# Correlated Product Security From Any One-Way Function and the New Notion of Decisional Correlated Product Security

Brett Hemenway Steve Lu Rafail Ostrovsky

February 23, 2010

**Abstract**

It is well-known that the $k$-wise product of one-way functions remains one-way, but may no longer be when the $k$ inputs are correlated. At TCC 2009, Rosen and Segev introduced a new notion known as Correlated Product secure functions. These functions have the property that a $k$-wise product of them remains one-way even under correlated inputs. Rosen and Segev gave a construction of injective trapdoor functions which were correlated product secure from the existence of Lossy Trapdoor Functions (introduced by Peikert and Waters in STOC 2008).

The first main result of this work shows the surprising fact that a family of correlated product secure functions can be constructed from any one-way function. Because correlated product secure functions are trivially one-way, this shows an equivalence between the existence of these two cryptographic primitives.

In the second main result of this work, we consider a natural decisional variant of correlated product security. Roughly, a family of functions are Decisional Correlated Product (DCP) secure if $f_1(x_1), \ldots, f_k(x_1)$ is indistinguishable from $f_1(x_1), \ldots, f_k(x_k)$ when $x_1, \ldots, x_k$ are chosen uniformly at random.

We argue that the notion of Decisional Correlated Product security is a very natural one. To this end, we show a parallel from the Discrete Log Problem and Decision Diffie-Hellman Problem to Correlated Product security and its decisional variant. This intuition gives very simple constructions of PRGs and IND-CPA encryption from DCP secure functions. Furthermore, we strengthen our first result by showing that the existence of DCP secure one-way functions is also equivalent to the existence of any one-way function.

When considering DCP secure functions with trapdoors, we give a construction based on Lossy Trapdoor Functions, and show that any DCP secure function family with trapdoor satisfy the security requirements for Deterministic Encryption as defined by Bellare, Boldyreva and O'Neill in CRYPTO 2007. In fact, we also show that definitionally, DCP secure functions with trapdoors are a strict subset of Deterministic Encryption functions by showing an example of a Deterministic Encryption function which according to the definition is not a DCP secure function.

**Keywords:** Correlated Product Security, Lossy Trapdoor Functions, Deterministic Encryption

# 1 Introduction

In 2008, Peikert and Waters [PW08] introduced the notion of Lossy Trapdoor Functions (LTDFs), and used them as a building block to create encryption secure against a Chosen-Ciphertext Attack (CCA). In a recent result [RS09], Rosen and Segev considered a relaxation of the primitive that they called Correlated Product secure functions. Intuitively, these are families of one-way trapdoor functions whose $k$-wise products remain one-way even if the inputs on each coordinate are correlated. Rosen and Segev showed that the original Peikert-Waters Construction of IND-CCA secure encryption remained secure when instantiated with Correlated Product secure injective one-way trapdoor functions. They went on to show that Lossy Trapdoor Functions were Correlated Product secure, yet there was a black-box separation between LTDFs and Correlated Product secure one-way trapdoor *permutations*.

Correlated Product secure functions appear to be simpler to achieve than Lossy Trapdoor Functions which have a *statistical* lossiness requirement. Despite this appearance of relative simplicity there have been few examples of correlated product secure functions that are not Lossy Trapdoor Functions. The notable exceptions are the constructions given in [Pei09] and [FGK$^+$10].

In 2007, Bellare, Boldyreva, and O'Neill [BBO07] introduced a new notion known as Deterministic Encryption (DE). The deterministic property of the encryption affords the scheme many practical applications, such as searchable encryption, but at the same time requires new security definitions. Subsequent works [BFO08, BFOR08] demonstrate equivalences between various definitions of DE and show that the existence of a special kind of LTDFs imply the existence of deterministic encryption, which in turn implies the existence of IND-CCA secure cryptosystems.

## 1.1 Our Results

In this work, we introduce (in Section 3) a notion of Decisional Correlated Product (DCP) security, which strengthens the definition of Rosen and Segev. We argue that this is a natural stepping-stone between Lossy Trapdoor Functions and Correlated Product secure functions. Intuitively, these are families of functions such that for any $k$ functions $f_1, \ldots, f_k$, the distributions $\{(f_1(x_1), \ldots, f_k(x_1))\}$ and $\{(f_1(x_1), \ldots, f_k(x_k))\}$ are indistinguishable when $x_1, \ldots, x_k$ are chosen uniformly at random. Of course, a family of constant functions satisfies this definition, so for non-trivial results, we either specify that the functions be (individually) one-way or that they be injective with large domain. It turns out that, under either one of these assumptions, these families can be shown to also be Correlated Product secure. This is proven in Section 4 as the following lemmas:

**Lemma 2.** If $\mathcal{F} = (G, F)$ is a family of $k$-DCP secure functions with super-polynomial size domain that are injective, then $\mathcal{F}$ is $k$-correlated product secure.

**Lemma 3.** If $\mathcal{F} = (G, F)$ is a family of $k$-DCP secure one-way functions, then $\mathcal{F}$ is $k$-correlated product secure.

In Section 5, we look at a number theoretic connection between DCP security and regular CP security. Roughly speaking, we demonstrate that the function family $f_s(x) = s^x$ is CP secure if the Discrete Log assumption holds, and is DCP secure if the DDH assumption holds. Of independent interest, we show that the classical DDH-based pseudorandom generator can be generalized to use DCP secure functions. We show there exists a PRG with linear stretch that only makes 2 invocations to DCP secure *one-way permutations that are not necessarily trapdoor*.

Our first main result considers families of one-way functions that are DCP secure. We show that such families are automatically (plain) Correlated Product secure, and demonstrate a construction from any pseudorandom function family. Due to the celebrated fact that a PRF family can be constructed from any one-way function ([GGM86, ILL89, HILL99]), we obtain an equivalence between the existence of one-way functions, DCP secure one-way function families, and CP secure function families. This is proven in Section 6 as the following theorem:

**Theorem 1.** The following statements are equivalent:

1. One-way functions exist.
2. $k$-DCP secure families of one-way functions exist.
3. $k$-CP secure families of one-way functions exist.

It is interesting to note that Correlated Product secure functions without trapdoor are equivalent to one-way functions, while the recent results of Vahlis [Vah10] show that Correlated Product secure functions with trapdoor cannot be constructed from enhanced one-way trapdoor permutations.

Our second main result considers DCP secure function families which also have trapdoor. We investigate the connection between this and other primitives. In Section 7, we show a construction of these one-way trapdoor DCP secure families from "universally lossy" LTDFs[1]. This is stated as the following theorem:

**Theorem 2.** Let $\epsilon(\lambda)$ be any function such that $1/2^{\epsilon(\lambda)}$ is negligible in $\lambda$. Let $\mathcal{F} = (G, F)$ be a family of LTDFs on domain $\{0,1\}^\lambda$, where the lossy mode is universal and loses $\lambda/2 + \epsilon(\lambda)$ bits of the input. Then $\mathcal{F}$ is a 2-DCP secure injective trapdoor function.

Finally, in Section 8, we show that these families definitionally satisfy the security requirements of Deterministic Encryption, but the converse is not true in general. We informally have:

**Theorem 3 (Informal).** DCP secure function families with trapdoor are also PRIV1 secure deterministic encryption schemes.

### 1.2 Previous Work

In [PW08] Peikert and Waters introduced a new paradigm for constructing IND-CCA secure encryption based on the newly defined primitive Lossy Trapdoor Functions (LTDFs). Their construction of IND-CCA was natural and appealing, but LTDFs proved difficult to construct because of their strong statistical lossiness properties. Despite the power of LTDFs, in [PW08] they were able to give constructions from DDH and Lattice-based assumptions, and the authors of [BFOR08] and [RS08, FGK$^+$10] (independently) found identical efficient constructions of LTDFs from Paillier's Decisional Composite Residuousity Assumption.

In [RS09], Rosen and Segev examined which properties of LTDFs were necessary to construct IND-CCA secure encryption via the methods in [PW08]. With this goal, they defined Correlated Product secure functions, and gave a construction of IND-CCA secure encryption from Correlated Product secure functions with trapdoor paralleling the construction in [PW08]. One of the main difficulties in constructing Lossy Trapdoor Functions has been finding candidate functions where the lossy branch is *statistically* lossy (i.e. the image of the function is significantly smaller than the domain). Correlated Product secure functions do not have these statistical requirements, and thus should be easier to construct than LTDFs. This intuition was reinforced in [RS09] where they showed that LTDFs are Correlated Product secure, and showed a black-box separation in the opposite direction. Correlated Product secure functions remain difficult to realize, however, and the recent results of Vahlis [Vah10], show a black-box separation between (enhanced) one-way trapdoor permutations and Correlated Product Secure functions.

The works [BFO08, BFOR08] show many different relationships between DE and other primitives. Indeed, they show that any LTDF is almost immediately a DE scheme, and show how a weaker notion of DE can be constructed from any one-way trapdoor permutation.

[1] This slight addition was introduced in [BFO08] to construct DE from LTDFs. We describe this in detail when we give our construction.

## 2 Preliminaries

If $A$ is a PPT machine, then we use $a \xleftarrow{\$} A$ to denote running the machine $A$ and obtaining an output, where $a$ is distributed according to the internal randomness of $A$. For a PPT machine $A$, we use $\mathsf{coins}(A)$ to denote the distribution of the internal randomness of $A$. So the distributions $\{a \xleftarrow{\$} A\}$ and $\{r \xleftarrow{\$} \mathsf{coins}(A) : a = A(r)\}$ are identical. If $R$ is a set, we use $r \xleftarrow{\$} R$ to denote sampling uniformly from $R$.

If $X$ and $Y$ are families of distributions indexed by a security parameter $\lambda$, we use $X \approx_s Y$ to mean the distributions $X$ and $Y$ are statistically close, i.e. for all polynomials $p$ and sufficiently large $\lambda$ we have the statistical distance $\Delta(X, Y)$ is sufficiently small:

$$\Delta(X, Y) := \sum_x |\Pr[X = x] - \Pr[Y = x]| < \epsilon(\lambda),$$

where $\epsilon(\lambda)$ is a negligible function. We use $X \approx_c Y$ to mean $X$ and $Y$ are computationally close, i.e. for all PPT adversaries $A$, all polynomials $p$, and all sufficiently large $\lambda$,

$$\left|\Pr[A^X = 1] - \Pr[A^Y = 1]\right| < \epsilon(\lambda),$$

where $\epsilon(\lambda)$ is a negligible function.

We define what we mean by the $k$-wise product of a Function Family.

**Definition 1** ($k$-wise product)**.** Let $\mathcal{F} = (G, F)$ be a collection of efficiently computable functions. $G$ is a (randomized) algorithm which takes as input a size parameter $1^\lambda$ and generates a key (or seed) $s$ for $F$. Each function $F(s, \cdot)$ takes as input an element of some domain $X$ and outputs some value in the range $Y$, both of which implicitly depend on the parameter $\lambda$. For notational purposes, we also write $F_s(\cdot) = F(s, \cdot)$.
For $k \geq 2$, we define a family of $k$-wise products $\mathcal{F}^k = (G^k, F^k)$ as follows:

- **Key Generation:**
  $G^k(1^\lambda)$ independently generates $s_i \xleftarrow{\$} G(1^\lambda)$, for $i = 1, \ldots, k$.
- **Evaluation:**
  To evaluate $F^k$ on input $((s_1, \ldots, s_k), (x_1, \ldots, x_k))$, we define

$$F^k((s_1, \ldots, s_k), (x_1, \ldots, x_k)) = (F_{s_1}(x_1), \ldots, F_{s_k}(x_k)).$$

**Definition 2** (Pairwise Independence)**.** A family of functions $\mathcal{H}$ such that $h : X \to Y$ is called *pairwise independent* if for all $x_1 \neq x_2$ in $X$ and for all $y_1, y_2 \in Y$, we have

$$\Pr_{h \xleftarrow{\$} \mathcal{H}}[h(x_1) = y_1 \text{ and } h(x_2) = y_2] = \frac{1}{|Y|^2}.$$

In particular this says that if $h$ is chosen uniformly from $\mathcal{H}$, then $h(x_1)$ and $h(x_2)$ are uniformly and independently distributed for all $x_1 \neq x_2$.

We remind the reader of the leftover hash lemma [HILL99, ILL89, IZ89], which states that a pairwise independent hash function acts as a strong extractor. In particular the hash function "smooths out" a min-entropy source to look nearly uniform.

**Lemma 1** (Leftover Hash Lemma)**.** Let $\mathcal{H}$ be a pairwise independent hash family, such that for all $h \in \mathcal{H}$, $h : X \to Y$. Let $D_X$ be a distribution over $X$ such that the min entropy $H_\infty(D_X) \geq \log|Y| + 2\log(1/\epsilon)$. Then if we define $\Lambda_1 = \{h \xleftarrow{\$} \mathcal{H}; x \xleftarrow{\$} D_X : (h, h(x))\}$, and $\Lambda_2 = \{h \xleftarrow{\$} \mathcal{H}; y \xleftarrow{\$} Y : (h, y)\}$, we have $\Delta(\Lambda_1, \Lambda_2) \leq \epsilon$.

### 2.1 Correlated Product Security

We review the definition of Correlated Product security, first defined in [RS09].

**Definition 3** (Correlated Product Security)**.** Let $\mathcal{F} = (G, F)$ be a collection of efficiently computable functions. Let $C_k = C_k(1^\lambda)$ be a distribution. We say that $\mathcal{F}$ is secure under $C_k$-correlated products if $\mathcal{F}_k$ is one-way with respect to the input distribution $C_k$. We henceforth only consider the case where $C_k$ is the uniform $k$-repetition distribution, i.e. $k$ copies of a uniformly chosen input.

We refer the reader to the Appendix for reminders of the definitions of the Discrete Log and DDH assumptions, Deterministic Encryption, Lossy Trapdoor Functions, and Pseudorandom Functions.

## 3 Decisional Correlated Product Security

In this work we introduce the notion of Decisional Correlated Product (DCP) security, which can be viewed as the decisional variant of Correlated Product security introduced in [RS09]. In [RS09], they primarily took $C_k$ to be the uniform $k$-repetition distribution, i.e. $C_k$ uniformly samples $x$ and outputs $k$ copies of $x$. We will also primarily focus on the $k$-repetition distribution, although we will consider a decisional variant of the problem.

First, we remark that Correlated Product security seems to be a much stronger notion than simply one-wayness. For example, the map $f_e : x \mapsto x^e \mod n$, is one-way trapdoor permutation under the RSA assumption. However, given $f_{e_1}(x), f_{e_2}(x)$, if $\gcd(e_1, e_2) = 1$, we can immediately recover $x$, by using the extended Euclidean algorithm to calculate $s, t$ such that $se_1 + te_2 = 1$, and noticing that $(x^{e_1})^s (x^{e_2})^t = x$. This example also shows that Decisional Correlated Product security does not follow immediately from Computational Correlated Product security, because if $d_1, d_2, d_3$ are relatively prime, and $e_i = ed_i$ for some fixed $e$, then $f_{e_1}, f_{e_2}, f_{e_3}$ will be Computationally Correlated Product secure under the RSA assumption, but will not be Decisional Correlated Product secure by a similar argument.

**Definition 4** (Decisional Correlated Product Security)**.** Let $\mathcal{F} = (G, F)$ be a collection of efficiently computable functions. We say that $\mathcal{F}^k$ is *$k$-wise Decisional Correlated Product secure* if for all efficient PPT adversaries $A$,

$$\left| \Pr\left[A^{\mathsf{indepdist}} = 1\right] - \Pr\left[A^{\mathsf{repdist}} = 1\right] \right| < \nu$$

for some negligible function $\nu$, and where the games $\mathsf{indepdist}$ and $\mathsf{repdist}$ are defined as in Figure 1.

| **Independent** | **Repetition** |
|---|---|
| $s_1 \xleftarrow{\$} G(1^\lambda), \ldots, s_k \xleftarrow{\$} G(1^\lambda)$ | $s_1 \xleftarrow{\$} G(1^\lambda), \ldots, s_k \xleftarrow{\$} G(1^\lambda)$ |
| $x_1 \xleftarrow{\$} X, \ldots, x_k \xleftarrow{\$} X$ | $x \xleftarrow{\$} X$ |
| $b \xleftarrow{\$} A(s_1, \ldots, s_k, F_{s_1}(x_1), \ldots, F_{s_k}(x_k))$ | $b \xleftarrow{\$} A(s_1, \ldots, s_k, F_{s_1}(x), \ldots, F_{s_k}(x))$ |
| Return $b$ | Return $b$ |

Figure 1: Decisional Correlated Product Security

To illustrate the power of this definition, we construct a very natural $\mathsf{IND\text{-}CPA}$ secure encryption from any family of 2-DCP secure injective trapdoor functions. Let the public key be $F_1, F_2, h$ where $h$ is a pairwise independent hash function. Define encryption as $E(m, r) = (F_1(r), h(F_2(r)) \oplus m)$. To decrypt, we simply invert $F_1$ to recover $r$, from this we can recover $h(F_2(r))$ and recover the message. If $F_i$ have domain $\{0,1\}^\lambda$, and $h$ maps from the range of $F_i$ to $\{0,1\}^{\lambda/2}$, then the leftover hash lemma tells us that $(F_1(r_1), h(F_2(r_2)) \oplus m)$ is statistically close to $(F_1(r_1), h(F_2(r_2)))$. So if $y_0, y_1$ are chosen

from the repetition-distribution $(y_0, h(y_1) \oplus m)$ is a valid ciphertext, while if $(y_0, y_1)$ are chosen from the independent distribution $(y_0, h(y_1) \oplus m)$ is independent of $m$, thus this scheme will be IND-CPA secure. We emphasize that this is not one of our main results, but simply an illustration of a natural construction that follows from this definition.

The notion of *Decisional* Correlated Product security is clearly a stronger notion than the (Computational) Correlated Product security defined in [RS09] for *injective functions*. In the next section, we examine under what conditions DCP security implies CP security.

# 4 Relations to (Computational) Correlated Product Security

The notion of $k$-DCP security seems like a stronger requirement than Computational Correlated Product security, but we observe that if we do not put any requirements on the functions, then $k$-DCP security may be satisfied by trivial functions. For example the constant functions are trivially $k$-DCP for any $k \geq 2$. The following lemmas give sufficient conditions for when a $k$-DCP secure family is $k$-correlated product secure.

**Lemma 2.** If $\mathcal{F} = (G, F)$ is a family of $k$-DCP secure functions with super-polynomial size domain and are injective, then $\mathcal{F}$ is $k$-correlated product secure.

*Proof.* Let $A$ be an efficient adversary that given $s_1, \ldots, s_k$, and $(F_{s_1}(x), \ldots, F_{s_k}(x))$, finds the inverse $(x'_1, \ldots, x'_k) = (x, x, \ldots, x)$ with non-negligible probability $\epsilon$, we exhibit an efficient distinguisher $D$ that uses $A$ to break the $k$-DCP security of $\mathcal{F}$.

**Algorithm 1** $D(s_1, \ldots, s_k, y_1, \ldots, y_k)$

$(x'_1, \ldots, x'_k) \xleftarrow{\$} A(s_1, \ldots, s_k, y_1, \ldots, y_k)$
**if** $x'_1 = x'_2 = \cdots = x'_k$ and $F_{s_i}(x'_i) = y_i$ for $i \in [k]$ **then**
  **return** 1
**else**
  **return** 0
**end if**

We must analyze the probability that $D$ outputs 1 in the repdist and indepdist games.

$$\begin{aligned}\Pr[D^{\mathsf{repdist}} = 1] &= \Pr[x'_1 = \cdots = x'_k \wedge F_{s_i}(x'_i) = y_i | \\ &\qquad x \xleftarrow{\$} X, s_i \xleftarrow{\$} G(1^\lambda), y_i = F_{s_i}(x), \{x'_i\}_{i=1}^k \xleftarrow{\$} A(\{s_i\}_{i=1}^k, \{y_i\}_{i=1}^k)] \\ &= \Pr[A \text{ successfully inverts}] = \epsilon.\end{aligned}$$

$$\begin{aligned}\Pr[D^{\mathsf{indepdist}} = 1] &= \Pr[x'_1 = \cdots = x'_k \wedge f_{s_i}(x'_i) = y_i | \\ &\qquad x_i \xleftarrow{\$} X, s_i \xleftarrow{\$} G(1^\lambda), y_i = F_{s_i}(x_i), \{x'_i\}_{i=1}^k \xleftarrow{\$} A(\{s_i\}_{i=1}^k, \{y_i\}_{i=1}^k)] \\ &= \Pr[x'_1 = \cdots = x'_k \wedge x'_i = x_i | \\ &\qquad x_i \xleftarrow{\$} X, s_i \xleftarrow{\$} G(1^\lambda), y_i = F_{s_i}(x_i), \{x'_i\}_{i=1}^k \xleftarrow{\$} A(\{s_i\}_{i=1}^k, \{y_i\}_{i=1}^k)] \\ &\leq \Pr[x_1 = x_2 | x_i \xleftarrow{\$} X] \leq \frac{1}{|X|}.\end{aligned}$$

Thus the difference $|\Pr[D^{\mathsf{repdist}} = 1] - \Pr[D^{\mathsf{indepdist}} = 1]| \geq \epsilon - \frac{1}{|X|}$ is non-negligible, as $|X|$ is super-polynomial.

□

Next, we show that if a family $\mathcal{F} = (G, F)$ is a DCP secure, and each function is *individually* one-way, then the family is also Correlated Product secure.

**Lemma 3.** If $\mathcal{F} = (G, F)$ is a family of $k$-DCP secure one-way functions, then $\mathcal{F}$ is $k$-correlated product secure.

*Proof.* Suppose on the contrary that they were not. Let $A$ be a PPT algorithm that breaks the correlated product security of $(G, F)$, in particular given $\{s_1, \ldots, s_k, F_{s_1}(x_1), \ldots, F_{s_k}(x_1)\}$ $A$ is able to find a pre-image $(x'_1, \ldots, x'_k)$ with some non-negligible probability $\epsilon$, where the $s_i$ are generated by $G$ at random, and $x_1$ is chosen uniformly at random. We use $A$ to build a PPT distinguisher $D$ that can win in the $k$-DCP game.

**Algorithm 2** $D(s_1, \ldots, s_k, y_1, \ldots, y_k)$

$(x'_1, \ldots, x'_k) \xleftarrow{\$} A(s_1, \ldots, s_k, y_1, \ldots, y_k)$
**if** $F_{s_i}(x'_i) = y_i$ for $i \in [k]$ **then**
  **return** 1
**else**
  **return** 0
**end if**

We analyze the probability that $D$ outputs 1. If indeed the inputs are correlated, i.e. $y_i = F_{s_i}(x_1)$, then $A$ succeeds with probability $\epsilon$ and so $D$ will output 1 with that probability.

On the other hand, if the inputs are random and independent, i.e. $y_i = F_{s_i}(x_i)$, then $(x_1, \ldots, x_k)$ is a uniformly chosen input from the product space. Because each $F_{s_i}(\cdot)$ is a one-way function, the product function $(F_{s_1}(\cdot), \ldots, F_{s_k}(\cdot))$ is also one-way. Since the inputs are uncorrelated, the probability that $A$ inverts it on a random value is negligible. Thus, in this case, $D$ outputs 1 with only negligible probability.

This contradicts the $k$-DCP security of $(G, F)$. □

Many of the results in this work will focus on the case where the family $\mathcal{F}$ are in fact injective, or injective with trapdoor, and so the Correlated Product security will follow immediately from the DCP security of $\mathcal{F}$.

# 5 A Number Theoretic Parallel

One way to illustrate the connection between Correlated Product security and DCP is to examine their relationship to the Discrete Log Problem, which will lead us to novel and efficient constructions of DCP secure functions.

Let $\mathcal{G} = \mathcal{G}(\lambda)$ be a family of cyclic groups indexed by the security parameter $\lambda$, where $|\mathcal{G}| = \ell = \ell(\lambda)$.

**Lemma 4.** If the Discrete Log Problem is hard in $\mathcal{G}$, then the family in Figure 2 forms a family of Correlated Product secure functions with respect to the uniform $k$-repetition distribution:

| **Key Generation:** | **Evaluation:** |
|---|---|
| $G(1^\lambda)$ will select an element $s$ uniformly from $\mathcal{G}$ | $F_s(\cdot) : [\ell] \to \mathcal{G}$<br>$F_s(x) = s^x \in \mathcal{G}$ |

Figure 2: Decisional Correlated Product Security from the Discrete Log Problem

*Proof.* Suppose there exists an efficient adversary $A$ that succeeds in breaking the uniform $k$-repetition Correlated Product security of this family of functions. We will use $A$ to solve the Discrete Log Problem in $\mathcal{G}$. Suppose we are given $g, y \in \mathcal{G}$, and we would like to find $x$ such that $g^x = y$.

We will generate $a_1, \ldots, a_k$ uniformly from $[\ell]$, and set $s_i = g^{a_i} \in \mathcal{G}$. Since $g$ generates $\mathcal{G}$, the $s_i$ will be uniformly distributed in $\mathcal{G}$. Then we give $A$ the vector $(s_1, \ldots, s_k, y^{a_1}, \ldots, y^{a_k})$. Since

$$y^{a_i} = (g^x)^{a_i} = (g^{a_i})^x = s_i^x,$$

we have

$$(s_1, \ldots, s_k, y^{a_1}, \ldots, y^{a_k}) = (s_1, \ldots, s_k, F_{s_1}(x), \ldots, F_{s_k}(x)),$$

so if $A$ succeeds in inverting $\mathcal{F}_k$, we learn the discrete log of $y$ to the base $g$. □

Notice, however, that we cannot apply the results of [RS09] to get a Chosen-Ciphertext secure cryptosystem based on the Discrete Log Problem because this construction does not have a trapdoor (although it is injective).

Now, we will show a connection between the Decisional Diffie-Hellman problem (DDH) and DCP secure functions.

We begin by showing that the DDH assumption is almost identical to the 2-DCP assumption.

**Lemma 5.** If the DDH problem is hard in $\mathcal{G}$, then the construction in Figure 2 forms a family of 2-DCP functions.

*Proof.* Suppose there exists an efficient adversary $A$ that succeeds in distinguishing the uniform independent distribution from the 2-repetition distribution with probability $\frac{1}{2} + \epsilon$. We will use $A$ to solve the DDH problem in a cyclic group $\mathcal{G}$ or order $\ell$. The DDH challenger provides a 4-tuple $(g, h_1, h_2, h_3) \in \mathcal{G}^4$, where $h_1 = g^a$, $h_2 = g^b$, and $h_3 = g^c$ where we would like to determine whether $c = ab \mod \ell$.

We will generate $r$ uniformly from $[\ell]$, and set $s_1 = g^r$, and $s_2 = h_1^r$. Then we give $A$ the vector $(s_1, s_2, h_2^r, h_3^r)$. Now, if $c = ab$, then

$$(g^r, h_1^r, h_2^r, h_3^r) = (g^r, (g^a)^r, (g^b)^r, (g^{ab})^r) = (g^r, g^{ar}, (g^r)^b, (g^{ar})^b) = (s_1, s_2, s_1^b, s_2^b) = (s_1, s_2, F_{s_1}(b), F_{s_2}(b)).$$

On the other hand, if $c$ is uniformly distributed in $\ell$, then

$$(s_1, s_2, h_2^r, h_3^r) = (s_1, s_2, F_{s_1}(b), F_{s_2}(a^{-1}c \bmod \ell)),$$

and $a^{-1}c \bmod \ell$ will be uniform modulo $\ell$ as long as $\gcd(a, \ell) = 1$. Thus, by outputting the same output as $A$, we succeed in solving the DDH problem with probability $\epsilon$. □

We would like to extend this proof to show that if the DDH problem is hard in $\mathcal{G}$, then the construction above gives a family of $k$-DCP functions for any polynomial sized $0 < k \in \mathbb{Z}$. This is true, but requires one additional observation.

**Lemma 6.** If the DDH problem is hard in $\mathcal{G}$, then the construction in Figure 2 forms a family of $k$-DCP functions for any $0 < k \in \mathbb{Z}$.

*Proof.* Suppose there exists an efficient adversary $A$ that succeeds in distinguishing the uniform $k$-independent distribution from the uniform $k$-repetition distribution with probability $\frac{1}{2} + \epsilon$. We will use $A$ to solve the DDH problem in $\mathcal{G}$. Suppose we are given $g, h_1, h_2, h_3 \in \mathcal{G}$, where $h_1 = g^a$, $h_2 = g^b$, and $h_3 = g^c$ where we would like to determine whether $c = ab \bmod \ell$.

We will generate $r_1, \ldots, r_k$ and $r'_1, \ldots, r'_k$ uniformly from $[\ell]$, and set

$$s_i = g^{r_i} h_1^{r'_i} = g^{r_i + a r'_i} \in \mathcal{G}.$$

It is easy to see that the $s_i$ are distributed uniformly and independently in $\mathcal{G}$. Now, let

$$y_i = h_2^{r_i} h_3^{r'_i} = g^{br_i + cr'_i} \in \mathcal{G},$$

and give $A$ the vector

$$(s_1, \ldots, s_k, y_1, \ldots, y_k).$$

If $c = ab$, then

$$y_i = g^{br_i + abr'_i} = \left(g^{r_i + ar'_i}\right)^b = s_i^b \in \mathcal{G},$$

which is a valid sampling of $\mathcal{F}_k$ on the uniform $k$-repetition distribution. On the other hand, if $c$ is uniformly distributed in $\ell$, then , thus

$$y_i = g^{br_i + cr'_i} = s_i^{(r_i + ar'_i)^{-1}(br_i + cr'_i) \bmod \ell} \in \mathcal{G}.$$

It is easy to see that if $c \neq ab$, then for distinct $i$ the exponents are uniformly and independently distributed modulo $\ell$. So, if $A$ guesses that the sample is from the uniform $k$-repetition distribution, we guess $c = ab$, and if $A$ guesses the sample comes from the uniform $k$-independent distribution, we guess that $c \neq ab$, by the above argument we are correct with the same probability that $A$ is, i.e. with probability $\frac{1}{2} + \epsilon$. □

*Remark.* The definition of $k$-DCP abstracts one of the most important properties of the DDH assumption. To see this, recall a simple DDH-based PRG. The description of the function is the group $\mathcal{G}$, and two elements $g, g^a$, and $f(b) = (g^b, (g^a)^b)$. The first element of the output is uniform if $b$ is uniform, and the pair is indistinguishable from uniform by the DDH assumption. Now, it is easy to see that this construction will go through as before with an injective $k$-DCP family of functions. In particular, the description of the PRG will be $\mathcal{F}, s_1, \ldots, s_k$, and $f(x) = F_{s_1}(x), \ldots, F_{s_k}(x)$. If $F_{s_i}(\cdot)$ is a permutation, this will be a PRG as it is. If the $F_{s_i}(\cdot)$ are merely injective, we will have to apply an extractor to "smooth" the output, but the proof of security remains exactly the same as in the DDH case.

# 6 Equivalence of OWF and (Decisional) Correlated Product secure families of OWFs

In this section, we aim to prove the main theorem relating the existence of OWFs to that of (Decisional) Correlated Product secure OWF families.

**Theorem 1.** The following statements are equivalent:

1. One-way functions exist.
2. $k$-DCP secure families of one-way functions exist.
3. $k$-CP secure families of one-way functions exist.

To do this, we first show how to construct a DCP secure family of one-way functions from any pseudorandom function family. The surprising idea is that a PRF family becomes DCP secure if we swap what we call the seed, and what we call the input. If the PRF output is sufficiently long, then the resulting functions are also one-way, thus we have a family of DCP secure one-way functions. The exact lengths necessary are given in Lemma 8.

We then show that DCP secure one-way function families are also (ordinary) CP secure. This will follow directly from the fact that a product of one-way functions remain one-way under uniform independent inputs (Lemma 3). Finally, CP secure OWF families obviously are one-way, which completes the cycle of implications.

Let $(\mathsf{PRFGen}, \mathsf{PRF})$ be a PRF family, such that if $s \xleftarrow{\$} \mathsf{PRFGen}(1^\lambda)$, with $s \in \{0,1\}^{w(\lambda)}$ then the domain of

$$\mathsf{PRF}(s, \cdot) : \{0,1\}^{n(\lambda)} \to \{0,1\}^{\ell(\lambda)}.$$

We can define a DCP family $(G, F)$, by

- **Sampling:** $G(1^\lambda)$ outputs a uniform value in $\{0,1\}^{\ell(\lambda)}$.
- **Evaluation:** For any $s \in \{0,1\}^{n(\lambda)}$,

$$F_s(\cdot) : \{0,1\}^{w(\lambda)} \to \{0,1\}^{\ell(\lambda)}$$
$$x \mapsto \mathsf{PRF}(x, s).$$

**Lemma 7.** $(G, F)$ forms a $k$-Decisional Correlated Product secure function family for any $k = poly(\lambda)$.

*Proof.* Define the distributions $\Lambda_0, \Lambda_1$ by sampling $s_1, \ldots, s_k \xleftarrow{\$} G(1^\lambda)$, and $x_1, \ldots, x_k \xleftarrow{\$} \{0,1\}^{w(\lambda)}$

$$\Lambda_0 = \{s_1, \ldots, s_k, F_{s_1}(x_1), \ldots, F_{s_k}(x_1)\}$$
$$\Lambda_1 = \{s_1, \ldots, s_k, F_{s_1}(x_1), F_{s_2}(x_2), \ldots, F_{s_k}(x_k)\}$$

Thus we must show that any adversary who can distinguish $\Lambda_0$ from $\Lambda_1$ can distinguish the underlying Pseudorandom Function from a truly random function.

Now, by the definition of $F$, we have

$$\Lambda_0 = \{s_1, \ldots, s_k, F_{s_1}(x_1), \ldots, \ldots, F_{s_k}(x_1)\} = \{s_1, \ldots, s_k, \mathsf{PRF}(x_1, s_1), \ldots, \mathsf{PRF}(x_1, s_k)\},$$
$$\Lambda_1 = \{s_1, \ldots, s_k, F_{s_1}(x_1), \ldots, \ldots, F_{s_k}(x_k)\} = \{s_1, \ldots, s_k, \mathsf{PRF}(x_1, s_1), \ldots, \mathsf{PRF}(x_k, s_k)\}.$$

Now, it is clear that the security of the Pseudorandom Function gives

$$\Lambda_0 \approx_c \{s_1, \ldots, s_k, U_{\ell(\lambda)}, \ldots, U_{\ell(\lambda)}\} \approx_c \Lambda_1,$$

which gives the result. □

**Lemma 8.** If the size of the key space of $F$ is a negligible fraction of the size of the output space, i.e. $1/2^{\ell(\lambda)-w(\lambda)}$ is negligible in $\lambda$, then $(G, F)$ forms a family of one-way functions.

*Proof.* Suppose to the contrary that for some key $s$, the function $F_s(\cdot)$ was not one-way. Let $A$ be a PPT inverter that succeeds with non-negligible probability $\epsilon$, i.e.

$$\Pr_x[F_s(z) = F_s(x) | z \leftarrow A(F_s(x))] = \epsilon$$

We use $A$ to construct a PPT algorithm $B$ that distinguishes between oracle access to $\mathsf{PRF}$ (with a randomly chosen seed $x$) and a truly random function $\mathcal{RO}$. The algorithm queries $s$ on the oracle, and receives $y$, which is either $y = \mathsf{PRF}(x, s) = F_s(x)$ for some $x$, or a truly random value. The distinguisher $B$ runs $A$ on $y$, and receives some output $x'$. If it is the case that $F_s(x') = y$, then $B$ outputs 1, otherwise $B$ outputs 0.

We analyze the probabilities $\Pr[B^{\mathcal{RO}(\cdot)} = 1]$ and $\Pr_x[B^{\mathsf{PRF}(x,\cdot)} = 1]$. In the former case, the probability that a random value is in the range of $\mathsf{PRF}(s, \cdot)$ is $\frac{|Range|}{2^\ell} \leq \frac{2^w}{2^\ell}$ which we assumed to be negligible. On the other hand,

$$\begin{aligned}\Pr_x[B^{\mathsf{PRF}(x,\cdot)} = 1] &= \Pr_x[\mathsf{PRF}(z, s) = y | z \leftarrow A(y)] \\ &= \Pr_x[\mathsf{PRF}(z, s) = \mathsf{PRF}(x, s) | z \leftarrow A(\mathsf{PRF}(x, s))] \\ &= \Pr_x[F_s(z) = F_s(x) | z \leftarrow A(F_s(x))] = \epsilon\end{aligned}$$

This contradicts the pseudorandomness of $\mathsf{PRF}$. □

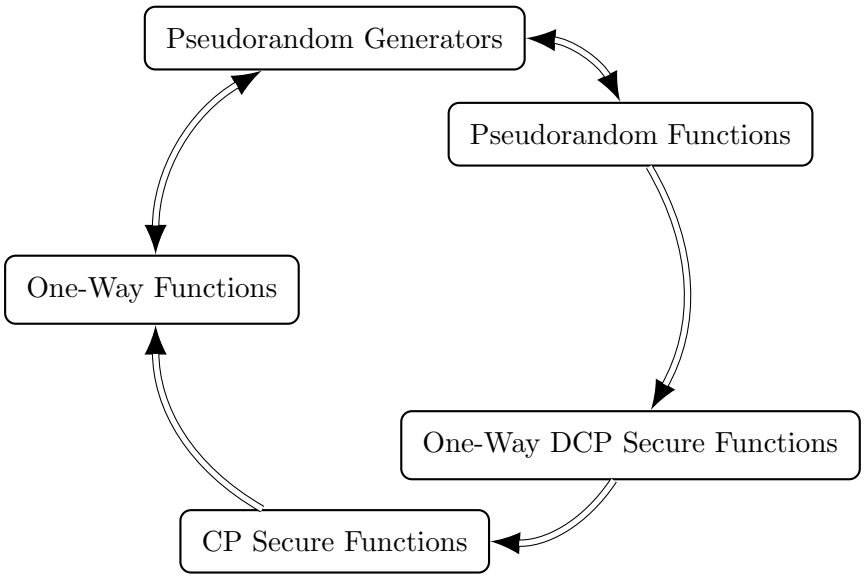
**Corollary 1.** One-way functions imply $k$-DCP secure one-way function families.

*Proof.* In Hastad, Impagliazzo, Levin and Luby [HILL99] it was shown that one-way functions imply PRGs, and in Goldreich, Goldwasser, Micali [GGM86] it was shown that PRGs imply the existence of PRF families with sufficiently long output, thus combining these results with our result, we have one-way functions imply $k$-DCP secure one-way functions. □

**Corollary 2.** One-way functions imply $k$-CP secure function families.

*Proof.* This follows immediately from applying Lemma 3 to Corollary 1. □

Since every Correlated Product secure function family is trivially a one-way function family, we find that we have




# 7 DCP with trapdoor from Lossy Trapdoor Functions

In the preceding sections, we examined DCP secure functions without trapdoors, and showed that one-way DCP secure functions *without trapdoor* could be constructed from any one-way function. Now, we show constructions of DCP with trapdoor. In particular, in this section, we show that lossy trapdoor functions (with certain constraints) are, in fact, DCP secure injective trapdoor functions. We remark that these are the same constraints on the LTDF which were used in [BFO08] to show that LTDFs are secure DE. These constraints will allow us to show that the LTDFs are also DCP secure.

The two constraints are that the lossy mode loses sufficiently many bits, and that it is "universal". The LTDF has a "universal lossy mode" if the lossy mode induces a pairwise independent hash function on its range. The main idea in showing LTDFs are DCP secure is that these constraints makes the lossy mode satisfy the premises of the leftover hash lemma, thereby allowing us to break the correlation of the inputs.

**Theorem 2.** Let $\epsilon(\lambda)$ be any function such that $1/2^{\epsilon(\lambda)}$ is negligible in $\lambda$. Let $\mathcal{F} = (G, F)$ be a family of LTDFs on domain $\{0,1\}^\lambda$, where the lossy mode is universal and loses $\lambda/2 + \epsilon(\lambda)$ bits of the input. Then $\mathcal{F}$ is a 2-DCP secure injective trapdoor function.

*Proof.* We show that distributions $\{s_1, s_2, F_{s_1}(x), F_{s_2}(x)\}$ and $\{s_1, s_2, F_{s_1}(x), F_{s_2}(y)\}$ are computationally indistinguishable, where $s_1, s_2 \xleftarrow{\$} G(1^\lambda)$, and $x, y$ are sampled uniformly at random from the domain.

The idea is that we view the pair $(s_1, s_2)$ as a single key for the (product) hash function $(F_{s_1}, F_{s_2})$. First, by the indistinguishability of ordinary keys from lossy ones, we have $(s_1, s_2, F_{s_1}(x), F_{s_2}(x)) \approx_c (s_1', s_2', F_{s_1'}(x), F_{s_2'}(x))$, where $s_1'$ and $s_2'$ are now lossy keys. By the universality of lossy mode, $F_{s_1'}$ and $F_{s_2'}$ are pairwise independent on their respective ranges $Y_1$ and $Y_2$. The product $(F_{s_1'}, F_{s_2'})$ is a pairwise independent hash function on $Y_1 \times Y_2$. The input distribution $X$ is the diagonal distribution on $\{0,1\}^\lambda \times \{0,1\}^\lambda$, which has $\lambda$ bits of (min) entropy. We have that

$$|Y_1 \times Y_2| + 2\log\left(\frac{1}{1/2^\epsilon}\right) = 2(\lambda/2 - \epsilon) + 2\epsilon = \lambda \leq H_\infty(X).$$

Therefore, by the leftover hash lemma, $\Delta(\{s_1', s_2', F_{s_1'}(x), F_{s_2'}(x)\}, \{s_1', s_2', y_1, y_2\}) \leq 1/2^\epsilon$, where $y_1$ and $y_2$ are uniform on the ranges $Y_1$ and $Y_2$ respectively. Finally, because the lossy functions are pairwise independent, $(s_1', s_2', y_1, y_2)$ is indistinguishable from $(s_1', s_2', F_{s_1'}(x), F_{s_2'}(y))$, which in turn is indistinguishable from $(s_1, s_2, F_{s_1}(x), F_{s_2}(y))$, where now the keys are for the injective functions. □

# 8 Decisional Correlated Product Security is Deterministic Encryption

In this section, we examine the consequences of DCP secure functions, again *with trapdoor*. We show that any 2-DCP secure functions with trapdoor are – almost without modification – a PRIV1 secure uniform deterministic encryption. We follow the terminology of [BFOR08], where a uniform deterministic encryption is one which is only guaranteed to be secure against message adversaries that choose messages from the uniform distribution, instead of simply any high min-entropy distribution.

Let $\mathcal{F} = (G, F)$ be a family of 2-Decisional Correlated Product secure Functions.

We can define a (Uniform) Deterministic Encryption by

| **KeyGen:** | **Encryption:** | **Decryption:** |
|---|---|---|
| $(s,t) \xleftarrow{\$} G(1^\lambda)$ | $E(pk, m) = F_{pk}(m)$ | $D(sk, c) = F_t^{-1}(c)$ |
| $pk = s$, $sk = t$ | | |

Figure 3: Decisional Correlated Product Secure functions with trapdoor are PRIV1 secure

**Theorem 3.** The scheme outlined in Figure 3 is BB-CSS secure.

*Proof.* First, we recall the notion of BB-CSS (Balanced Boolean Comparison-based Semantic Security) as defined in [BFOR08]. This is similar to the Comparison Semantic Security PRIV1, outlined by the games privreal and prividеal, except that the side information $t$ is required to be a balanced boolean function, i.e. $\Pr[t = 0] \approx \Pr[t = 1] \approx \frac{1}{2}$.

For simplicity, we assume that $\Pr[t = 0] = \Pr[t = 1] = \frac{1}{2}$, but it is easy to see that if the distributions are only negligibly close to $\frac{1}{2}$ then the argument goes through as well.

Notice that in this setting *any* adversary has a $\frac{1}{2}$ chance of winning in the prividеal game since his view is independent of the actual side information, thus it is enough to consider the adversary's probability of winning in the privreal game.

Now, suppose there exists an adversary $A = (A_m, A_g)$, such that $(m, t) \xleftarrow{\$} A_m(1^\lambda)$, where $m$ is uniform on $X$ the domain of $f_s$, and $t$ is uniform on $\{0,1\}$. The guessing adversary $A_g$ on input $pk, c$ outputs a guess $t'$. If $c = E(Pk, m)$, then $\Pr[t = t'] = \frac{1}{2} + \epsilon$.

We show how to use $A$ to create a distinguisher $D$ that can distinguish the 2-repetition distribution from the 2-independent distribution. The algorithm $D$ takes as input the description of two functions $s_0, s_1$, and two outputs $y_0, y_1$, which come from either the repetition distribution (in which case $y_i = F_{s_i}(x)$) or the independent distribution (in which case $y_i = F_{s_i}(x_i)$, for two independently sampled $x_i$).

**Algorithm 3** $D(s_0, s_1, y_0, y_1)$

$t'_0 \xleftarrow{\$} A_g(s_0, y_0)$
$t'_1 \xleftarrow{\$} A_g(s_1, y_1)$
**if** $t'_0 = t'_1$ **then**
  **return** Repetition
**else**
  **return** Independent
**end if**

Now, we must analyze the probability that $D$ succeeds. If $y_0, y_1$ were generated from the repetition distribution, then since $A_g$ succeeds with probability $\frac{1}{2} + \epsilon$, the probability that $D$ guesses "repetition" is $(\frac{1}{2} + \epsilon)^2 + (\frac{1}{2} - \epsilon)^2 = \frac{1}{2} + 2\epsilon^2$. If $y_0, y_1$ were generated from the independent distribution, because the side information is a balanced boolean function, the probability that the $t_0, t_1$ that would have been generated by $A_m$ are equal is $\frac{1}{2}$. Intuitively, this should mean the probability that $D$ correctly guesses "independent" is just $\frac{1}{2}$. This is in fact the case, because

$$\begin{aligned}
&\Pr[D \text{ correctly guesses independent}] \\
&\quad = \frac{1}{2}\Pr[D \text{ guesses independent} | t_0 = t_1] + \frac{1}{2}\Pr[D \text{ guesses independent} | t_0 \neq t_1] \\
&\quad = \frac{1}{2}\left(2\left(\frac{1}{2}+\epsilon\right)\left(\frac{1}{2}-\epsilon\right)\right) + \frac{1}{2}\left(\left(\frac{1}{2}+\epsilon\right)^2 + \left(\frac{1}{2}-\epsilon\right)^2\right) = \frac{1}{2}.
\end{aligned}$$

Thus the probability that $D$ is correct is $\frac{1}{2} + \epsilon^2$. □

**Corollary 3.** The scheme outlined above is PRIV1 secure.

*Proof.* In [BFOR08], they show that BB-CSS security (Comparison based Semantic Security against Balanced Boolean side information) implies B-CSS security (Comparison based Semantic Security against any Boolean side information), which in turns implies A-CSS which is security against Arbitrary side information. A-CSS security is the terminology in [BFOR08] for PRIV1 security. The only thing to do is to notice that both proofs in [BFOR08] go through unchanged when the adversaries are restricted to be uniform adversaries. □

*Remark.* We note that if the function family $\mathcal{F} = (G, F)$ were assumed to be Decisional Correlated Product (DCP) secure when the inputs were chosen not uniformly, but simply from some high min-entropy distribution, the same proof would go through to show PRIV1 security against any (not necessarily uniform) adversary $A_m$.

*Remark.* On the other hand, there is an example (outlined below) of a PRIV1 secure uniform DE scheme that is not $n$-DCP secure (treating the public key as the seed, key generation as $G$, and encryption as $F$), where $n$ is the size of the message. This does not preclude the construction of a DCP secure family from such a DE scheme, but instead shows that these two notions are not *definitionally* equivalent. To see that a PRIV1 secure DE need not be $n$-DCP secure, take any IND-CPA secure (randomized) encryption scheme, and transform it into a "leaky" scheme that leaks the first bit of randomness used in encryption by simply taking an extra dummy bit of randomness and revealing it in the ciphertext.

The construction of uniform DE from one-way trapdoor permutations given in [BFOR08] makes use of an IND-CPA secure (randomized) encryption scheme.

Without fully reproducing the [BFOR08] construction, we only need to point out that the first bit of randomness is the hard-core predicate defined by the dot product of the message and a vector from the public key. If the "leaky" encryption of the same message under $n$ different public keys is revealed, the message can be reconstructed using linear algebra. This immediately breaks (Decisional) Correlated Product security.

# 9 Conclusion and Open Problems

In this work we suggested a new primitive, the decisional variant of Correlated Product (DCP) secure functions. We argue that this primitive has many appealing properties. To this end, we show a parallel between Correlated Product security and DCP and the Discrete Log Problem and its decisional variant DDH. We also show how to construct simple primitives from DCP such as PRGs and IND-CPA secure encryption.

Our main results examine two main cases: DCP functions with trapdoor and without trapdoor. We show that DCP secure functions (and CP secure functions) without trapdoor are equivalent to one-way functions. This is a somewhat surprising result since notions of correlated product security appear to be much stronger than simple one-wayness. When examining DCP secure functions with trapdoor, we show that they are implied by Lossy Trapdoor Functions, and that DCP secure functions are immediately a Deterministic Encryption scheme.

An interesting line of future research would be to examine further constructions of DCP secure functions with trapdoor.

## References


[BBO07] Mihir Bellare, Alexandra Boldyreva, and Adam O'Neill. Deterministic and efficiently searchable encryption. In *CRYPTO '07*, volume 4622 of *Lecture Notes in Computer Science*, pages 535–552. Springer Berlin / Heidelberg, 2007.

[BFO08] Alexandra Boldyreva, Serge Fehr, and Adam O'Neill. On notions of security for deterministic encryption, and efficient constructions without random oracles. In David Wagner, editor, *CRYPTO '08*, volume 5157 of *Lecture Notes in Computer Science*, pages 335–359. Springer, 2008.

[BFOR08] Mihir Bellare, Marc Fischlin, Adam O'Neill, and Thomas Ristenpart. Deterministic encryption: Definitional equivalences and constructions without random oracles. In David Wagner, editor, *CRYPTO '08*, volume 5157 of *Lecture Notes in Computer Science*, pages 360–378. Springer, 2008.

[FGK+10] David Mandell Freeman, Oded Goldreich, Eike Kiltz, Alon Rosen, and Gil Segev. More constructions of lossy and correlation-secure trapdoor functions. In *Public Key Cryptography 2010 (PKC 2010)*, Lecture Notes in Computer Science, 2010. To appear.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, 1986.

[HILL99] Johan Hastad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[ILL89] Russell Impagliazzo, Leonid A. Levin, and Michael Luby. Pseudo-random generation from one-way functions (extended abstract). In *STOC 89*, pages 12–24, 1989.

[IZ89] Russell Impagliazzo and David Zuckerman. How to recycle random bits. In *FOCS '89*, pages 248–253, 1989.

[Pei09] Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In *STOC '09: Proceedings of the 41st annual ACM symposium on Theory of computing*, pages 333–342, New York, NY, USA, 2009. ACM.

[PW08] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In *STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 187–196, New York, NY, USA, 2008. ACM.

[RS08] Alon Rosen and Gil Segev. Efficient lossy trapdoor functions based on the composite residuosity assumption. http://eprint.iacr.org/2008/134, 2008.

[RS09] Alon Rosen and Gil Segev. Chosen-ciphertext security via correlated products. In *TCC '09: Proceedings of the 6th Theory of Cryptography Conference on Theory of Cryptography*, pages 419–436, Berlin, Heidelberg, 2009. Springer-Verlag.

[Vah10] Yevgeniy Vahlis. Two is a crowd? a black-box separation of one-wayness and security under correlated inputs. In *TCC '10*, volume 5978 of *Lecture Notes in Computer Science*, pages 165–182. Springer Berlin / Heidelberg, 2010.

# Appendix

## A Review of Definitions

### A.1 Discrete Log and Decisional Diffie-Hellman Assumptions

Recall the Discrete Log and DDH assumptions.

Let $\mathcal{G} = \mathcal{G}(\lambda)$ be a family of cyclic groups indexed by the security parameter $\lambda$. With $|\mathcal{G}| = \ell = \ell(\lambda)$.

**Definition 5.** We say that the Discrete Log Problem is hard in $\mathcal{G}$ if for all PPT adversaries $A$, we have

$$\Pr[g \xleftarrow{\$} \mathcal{G}, x \xleftarrow{\$} \mathbb{Z}/\ell\mathbb{Z}, h \leftarrow g^x, y \leftarrow A(g,h); x = y] < \epsilon(\lambda)$$

where $\epsilon(\lambda)$ is a negligible function.

**Definition 6.** We say that the Decisional Diffie Hellman (DDH) problem is hard in $\mathcal{G}$ if the two distributions

$$\{(g, h, g^a, h^a)\} \qquad \{(g, h, g^a, h^b)\}$$

are computationally indistinguishable, where $g, h \xleftarrow{\$} \mathcal{G}$ and $a, b \xleftarrow{\$} \mathbb{Z}/\ell\mathbb{Z}$.

### A.2 Deterministic Encryption

Deterministic Encryption was formally introduced in [BBO07], and many equivalent definitions and constructions were made in [BFO08], [BFOR08]. Although there are many equivalent definitions of security for a deterministic cryptosystem, we present the one that appears in [BBO07], since it will be the easiest for us to work with in our constructions.

**Definition 7.** (Deterministic Encryption/PRIV1)

Let $(G, E, D)$ be a Public Key Cryptosystem (PKC), we say that $(G, E, D)$ is PRIV1 secure if

$$\left|\Pr\left[A^{\mathsf{privreal}} = 1\right] - \Pr\left[A^{\mathsf{privideal}} = 1\right]\right| < \nu$$

for some negligible function $\nu$, and where the games $\mathsf{privreal}$ and $\mathsf{privideal}$ are defined in Figure 4.

| **PRIV1 Real** | **PRIV1 Ideal** |
|---|---|
| $(pk, sk) \xleftarrow{\$} K(1^\lambda)$ | $(pk, sk) \xleftarrow{\$} K(1^\lambda)$ |
| $(x_1, t_1) \xleftarrow{\$} A_m(1^\lambda)$ | $(x_0, t_0) \xleftarrow{\$} A_m(1^\lambda)$ |
| $c \xleftarrow{\$} E(1^\lambda, pk, x_1)$ | $(x_1, t_1) \xleftarrow{\$} A_m(1^\lambda)$ |
| $g \xleftarrow{\$} A_g(1^\lambda, pk, c)$ | $c \xleftarrow{\$} E(1^\lambda, pk, x_0)$ |
| If $g = t_1$ then return 1 else return 0. | $g \xleftarrow{\$} A_g(1^\lambda, pk, c)$ |
| | If $g = t_1$ then return 1 else return 0. |

Figure 4: PRIV1 Security for Deterministic Encryption

More explicitly, in the real game,

- The challenger generates a public/private key pair.
- The message adversary generates a plaintext $x_1$ and some side information $t_1$ about $x_1$.

- Then the challenger encrypts $x_1$ using $pk$.
- The guessing adversary tries to guess the side information $t_1$ from the ciphertext, $c$ and the public key.
- If the adversary correctly guesses the side information $t_1$, he wins the game, otherwise he loses.

In the ideal game,

- The challenger generates a public/private key pair.
- The message adversary generates two plaintext side-information pairs $(x_0, t_0)$, $(x_1, t_1)$.
- Then the challenger encrypts $x_0$ using $pk$.
- The guessing adversary tries to guess the side information $t_1$ from the ciphertext, $c$ (which is independent of $x_1$) and the public key.
- If the adversary correctly guesses the side information $t_1$, he wins the game, otherwise he loses.

In [BFO08], they give constructions of PRIV1 secure deterministic encryption from Lossy Trapdoor Functions. We consider the weakening of $A_m$ to require that it returns a uniform distribution on $x_i$ it outputs. This is the same modification used in [BFOR08] to be able to construct DE from OWTDP. A DE scheme that is secure against this type of adversary is known as *secure on uniform messages*. In [BFOR08], they give constructions of PRIV1 secure deterministic encryption on uniform messages from one-way trapdoor permutations.

### A.3 Lossy Trapdoor Functions

Lossy Trapdoor Functions were first defined in [PW08], and we review the definition here.

A tuple $(S_{\text{ltdf}}, F_{\text{ltdf}}, F^{-1}_{\text{ltdf}})$ of PPT algorithms is called a family of $(d, k)$-Lossy Trapdoor Functions if the following properties hold:

- **Sampling Injective Functions:** $S_{\text{ltdf}}(1^\lambda, 1)$ outputs $s, t$ where $s$ is a function index, and $t$ its trapdoor. We require that $F_{\text{ltdf}}(s, \cdot)$ is an injective deterministic function on $\{0,1\}^d$, and we have that $F^{-1}_{\text{ltdf}}(t, F_{\text{ltdf}}(s, x)) = x$ for all $x$.
- **Sampling Lossy Functions:** $S_{\text{ltdf}}(1^\lambda, 0)$ outputs $(s, \perp)$ where $s$ is a function index and $F_{\text{ltdf}}(s, \cdot)$ is a function on $\{0,1\}^d$, where the image of $F_{\text{ltdf}}(s, \cdot)$ has size at most $2^{d-k}$.
- **Indistinguishability:** The first outputs of $S_{\text{ltdf}}(1^\lambda, 0)$ and $S_{\text{ltdf}}(1^\lambda, 1)$ are computationally indistinguishable.

### A.4 Pseudorandom Functions

A Pseudorandom Function (PRF) [GGM86] is a deterministic function $\mathsf{PRF} : \{0,1\}^\lambda \times \{0,1\}^a \to \{0,1\}^b$, such that if $\mathcal{RO} : \{0,1\}^a \to \{0,1\}^b$ is a truly random function, then for all PPT adversaries $A$,

$$\left| \Pr[A^{\mathcal{RO}(\cdot)} = 1] - \Pr[A^{\mathsf{PRF}(s,\cdot)} = 1] \right| < \nu$$

for some negligible function $\nu = \nu(\lambda)$. Here, the probability is taken over the coins of $A$, the choice of $s \xleftarrow{\$} \{0,1\}^\lambda$, and the choice of $\mathcal{RO}$. It is known how to construct a Pseudorandom Function from any one-way function (see [GGM86, ILL89, HILL99]).