compromise security and moreover, permits secure randomness re-use.

We now discuss applications of the reproducibility test and theorem to various asymmetric schemes.

El Gamal and Cramer-Shoup. The corresponding randomness re-using schemes are those of Kurosawa [22], which he proved secure under the DDH (Decisional Diffie-Hellman) assumption. As noted above, however, his target notion of security is weak. Thus one needs to ask whether the schemes remain secure under our stronger notion of security. This is important because these are the schemes permitting the computation and broadcast ciphertext size-reductions noted above.

We show that the base El Gamal and Cramer-Shoup schemes are both reproducible, and our reproducibility theorem then says that indeed, Kurosawa's schemes remain secure with respect to our more stringent security notions. We then extend these results by providing reductions of improved concrete security. These improvements bypass the reproducibility theorem, instead directly exploiting the reproducibility property of the base schemes and, as in [3], using self-reducibility properties of the DDH problem [28, 24, 27].

DHIES. This is a Diffie-Hellman based asymmetric encryption scheme adopted by draft standards ANSI X9.63EC and IEEE P1363a. It has El Gamal-like cost in public-key operations while achieving Cramer-Shoup-like security (IND-CCA), although the proof [2] relies on significantly stronger assumptions than the DDH assumption used in [13]. Unlike El Gamal and Cramer-Shoup it does not assume the plaintext is a group element, but handles arbitrary plaintext strings via an integrated construction involving a symmetric encryption scheme. Randomness re-use for this scheme is attractive since it results in bandwidth and computational savings in various applications just as for the El Gamal scheme, so it is important to assess security.

We consider the case when the symmetric encryption scheme used in (asymmetric) DHIES scheme is CBC mode combined with any block cipher, e.g. AES (the most popular choice in practice) and show that then DHIES is reproducible. As usual, our reproducibility theorem then implies that the corresponding randomness re-using multi-recipient scheme is IND-CCA under the assumptions used to establish that DHIES is IND-CCA.

Pairings-based escrow El Gamal. Boneh and Franklin [10] introduced an El Gamal like scheme with global escrow capabilities, based on the Weil pairing. We show that this scheme is reproducible. Our reproducibility theorem coupled with the result of [10] then implies that the corresponding randomness re-using multi-recipient scheme is IND-CPA in the random oracle model under the Bilinear Diffie-Hellman assumption. Our reproducibility algorithm exploits properties of the Weil pairing. Again, as for El Gamal scheme, re-using randomness permits computational and bandwidth savings.

Randomness re-use in symmetric encryption A novel element of our work compared to [22, 3] is consideration of the symmetric setting. In the full version of this paper [4] we show that reproducibility and the corresponding theorem apply in this setting too. We prove that CBC encryption with random IV, based on a given block cipher permits secure randomness re-use in the multi-recipient setting.

Minimal assumptions. In we determine minimal assumptions under which one can prove the existence of an encryption scheme permitting secure randomness re-use. We show that there exists an encryption scheme which under randomness re-use yields an IND-CPA multi-receiver encryption scheme if and only if there exists a standard IND-CPA encryption scheme. The analog holds for IND-CCA, and these results hold in both the symmetric and the asymmetric settings.

2 Definitions

We recall the standard definitions. An *asymmetric encryption scheme* $\mathcal{AE} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ consists of four polynomial-time algorithms. The randomized *common-key generation* algorithm $\mathcal{G}$ takes as input a security parameter $k \in \mathsf{N}$ and returns a *common key* I; we write $I \overset{R}{\leftarrow} \mathcal{G}(k)$. I could include a prime number and a generator of a group, which all parties use to create their keys.) The randomized *key generation* algorithm $\mathcal{K}$ takes as input the common key I and returns a pair (pk, sk) consisting of a public key and a corresponding secret key; we write $(pk, sk) \overset{R}{\leftarrow} \mathcal{K}(I)$. In our context it is important to make explicit the random choices underlying the (randomized) *encryption* algorithm $\mathcal{E}$. On input a public key pk, a plaintext M, and coin tosses r, it returns the ciphertext $C = \mathcal{E}_{pk}(M; r)$. The notation $C \overset{R}{\leftarrow} \mathcal{E}_{pk}(M)$ is shorthand for $r \overset{R}{\leftarrow} \text{Coins}_{\mathcal{E}}(I, pk)\,;\, C \leftarrow \mathcal{E}_{pk}(M; r)$, where $\text{Coins}_{\mathcal{E}}(I, pk)$ is a set from which $\mathcal{E}$ draws its coins. The deterministic *decryption* algorithm $\mathcal{D}$ takes the secret key sk and a ciphertext C to return the corresponding plaintext or a special symbol $\perp$ indicating that the ciphertext was invalid; we write $x \leftarrow \mathcal{D}_{sk}(C)$. Associated to each common key I is a *message space* $\mathrm{MsgSp}(I)$ from which M is allowed to be drawn. We require that $\mathcal{D}_{sk}(\mathcal{E}_{pk}(M)) = M$ for all $M \in \mathrm{MsgSp}(I)$. We will use the terms "plaintext" and "message" interchangeably.

Let $\mathsf{Adv}^{\mathrm{cpa}}_{\mathcal{AE},A_{\mathrm{cpa}}}(\cdot)$ (resp. $\mathsf{Adv}^{\mathrm{cca}}_{\mathcal{AE},A_{\mathrm{cca}}}(\cdot)$) denote the advantage of adversary A_{cpa} (resp. A_{cca}) in breaking the scheme $\mathcal{AE}$ under a chosen-plaintext (resp. chosen-cipher-text) attack, as per the usual standard notions of security IND-CPA and IND-CCA. (We recall the formal definitions in [4]).

MULTI-RECIPIENT ENCRYPTION SCHEMES. In order to allow consideration of methods of producing multiple ciphertexts based on the same randomness, this paper introduces a primitive that we call a multi-recipient encryption scheme. Formally an *asymmetric multi-recipient encryption scheme* $\overline{\mathcal{AE}} = (\mathcal{G}, \mathcal{K}, \overline{\mathcal{E}}, \mathcal{D})$ consists of four algorithms. The common-key generation algorithm $\mathcal{G}$, the key generation algorithm $\mathcal{K}$, and the decryption algorithm $\mathcal{D}$ are as in a standard asymmetric encryption scheme above. On input a *public-key vector* $\mathbf{pk} = (\mathbf{pk}[1], \ldots, \mathbf{pk}[n])$, a *plaintext vector* $\mathbf{M} = (\mathbf{M}[1], \ldots, \mathbf{M}[n])$, and coin tosses r, the *multi-encryption* algorithm $\overline{\mathcal{E}}$ returns the *ciphertext vector* $\mathbf{C} = (\mathbf{C}[1], \ldots, \mathbf{C}[n]) = \overline{\mathcal{E}}_{\mathbf{pk}}(\mathbf{M})$. The notation $\mathbf{C} \overset{R}{\leftarrow} \overline{\mathcal{E}}_{\mathbf{pk}}(\mathbf{M})$ is shorthand for $r \overset{R}{\leftarrow} \mathrm{Coins}_{\overline{\mathcal{E}}}(I, \mathbf{pk})$; $\mathbf{C} \leftarrow \overline{\mathcal{E}}_{\mathbf{pk}}(\mathbf{M}; r)$, where $\mathrm{Coins}_{\overline{\mathcal{E}}}(I, \mathbf{pk})$ is a set from which $\mathcal{E}$ draws its coins. Associated with a common key I is a *message space* $\mathrm{MsgSp}(I)$ from which the components of $\mathbf{M}$ are allowed to be drawn. We require that for all $\mathbf{M}$ with components in the message space, the following experiment returns 1 with probability 1:

$$\text{For } i = 1, \ldots, n \text{ do } (\mathbf{pk}[i], \mathbf{sk}[i]) \overset{R}{\leftarrow} \mathcal{K}(k) \text{ EndFor; } \mathbf{C} \overset{R}{\leftarrow} \overline{\mathcal{E}}_{\mathbf{pk}}(\mathbf{M}) \text{ ;}$$
$$i \overset{R}{\leftarrow} \{1, \ldots, n\} \text{ ; If } (\mathcal{D}_{\mathbf{sk}[i]}\mathbf{C}[i]) = \mathbf{M}[i] \text{ then return 1 else return 0}$$

SRS MUTLI-RECEIVER ENCRYPTION SCHEME. We are interested in a specific multi-receiver encryption scheme, obtained from a given asymmetric encryption scheme by using the same coins to encrypt the different messages in the message vector.

Definition 1. *The same random string (SRS) multi-receiver encryption scheme associated to a given asymmetric encryption scheme* $\mathcal{AE} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *is the multi-recipient encryption scheme* $\overline{\mathcal{AE}} = (\mathcal{G}, \mathcal{K}, \overline{\mathcal{E}}, \mathcal{D})$ *in which the common key generation, key generation algorithms and decryption algorithms are that of* $\mathcal{AE}$ *and the multi-recipient encryption algorithm is defined as follows:*

```
$\overline{\mathcal{E}}_{\mathbf{pk}}(\mathbf{M})$
        Let $n$ be the number of components of $\mathbf{M}$   [and also of $\mathbf{pk}$]
        $r \overset{R}{\leftarrow} Coins_{\mathcal{E}}(I, \mathbf{pk})$ ;
        For $i = 1, \ldots n$ do $\mathbf{C}[i] \leftarrow \mathcal{E}_{pk_i}(\mathbf{M}[i]; r)$ EndFor Return $\mathbf{C}$.
```

We refer to $\mathcal{AE}$ *as the base scheme of* $\overline{\mathcal{AE}}$.

We do not specify how $\mathbf{C}[i]$ is communicated to user i. It could be that the whole ciphertext vector $\mathbf{C}$ is sent via a broadcast or multi-cast channel and, if all $\mathbf{C}[i]$

have a common part due to a randomness re-use, this part can be sent only once. It could also be that $\mathbf{C}[i]$ is sent to party i directly. This issue depends on the specific application and is not relevant for security of the scheme. For examples of SRS schemes see Section 6.

3 Not Every SRS Scheme is Secure

We consider general embedding schemes which first apply a randomized invertible transform to a message and then apply a trapdoor permutation to the result. The example of such schemes is RSA-PKCS#1 [25] that has been proven to be IND-CCA secure (in the random oracle model) [16] and hence is also IND-CCA secure in a multi-user setting [1, 3]. Nonetheless, the associated SRS scheme is insecure. The attack is as follows. Let N_i be the public modulus of user i and assume all users have encryption exponent 3. Suppose the sender wants to send a single message M to three receivers, namely $\mathbf{M} = (M, M, M)$. Under the SRS scheme, it will pick a random string r, using M and a random r will compute a transform x, set $\mathbf{C}[i] = x^3 \bmod N_i$, and send $\mathbf{C}[i]$ to i. An adversary given $\mathbf{C}$ can use Håstad's attack (based on the fact that the modulii are relatively prime) to recover x, and them recover M by inverting the transform. The same attack applies regardless of embedding method, since the latter must be an invertible transform.

This indicates that secure randomness re-use is not possible for *all* base encryption schemes: there exist base encryption schemes that are secure, yet the associated the SRS multi-recipient encryption scheme is not secure. As we will see later, no encryption scheme where the random string used in encryption algorithm is a by-product of decryption can be a base of a secure SRS scheme,however, there are large classes of base encryption schemes for which the associated SRS scheme is secure. Before we can get there, we need to discuss what "secure" means.

4 Security of Multi-recipient Schemes

We provide the definition and follow it with a discussion illustrating how it takes into account the various security issues mentioned in the introduction.

Model and definition. Let $\overline{\mathcal{AE}} = (\mathcal{G}, \mathcal{K}, \overline{\mathcal{E}}, \mathcal{D})$ be an asymmetric, multi-recipient encryption scheme. (We are particularly interested in the case where this is an SRS scheme, but the definition is not restricted to this case.) Let n be a polynomial. For atk $\in \{\mathrm{cpa}, \mathrm{cca}\}$ and for an adversary B attacking the scheme we define the experiment $\mathbf{Exp}^{n\text{-mr-atk-b}}_{\overline{\mathcal{AE}},B}(k)$ as follows. B runs in three stages. In the select stage the adversary is given an initial information string and outputs l such that $1 \le l \le n$, which indicates that it wants to corrupt $n - l$ users, assumed without loss of generality to be users $l + 1, \ldots, n$. In the find stage the adversary is given I and the public keys of the honest users $1, \ldots, l$. It outputs *two* l-vectors of messages corresponding to choices for the honest users;

one $(n-l)$-vector of messages corresponding to choices for the corrupted users; a $(n-l)$-vector of public keys for the corrupted users; and a $(n-l)$-vector of corresponding secret keys (see the discussion below.) Based on a challenge bit b, one of the two l-vectors is selected, and the components of the $(n-l)$-vector of messages are appended to yield a challenge n-vector of messages $\mathbf{M}$. The latter is encrypted via the multi-encryption algorithm to yield a challenge ciphertext $\mathbf{C}$ that is returned to the adversary, now in its guess stage. It wins if it returns a bit d that equals the challenge bit b. In each stage the adversary will output state information that is returned to it in the next stage. When $\mathrm{atk} = \mathrm{cca}$, the adversary gets oracles $\mathcal{D}_{sk_i}(\cdot)$ for $1 \leq i \leq l$ with the restriction of not querying them on the corresponding components of the challenge ciphertext vector.

Definition 2. *Let* $\overline{\mathcal{AE}} = (\mathcal{G}, \mathcal{K}, \overline{\mathcal{E}}, \mathcal{D})$ *be a multi-recipient encryption scheme, let* n *be a polynomial and let* $\mathrm{atk} \in \{\mathrm{cpa}, \mathrm{cca}\}$. *Then for any security parameter* k *ind-*atk *advantage of an adversary* B *is*

$$\mathsf{Adv}^{n\text{-mr-atk}}_{\overline{\mathcal{AE}},B}(k) \;=\; \Pr\left[\,\mathbf{Exp}^{n\text{-mr-atk-0}}_{\overline{\mathcal{AE}},B}(k) = 0\,\right] - \Pr\left[\,\mathbf{Exp}^{n\text{-mr-atk-1}}_{\overline{\mathcal{AE}},B}(k) = 0\,\right].$$

Definition 3. *Let* $\overline{\mathcal{AE}} = (\mathcal{G}, \mathcal{K}, \overline{\mathcal{E}}, \mathcal{D})$ *be a multi-recipient encryption scheme. We say that it is IND-CPA (resp. IND-CCA) secure if the function* $\mathsf{Adv}^{n\text{-mr-cpa}}_{\overline{\mathcal{AE}},B}(\cdot)$ *(respectively* $\mathsf{Adv}^{n\text{-mr-cca}}_{\overline{\mathcal{AE}},B}(\cdot)$*) is negligible for any polynomial-time adversary* B *and any polynomial* n.

It is convenient to introduce a notion of security for base encryption schemes based on the security of the corresponding SRS scheme. We stress that the following is a notion of security for (standard) asymmetric encryption schemes, not for multi-recipient encryption schemes.

Definition 4. *Let* $\mathcal{AE}$ *be an asymmetric encryption scheme. We say that it is SRS-IND-CPA (resp. SRS-IND-CCA) secure (or, briefly SRSS) if the SRS multi-recipient asymmetric encryption scheme* $\overline{\mathcal{AE}}$ *associated to* $\mathcal{AE}$ *is IND-CPA (resp. IND-CCA) secure.*

Discussion and comparison with the model of security of [22]. Previous works [1, 3, 22] only considered outsider attacks, meaning the adversary was not one of the receivers. A novel element of our model relative to [1, 3, 22] is the consideration of insider attacks. The adversary is allowed to corrupt some fraction of the users and choose secret and public keys for them.

We argue that it is necessary for a model of security of multi-recipient schemes to take into account insider attacks. The model of [22] does not address this problem and we show that there exist multi-recipient encryption schemes which can be proven secure using the model of [22] but are obviously insecure and can easily be shown insecure using our model of security.

It is proved in [22] that El Gamal scheme permits secure randomness re-use in the multi-recipient setting. Now consider a modified encryption scheme which differs from El Gamal in that its encryption algorithm when invoked on one particular public key (e.g. g^3) in addition to a ciphertext returns randomness used to compute it. Assume this fact is known to the adversary. When this scheme used in a multi-recipient setting with randomness re-use the adversary can certify this public key and later after receiving a ciphertext can obtain the random string used to compute the ciphertexts of other users and thus break the scheme. Under our model the advantage of such adversary in breaking this scheme will be 1. But in the model of [22] all the public keys assumed to be random, and the scheme can be proven secure.

Consider another example which exploits a different weakness of the model of [22]. Let $\mathcal{AE}' = (\mathcal{G}', \mathcal{K}', \mathcal{E}', \mathcal{D}')$ be some IND-CPA secure encryption scheme. Consider a multi-recipient scheme $\overline{\mathcal{AE}}$ with user i's public key $pk_i = (g^{x_i}, pk'_i)$, where g^{x_i} is a public key for El Gamal encryption and pk'_i is a public key of $\mathcal{AE}'$. Let the encryption algorithm of $\mathcal{AE}'$ be as follows. It first draws a random value r at random. Then it computes $C[i]$ as $(g^r, (g^{x_i})^r M[i], C'[i])$ where $C'[i] = \mathcal{E}'_{pk'_i}(r)$. In other words each ciphertext consists of an El Gamal ciphertext computed with common randomness and of encryption of this common randomness under some fixed encryption scheme. We claim that there exists an attack on $\overline{\mathcal{AE}}$ but the scheme can be proven secure under the model of [22]. We first show that $\overline{\mathcal{AE}}$ is insecure in practice by presenting an attack. An adversary A "corrupts" the first user and chooses $pk_1 = (g^{x_1}, pk'_1)$ in normal way so that it knows x_1, sk'_1. When A receives a ciphertext vector $\mathbf{C}$ it decrypts $C'[1]$ using sk'_1 and obtains r. Now A can test whether particular messages were encrypted under the public keys of other users. Under our model of security A would have advantage 1. We now show that $\overline{\mathcal{AE}}$ is secure under the model of [22]. Let B be an adversary attacking $\overline{\mathcal{AE}}$ under the model of [22]. Then it is possible to construct an adversary D which attacks SRS El Gamal multi-recipient scheme. But [22] proves the latter scheme is secure, so this would imply that $\overline{\mathcal{AE}}$ is secure. D simply provides all the public keys it is given to B and outputs message vectors that B outputs. D then receives a challenge ciphertext vector $\mathbf{C}_D$, picks a random r' and computes a challenge $\mathbf{C}_B$ for B such that $\mathbf{C}_B[i] = (\mathbf{C}_D[i], \mathcal{E}'_{pk'_i}(r'))$. Since $\mathcal{AE}'$ is IND-CPA then the view of B in the simulated experiment is indistinguishable from the real experiment. Therefore the advantage of B is at most the advantage of D, but it is proven in [22] that the latter is negligible.

Moreover, the model of [22], as well as of [3, 1] do not take into account the possibility of rogue-key attack. This can be particularly damaging in the context of random-string re-use. For example, suppose the adversary registers public keys $(g^x)^2 = g^{2x}$ and $(g^x)^3 = g^{3x}$ where g^x is the key of a legitimate user. Suppose that symmetric session keys K_1, K, K are El Gamal encrypted with the same randomness r under public keys g^x, g^{2x}, g^{3x} and broadcast to the users. Thus the adversary sees the three corresponding ciphertexts $(g^r, g^{rx} \cdot K_1), (g^r, g^{2rx} \cdot K), (g^r, g^{3rx} \cdot K)$. From them it can compute $K_1 = [g^{rx} \cdot K_1] \cdot [g^{2rx} \cdot K] \cdot [g^{3rx} \cdot K]^{-1}$ and obtain the session key of the legitimate user. As a consequence, the adversary will be able to decrypt the secret information encrypted under this session key addressed to the legitimate user.

As we mentioned in the introduction, to prevent attacks of this type we put some limitation on the adversary in this regard, in particular to disallow it from creating public keys whose corresponding secret keys it does not know. The model incorporates this by requiring the adversary to supply, along with public keys for the corrupted users, corresponding secret keys. This models the effect of appropriate proofs of knowledge of the secret key that are assumed to be done as part of the key certification process. The alternative is to explicitly consider the certification process in the model, and then, in proofs of security, use the extractors, guaranteed by the proof of knowledge property [6], to extract the secret keys from the adversary. This being quite a complication of the model, we have chosen to build in the intended effects of the proofs of knowledge.

5 Reproducibility Test and Theorem

We provide a condition under which a given encryption scheme can be a base of the secure SRS scheme. Informally speaking, the condition is satisfied for those encryption schemes for which it is possible, using a public key and ciphertext of a random message, to create ciphertexts for arbitrary messages under arbitrary keys, such that all ciphertexts employ the same random string as that of the given ciphertext.

Definition 5. *Fix a public-key encryption scheme $\mathcal{AE} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$. Let n be polynomial in k, and let R be an algorithm that takes as input a public key and ciphertext of a random message, another random message together with a public-secret key pair, and returns a ciphertext. Consider the following experiment.*

Experiment $\mathbf{Exp}^{repr}_{\mathcal{AE},R}(k)$

```
    $I \stackrel{R}{\leftarrow} \mathcal{G}(k)$ ; $(pk, sk) \stackrel{R}{\leftarrow} \mathcal{K}(I)$ ; $M \stackrel{R}{\leftarrow} \mathrm{MsgSp}(I)$ ; $r \stackrel{R}{\leftarrow} Coins_{\mathcal{E}}(I, pk)$
    $C \leftarrow \mathcal{E}_{pk}(M, r)$ ; $(pk', sk') \stackrel{R}{\leftarrow} \mathcal{K}(I)$ ; $M' \stackrel{R}{\leftarrow} \mathrm{MsgSp}(I)$
    If $(\mathcal{E}_{pk'}(M', r) = R(pk, C, M', pk', sk'))$ then Return 1 else Return 0 EndIf
```

We say that $\mathcal{AE}$ is reproducible if for any k there exists a probabilistic, poly-time algorithm R called the reproduction algorithm such that $\mathbf{Exp}^{repr}_{\mathcal{AE},R}(k)$ outputs 1 with the probability 1.

Later we will show that many popular discrete-log-based encryption schemes are reproducible. It is an open question whether there exist reproducible encryption schemes of other types.

We now state the main reproducibility theorem. It implies that if an encryption scheme is reproducible and is IND-CPA (resp. IND-CCA) secure, then it is also SRS-IND-CPA (resp. SRS-IND-CCA) secure. The proof is in the full version of this paper [4].

Theorem 1. *Fix a public-key encryption scheme $\mathcal{AE} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ and a polynomial $n(\cdot)$. Let $\overline{\mathcal{AE}} = (\mathcal{G}, \mathcal{K}, \overline{\mathcal{E}}, \mathcal{D})$ be the associated SRS scheme. If $\mathcal{AE}$ is reproducible then for any poly-time adversary B_{atk}, there exists a poly-time adversary A_{atk}, where* atk = {cpa, cca}*, such that for any k*

$$\mathsf{Adv}^{n\text{-mr-atk}}_{\overline{\mathcal{AE}},B_{\mathrm{atk}}}(k) \;\leq\; n(k) \cdot \mathsf{Adv}^{\mathrm{atk}}_{\mathcal{AE},A_{\mathrm{atk}}}(k).$$

6 Analysis of Specific Schemes

In this section we show that many popular encryption schemes are reproducible. Using the known results about security of these schemes and the result of Theorem 1 this would imply that these schemes are also SRSS.

We first consider three DDH-based schemes which work over a group of prime order. 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.

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

```
$\underline{\mathcal{K}(q, g)}$:
$x \overset{R}{\leftarrow} Z_q$ ; $X \leftarrow g^x$
$pk \leftarrow (q, g, X)$ ; $sk \leftarrow (q, g, x)$
Return $(pk, sk)$
```

```
$\underline{\mathcal{E}_{pk}(M)}$:
$r \overset{R}{\leftarrow} Z_q$ ; $Y \leftarrow g^r$
$T \leftarrow X^r$ ; $W \leftarrow TM$
Return $(Y, W)$
```

```
$\underline{\mathcal{D}_{sk}(Y, W)}$:
$T \leftarrow Y^x$
$M \leftarrow WT^{-1}$
Return $M$
```

Lemma 1. *The El Gamal encryption scheme $\mathcal{EG} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ is reproducible.*

Proof. On input $(g^x, (g^r, g^{rx} \cdot M), M', g^{x'}, x')$ a polynomial time reproduction algorithm R returns $(g^r, (g^r)^{x'} \cdot M')$. It is easy to see that R always outputs a valid ciphertext which is created using the same random string as the given ciphertext and therefore the experiment $\mathbf{Exp}^{repr}_{\mathcal{EG},R}(k)$ always outputs 1.

The El Gamal scheme in a group of prime order is known to be IND-CPA under the assumption that the decision Diffie-Hellman (DDH) problem is hard. (This is noted in [12, 24, 13, 29]). Let $\mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{G},D}()$ denote the advantage of D in solving the Decisional Diffie-Hellman (DDH) problem for $\mathcal{G}$. We say that the DDH problem is hard for $\mathcal{G}$ if the function $\mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{G},D}(\cdot)$ is negligible for every algorithm D whose time-complexity is polynomial in k. (We recall the full formal definition for the DDH problem in [3].) Theorem 1 and Lemma 1 imply that it is also SRS-IND-CPA or, equivalently, $\overline{\mathcal{EG}}$ is IND-CPA secure and the security degrades linearly as the number of users n increases. The following theorem shows that it is possible to obtain a tighter relation than the one implied by Theorem 1.

Theorem 2. *Let $\mathcal{G}$ be a prime-order-group generator, $\mathcal{EG} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ the associated El Gamal encryption scheme, and $\overline{\mathcal{EG}} = (\mathcal{G}, \mathcal{K}, \overline{\mathcal{E}}, \mathcal{D})$ the associated SRS multi-recipient encryption scheme as per Construction 1. Let n be a polynomial. Then for any adversary B there exists a distinguisher D such that for any k*

$$\mathsf{Adv}^{n\text{-mr-cpa}}_{\overline{\mathcal{EG}},B}(k) \;\leq\; 2 \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{G},D}(k) + \frac{1}{2^{k-2}},$$

where the running time of D is one of B plus $O(n(k) \cdot k^3)$.

The proof of the above theorem is in the full version of this paper [4]. [22] proves a similar result but for a weaker notion of security of multi-recipient schemes.

```
$\underline{\mathcal{G}(k)}$:
$(q, g_1) \overset{R}{\leftarrow} \overline{\mathcal{G}}$
$g_2 \overset{R}{\leftarrow} G_q$
$K \overset{R}{\leftarrow} \mathcal{GH}(k)$
Return $(q, g_1, g_2, K)$
```

```
$\underline{\mathcal{K}(q, g_1, g_2, K)}$:
$x_1, x_2, y_1, y_2, z \overset{R}{\leftarrow} Z_q$
$c \leftarrow g_1^{x_1} g_2^{x_2}$ ; $d \leftarrow g_1^{y_1} g_2^{y_2}$
$h \leftarrow g_1^z$
$pk \leftarrow (g_1, g_2, c, d, h, K)$
$sk \leftarrow (x_1, x_2, y_1, y_2, z)$
Return $(pk, sk)$
```

```
$\underline{\mathcal{E}_{pk}(M)}$:
$r \overset{R}{\leftarrow} Z_q$
$u_1 \leftarrow g_1^r$ ; $u_2 \leftarrow g_2^r$
$e \leftarrow h^r M$
$\alpha \leftarrow \mathcal{EH}_K(u_1, u_2, e)$
$v \leftarrow c^r d^{r\alpha}$
Return $(u_1, u_2, e, v)$
```

```
$\underline{\mathcal{D}_{sk}(u_1, u_2, e, v)}$:
$\alpha \leftarrow \mathcal{EH}_K(u_1, u_2, e)$
If ${u_1}^{x_1 + y_1 \alpha} {u_2}^{x_2 + y_2 \alpha} = v$
    then $M \leftarrow e/{u_1}^z$
    else $M \leftarrow \perp$
EndIf
Return $M$
```

Fig. 1. Cramer-Shoup scheme

Cramer-Shoup. We now consider an SRS encryption scheme based on the Cramer-Shoup scheme [13] in order to get IND-CCA security properties. We first recall the Cramer-Shoup scheme. Let $\overline{\mathcal{G}}$ be a prime-order-group generator. The algorithms of the associated Cramer-Shoup scheme $\mathcal{CS} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ are depicted in Figure 1. The proof of the following lemma is in [4].

Lemma 2. *The Cramer-Shoup encryption scheme $\mathcal{CS} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ is reproducible.*

Let $\mathsf{Adv}^{cr}_{\mathcal{H},\mathcal{C}}(k)$ denote the advantage of an adversary C breaking collision-resistance of $\mathcal{H}$ (the full version [4] recalls the formal definition of collision resistance). If the DDH problem is hard for $\mathcal{G}$ and if $\mathcal{H}$ is collision-resistant then $\mathcal{CS}$ is IND-CCA secure [13]. Theorem 1 and Lemma 2 imply that it is also SRS-IND-CCA or, equivalently, $\overline{\mathcal{CS}}$ is IND-CCA secure. We match the result of [22] in getting a better security result than the one implied by Theorem 1 but we do it for a stronger notion of security of multi-recipient schemes. The following theorem states our improvement. The proof is in [4].

Theorem 3. *Let $\mathcal{G}$ be a prime-order-group generator, $\mathcal{CS} = (\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ the associated Cramer-Shoup encryption scheme and $\overline{\mathcal{CS}} = (\mathcal{G}, \mathcal{K}, \overline{\mathcal{E}}, \mathcal{D})$ the associated SRS multi-recipient encryption scheme as per Construction 1. Let n be a polynomial. Then for any adversary B, which makes q_{d} decryption oracle queries, there exists an adversary A, a distinguisher D and an adversary C such that for any k*

$$\mathsf{Adv}^{n\text{-mr-cca}}_{\overline{\mathcal{CS}},B}(k) \;\leq\; 2\mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{G},D}(k) + 2\mathsf{Adv}^{cr}_{\mathcal{H},\mathcal{C}}(k) + \frac{q_d(k)+2}{2^{k-3}},$$

and the running time of D and C is that of B plus $O(n(k) \cdot k^3)$.

DHIES. We consider another DDH-based encryption scheme, DHIES [2], which is in several draft standards. It combines public and symmetric key encryption methods, a message authentication code and a hash function and provides security against chosen-ciphertext attacks. Let $\mathsf{SE} = (\mathrm{K}, \mathrm{E}, \mathrm{D})$ be a symmetric encryption scheme with key length kl and let $\mathsf{MAC} = (\mathcal{T}, \mathcal{V})$ be a message authentication code with key length ml, tagging algorithm $\mathcal{T}$ and verification algo-

```
$\mathcal{E}_{(q,g,X)}(M)$:
$r \xleftarrow{R} Z_q$ ; $Y \leftarrow g^r$ ; $K \leftarrow H(X^r)$
Let $sk_m$ be the first $ml$ bits of $K$
Let $sk_e$ be the last $kl$ bits of $K$
$C \xleftarrow{R} \mathrm{E}_{sk_e}(M)$ ; $T \leftarrow \mathcal{T}_{sk_m}(C)$
Return $(Y,C,T)$
```

```
$\mathcal{D}_x(Y,C,T)$:
$K \leftarrow H(Y^x)$
Let $sk_m$ be the first $ml$ bits of $K$
Let $sk_e$ be the last $kl$ bits of $K$
$M \leftarrow \mathrm{D}_{sk_e}(C)$
If $\mathcal{V}_{sk_m}(M,T) = 1$ then Return $M$
else Return $\perp$ EndIf
```

Fig. 2. DHIES

rithm $\mathcal{V}$. Let $H\colon \{0,1\}^{gl} \to \{0,1\}^{ml+kl}$ be a function. We assume MAC is deterministic. The common key and key generation algorithms of $\mathcal{DHIES}[\mathsf{SE},H,\mathsf{MAC}] = (\mathcal{G},\mathcal{K},\mathcal{E},\mathcal{D})$ are the same as the ones of El Gamal encryption scheme. The rest of the algorithms are in Figure 2. The proof of the following is in [4].

Lemma 3. *Let the symmetric encryption scheme used by DHIES scheme be any block cipher such as AES in CBC mode (we will refer to it as $\mathcal{CBC}$ encryption scheme.) Then $\mathcal{DHIES}[\mathcal{CBC},H,\mathsf{MAC}] = (\mathcal{G},\mathcal{K},\mathcal{E},\mathcal{D})$ is reproducible.*

Escrow El Gamal. Boneh and Franklin [10] suggested the El Gamal encryption scheme with global escrow capabilities. The $\mathcal{EEG} = (\mathcal{G},\mathcal{K},\mathcal{E},\mathcal{D})$ scheme uses Weil pairing and is defined as follows. The algorithm $\mathcal{G}$ on input the security parameter k chooses a k-bit prime p such that $p \equiv 2 \bmod 3$ and $p \equiv 6q-1$ for some prime $q \geq 3$. Let E be the elliptic curve defined by $y^2 = x^3 + 1$ over F_p. Then it chooses a random $P \in E/\mathsf{F}_p$ of order q, computes $Q = sP$ for a random $s \in \mathsf{Z}_q^*$ and chooses a hash function $H\colon \mathsf{F}_{p^2} \to \{0,1\}^m$. The message space is $\{0,1\}^m$. The escrow key is s. $\mathcal{G}$ outputs (p,m,P,Q,H). The rest of the algorithms are as follows:

```
$\mathcal{K}(p,m,P,Q,H)$:
$x \xleftarrow{R} Z_q^*$ ; $X \leftarrow xP$
$pk \leftarrow (p,P,Q,X)$ ;
$sk \leftarrow (p,P,Q,x)$
Return $(pk,sk)$
```

```
$\mathcal{E}_{pk}(M)$:
$r \xleftarrow{R} Z_q^*$
$g \leftarrow \hat{e}(pk,Q)$
Return $(rP, M \oplus H(g^r))$
```

```
$\mathcal{D}_{sk}(U,V)$:
$M \leftarrow V \oplus H(\hat{e}(U,xQ))$
Return $M$
```

We do not define the decryption using the escrow key since it is not relevant for our goal. The proof is in [4].

Lemma 4. *The escrow El Gamal encryption scheme $\mathcal{EEG} = (\mathcal{G},\mathcal{K},\mathcal{E},\mathcal{D})$ is reproducible.*

A standard argument shows that $\mathcal{EEG}$ is IND-CPA secure in the random oracle model assuming Bilinear Diffie-Hellman assumption (see [10] for proper definitions). The results of Theorem 1 and Lemma 4 can be easily adjusted for the random oracle model and they would imply that $\mathcal{EEG}$ is also SRS-IND-CPA or, equivalently, the corresponding multi-recipient scheme $\overline{\mathcal{EEG}}$ is IND-CPA secure, both in the random oracle model.

Acknowledgements

We thank Diana Smetters for useful discussions. Part of this research has been done when Alexandra Boldyreva was in PARC. Mihir Bellare and Alexandra Boldyreva were supported in part by NSF Grant CCR-0098123 and NSF Grant ANR-0129617. Alexandra was also supported by SDSC Graduate Student Diversity Fellowship.

References

[1] O. Baudron, D. Pointcheval and J. Stern, "Extended notions of security for multicast public key cryptosystems." *ICALP 2000* 86, 87, 91, 92, 93

[2] M. Abdalla, M. Bellare, and P. Rogaway, "The Oracle Diffie-Hellman Assumptions and an Analysis of DHIES," *CT-RSA 01, Lecture Notes in Computer Science Vol. 2020, D. Naccache ed, Springer-Verlag, 2001.* 88, 96

[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 86, 87, 88, 89, 91, 92, 93, 95

[4] M. Bellare, A. Boldyreva, and J. Staddon "Randomness Re-Use in Multi-Recipient Encryption Schemes", Full version of this paper. Available at http://www-cse.ucsd.edu/users/aboldyre 89, 90, 94, 95, 96, 97

[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 and O. Goldreich, "On defining proofs of knowledge," *Advances in Cryptology – Crypto '92*, LNCS Vol. 740, E. Brickell ed., Springer-Verlag, 1992. 94

[7] S. Berkovits, "How to Broadcast a Secret", *Advances in Cryptology – Eurocrypt '91*, LNCS Vol. 547, D. Davies ed., Springer-Verlag, 1991.

[8] M. Blum and S. Micali, "How to generate cryptographically strong sequences of pseudo-random bits," *SIAM J. on Computing* Vol. 13, No. 4, November 1984.

[9] D. Boneh. "Simplified OAEP for the RSA and Rabin Functions," *Advances in Cryptology – Crypto '01*, LNCS Vol. 2139, J. Kilian ed., Springer-Verlag, 2001.

[10] D. Boneh and M. Franklin. "Identity-based encryption from the Weil Pairing," *Advances in Cryptology – Crypto '01*, LNCS Vol. 2139, J. Kilian ed., Springer-Verlag, 2001. 89, 97

[11] J. Camenisch and M. Michels, "Confirmer signature schemes secure against adaptive adversaries," *Advances in Cryptology – Eurocrypt '00*, LNCS Vol. 1807, B. Preneel ed., Springer-Verlag, 2000.

[12] R. Canetti,, "Towards Realizing Random Oracles: Hash Functions that Hide All Partial Information,", *Advances in Cryptology – Crypto '97*, LNCS Vol. 1294, B. Kaliski ed., Springer-Verlag, 1997 95

[13] 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. 86, 88, 95, 96

[14] T. ElGamal, "A public key cryptosystem and signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol 31, 1985.

[15] A. Fiat and M. Naor, "Broadcast Encryption", *Advances in Cryptology – Crypto '93*, LNCS Vol. 773, D. Stinson ed., Springer-Verlag, 1993.

[16] 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. 91

[17] S. Goldwasser and S. Micali, "Probabilistic encryption," *Journal of Computer and System Science*, Vol. 28, 1984, pp. 270–299.

[18] O. Goldreich, S. Goldwasser and S. Micali, "How to construct random functions,"*Journal of the ACM,* Vol. 33, No. 4, 210–217, (1986).

[19] J. Håstad, "Solving simultaneous modular equations of low degree," *SIAM J. on Computing* Vol. 17, No. 2, April 1988. 87

[20] J. Håstad, R. Impagliazzo, L. Levin, and M. Luby, "A pseudorandom generation from any one-way function ," *SIAM Journal on Computing,* Vol. 28, No. 4, 1364–1396, 1999.

[21] R. Impagliazzo and M. Luby, "One-way functions are essential for complexity based cryptography," Proceedings *of the* 30th *Symposium on Foundations of Computer Science*, IEEE, 1989

[22] K. Kurosawa, "Multi-Recipient Public-Key Encryption with Shortened Ciphertext," *Proceedings of the Fifth International workshop on practice and theory in Public Key Cryptography (PKC'02).* 86, 87, 88, 89, 92, 93, 95, 96

[23] S. Micali, C. Rackoff and R. H. Sloan, "The notion of security for probabilistic cryptosystems," *Advances in Cryptology – Crypto '86*, LNCS Vol. 263, A. Odlyzko ed., Springer-Verlag, 1986.

[24] 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. 88, 95

[25] "PKCS-1," RSA Labs, `http://www.rsasecurity.com/rsalabs/pkcs/pkcs-1/`. 91

[26] 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.

[27] V. Shoup, "On formal models for secure key exchange, " Theory of Cryptography Library Record 99-12, `http://philby.ucsd.edu/cryptolib/`. 88

[28] M. Stadler, "Publicly verifiable secret sharing," *Advances in Cryptology – Eurocrypt '96*, LNCS Vol. 1070, U. Maurer ed., Springer-Verlag, 1996. 88

[29] 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)*, Lecture Notes in Computer Science Vol. 1431, H. Imai and Y. Zheng eds., Springer-Verlag, 1998. 95

[30] D. Wallner, E. Harder and R. Agee, "Key Management for Multicast: Issues and Architectures," Internet Request for Comments, **2627** (June 1999). Available at: `ftp.ietf.org/rfc/rfc2627.txt.`

[31] A. C. Yao. "Theory and application of trapdoor functions," Proceedings *of the* 23rd *Symposium on Foundations of Computer Science*, IEEE, 1982