Electronic Colloquium on Computational Complexity, Revision 2 of Report No. 127 (2009)

Extended-DDH and Lossy Trapdoor Functions*

Brett Hemenway † Rafail Ostrovsky‡

March 22, 2012

Abstract

Lossy Trapdoor Functions (LTFs) were introduced by Peikert and Waters in STOC '08 and since then have found many applications and have proven to be an extremely useful and versatile cryptographic primitive. Lossy trapdoor functions were used to build the first injective trapdoor functions based on DDH, the first IND-CCA cryptosystems based on lattice assumptions, and they are known to imply deterministic encryption, collision resistant hash-functions, oblivious transfer and a host of other important primitives. While LTFs can be instantiated under most known cryptographic hardness assumptions, no constructions until today existed based on generic cryptographic primitives. In this work, we show that any Homomorphic Smooth Hash Proof System, introduced by Cramer and Shoup in EUROCRYPT '02, can be used to construct LTFs. In addition to providing a connection between two important cryptographic primitives – our construction implies the first construction of LTFs based on the QR assumption.

Smooth Hash Proof Systems (SHPs) can be seen as a generalization of the DDH assumption, yet can be built on other cryptographic assumptions, such as the DCR or QR assumptions. Yet, until today, a "translation" of results proven secure under DDH to results under DCR or QR has always been fraught with difficulties. Thus, as our second goal of this paper, we ask the following question: is it possible to streamline such translations from DDH to QR and other primitives? Our second result formally provides this connection. More specifically, we define an Extended Decisional Diffie Hellman (EDDH) assumption, which is a simple and natural generalization of DDH. We show that EDDH can be instantiated under both the DCR and QR assumptions. This gives a much simpler connection between the DDH and the DCR and QR assumptions and provides an easy way to translate proofs from DDH to DCR or QR. That is, the advantage of the EDDH assumption is that most schemes (including LTFs) proven secure under the DDH assumption can easily be instantiated under the DCR and QR assumptions with almost no change to their proofs of security.

*A preliminary version of this work appeared in PKC 2012

†email:bhemen@umich.edu

‡email:rafail@cs.ucla.edu

ISSN 1433-8092

1 Introduction

The first practical IND-CCA secure cryptosystem was built by Cramer and Shoup under the Decisional Diffie-Hellman (DDH) assumption [CS98]. In a follow up work, Cramer and Shoup introduced projective hash proofs as a means of generalizing their original DDH-based construction [CS02]. This generalization allowed them to create unified constructions of IND-CCA secure cryptosystems based on Paillier's Decisional Composite Residuosity (DCR) assumption and the Quadratic Residuosity (QR) assumption.

Since their introduction, projective hash proof systems have proven to be an effective tool for generalizing constructions that were originally proven secure under the DDH assumption. Indeed, many important results use the framework of projective hash proofs to take a system built using the DDH assumption and instantiate it using the DCR or QR assumptions.

Cramer and Shoup [CS02] converted the DDH-based construction of IND-CCA encryption [CS98] to one based on the DCR or QR assumptions. Kalai and Halevi [Kal05, HK07] converted the DDH-based construction of OT given by Naor and Pinkas [NP01] to one based on the DCR or QR assumptions. Brakerski and Goldwasser [BG10] converted the DDH-based construction of circular secure encryption given by Boneh, Halevi, Hamburg and Ostrovsky [BHHO08] to one based on the DCR or QR assumptions[1].

This series of works generalizing DDH-based constructions suggests the heuristic that "anything that can be done with DDH can be done with DCR or QR." Like any heuristic it is not completely accurate, but it appears to provide the right intuition.

While projective hash proof systems suggest a means for converting a DDH-based scheme to a DCR or QR based scheme, the generality of projective hash proof systems framework often means that converting the actual proofs of security can be fairly technical. This is evidenced in the works of [CS02, Kal05, HK07, BG10] which provided significant technical contributions beyond the original constructions of [CS98, NP01, BHHO08].

This work makes two contributions: First, we show that Lossy Trapdoor Functions (LTFs) of Peikert and Waters [PW08] can be built under general assumptions, namely any homomorphic smooth hash proof system. This provides a connection between two important cryptographic primitives. Second, we introduce the *Extended Decisional Diffie-Hellman (EDDH)* assumption, and show how it can be instantiated using the DCR and QR assumptions. This second result provides a justification for the heuristic noted above that the DCR and QR assumptions "imply" the DDH assumption. While the EDDH assumption does not appear to be as general as the notion of projective hash proof systems, its simplicity gives it some advantages. In particular, the EDDH assumption provides a much simpler method for identifying

[1] Brakerski and Goldwasser did not explicitly use the language of projective hash proofs, but their construction fits the framework exactly.

which DDH-based constructions can be instantiated under the DCR or QR assumptions, and proofs of security under the EDDH assumption are almost identical to those under the DDH assumption. Using the framework of EDDH, it becomes almost immediate that the DDH constructions of [NP01, BHHO08, PW08] can be instantiated under the DCR or QR assumptions with almost no modifications to the proofs of security.

As mentioned above, our first result is a construction of lossy trapdoor functions (LTFs) from general assumptions. Lossy trapdoor functions were introduced by Peikert and Waters [PW08]. LTFs provided the first injective trapdoor functions based on the Decisional Diffie-Hellman (DDH) assumption, and the first chosen ciphertext (IND-CCA) secure cryptosystem based on lattice assumptions. In addition to providing natural constructions of injective trapdoor functions and IND-CCA secure cryptosystems, Peikert and Waters went on to show that LTFs provide very natural constructions of many cryptographic primitives, including pseudo-random generators, collision-resistant hash functions, and oblivious transfer. The extremely intuitive nature of these many constructions provided early evidence of the value of LTFs as a cryptographic primitive. Since the original work of Peikert and Waters, lossy trapdoor functions have been shown to imply many other important cryptographic primitives. In [BFO08], Boldyreva, Fehr and O'Neill showed that LTFs imply deterministic encryption. Deterministic encryption was introduced in [BBO07], and captures the strongest notion of security possible for a deterministic function. In contrast to one-way functions, which do leak the parity of a random subset of the bits of its input [GL89], deterministic encryption does not leak *any fixed function*[2] of its input. Deterministic encryption has applications to efficiently searchable encryption, and securing legacy systems. Lossy trapdoor functions were then shown to imply correlated product secure functions by Rosen and Segev in [RS09]. Roughly a family of correlated product secure functions is a family of functions that remain one-way even when the output of multiple functions is given *on the same input*. In [MY09], Mol and Yilek introduced a relaxation of lossy trapdoor functions called *slightly lossy trapdoor functions*, and showed that even slightly lossy trapdoor functions are sufficient to achieve correlated product secure functions. Lossy functions, (without the need for a trapdoor) have been shown to imply leaky pseudo-entropy functions [BHK11].

Lossy trapdoor functions have been constructed from a variety of concrete hardness assumptions. In [PW08], Peikert and Waters constructed LTFs from the DDH assumption and lattice assumptions, and an efficient construction of LTFs from Paillier's Decisional Composite Residuosity (DCR) assumption was given independently in [BFO08] and [RS08]. In concurrent, independent work, Freeman et al. [FGK$^+$10] give constructions of LTFs from the D-Linear Assumption and constructions of slightly lossy trapdoor functions from the QR assumption.

While we have seen a wide variety of important consequences of lossy trapdoor

[2]independent of the choice of the key for the deterministic encryption.

functions, there remains a lack of general constructions. This work provides the first constructions of LTFs from generic primitives (in this case homomorphic smooth hash proof systems, and diverse group systems) as well as the first construction of fully lossy trapdoor functions from the well-known Quadratic Residuosity (QR) assumption.

This result has a number of other consequences. Applying our construction to the results of [BFO08], we achieve the first construction of deterministic encryption from smooth homomorphic hash proof systems. Applying our results to those of [RS09], we give the only known construction of correlated product secure functions from a generic primitive other than lossy trapdoor functions,[3] and the first known construction of correlated product secure functions from the QR assumption.[4] Applying the separation of Rosen and Segev, we provide a black-box separation of smooth homomorphic hash proof systems and one-way trapdoor permutations.

The second contribution of this work is a development of the connection between the DDH, DCR and QR assumptions. Projective hash proof systems [CS02] showed that many properties of DDH-based protocols could be achieved using the DCR or QR assumptions. In this work, we introduce the Extended DDH (EDDH) assumption, and show how the EDDH assumption is implied by the DDH, DCR and QR assumptions. One formulation of the DDH assumption is that the distributions $\{g, g^a, g^b, g^{ab}\}$, $\{g, g^a, g^b, g^c\}$ are computationally indistinguishable. Equivalently, $\{g, g^a, g^b, g^{ab}\} \approx_c \{g, g^a, g^b, g^{ab}r\}$ for some uniformly chosen element r in the group. The EDDH assumption is the same, except that r is chosen from a subgroup instead of the entire group. Thus the EDDH assumption states that $\{g, g^a, g^b, g^{ab}\}$ and $\{g, g^a, g^b, g^{ab}r\}$ are computationally indistinguishable when r is chosen uniformly from a given subgroup of the universe group. See Definition 6 for the formal definition. The value of the EDDH assumption is that it provides a very simple method for converting constructions based on the DDH assumption into constructions which can be proven secure under the DCR or QR assumptions. Since the semantics of the EDDH assumption are very similar to those of the DDH assumption in many cases proofs of security under the DDH assumption go through almost unchanged under the EDDH assumption.

1.1 Previous Work

Lossy Trapdoor Functions (LTFs) were introduced by Peikert and Waters in [PW08], simultaneously providing the first construction of one-way trapdoor functions from the Decisional Diffie Hellman and the first IND-CCA secure cryptosystem based on

[3] There are two concrete constructions of correlated product secure functions that are not lossy trapdoor functions. A construction based on the Learning With Error (LWE) problem given by Peikert in [Pei09], and a construction based on the hardness of syndrome decoding given by Freeman et al. in [FGK+10].

[4] A completely different construction of correlated product secure functions from the QR assumption is given in the concurrent, independent work of Freeman et al. [FGK+10].

lattice assumptions.

Roughly, a family of lossy trapdoor functions is a family of functions with two computationally indistinguishable branches. An injective branch with a trapdoor, and a lossy branch which statistically loses information about its input, in particular the image size of the lossy branch is required to be much smaller than its domain size. If the lossy branch is lossy enough, this immediately implies that the injective branch is an injective one-way trapdoor function. Peikert and Waters gave constructions of lossy trapdoor functions from the DDH assumption and lattice-based assumptions. In [BFO08], [RS08], Boldyreva et al. and Rosen and Segev gave efficient constructions of lossy trapdoor functions from Paillier's DCR assumption. A construction of lossy trapdoor functions from the D-Linear assumption, and slightly lossy trapdoor functions from the QR assumption are given in the concurrent, independent work of [FGK$^+$10].

Lossy trapdoor functions are known to imply IND-CCA secure encryption. In addition to IND-CCA secure encryption, LTFs were shown to imply collision-resistant hash functions [PW08], deterministic encryption [BFO08], lossy encryption [PVW08] and correlated product secure functions [RS09].

Projective Hash Proof Systems were introduced by Cramer and Shoup in [CS02], generalizing their construction of IND-CCA encryption from the Decisional Diffie-Hellman (DDH) assumption given in [CS98]. In [CS02], Cramer and Shoup defined two types of hash proof systems, smooth projective hash families, which immediately implied IND-CPA secure encryption, and universal hash families, which could be used as a type of designated verifier proof system for the specific class of language given by smooth projective hash families. They went on to show that universal hash proof systems imply smooth projective hash proof systems, so it was sufficient to construct only universal hash proof systems. Their general construction, however, was fairly inefficient, and in all of their constructions they were able to avoid the general construction of smooth projective hash proof systems, and create efficient smooth projective hash proof systems directly. In this work, we will deal only with smooth projective hash proof systems.

In order to construct explicit hash proof systems, Cramer and Shoup defined another primitive called a *Diverse Group System*. Diverse Group Systems seemed to capture the essential part of the algebraic structure of a cyclic group, and they gave a very natural construction of projective hash proof systems from Diverse Group Systems. They went on to construct diverse group systems from the DDH assumption, the Quadratic Residuosity (QR) assumption and the Decisional Composite Residuosity (DCR) assumption.

The first result of this work is a proof that smooth homomorphic hash proof systems imply lossy trapdoor functions. By providing a link between smooth homomorphic hash proof systems, and lossy trapdoor functions, we provide a number of new connections as well. This work provides the first construction of lossy trapdoor functions from a generic primitive. Additionally, it provides the first construction of

deterministic encryption from smooth homomorphic projective hash proof systems.

Our first result uses the framework of smooth projective hashing to generalize the DDH-based construction of LTFs from [PW08]. Smooth projective hash proof systems have been used to generalize DDH-based constructions in the past. Kalai and Halevi [Kal05, HK07] used them to generalize Naor and Pinkas's OT protocol [NP01], and Brakerski and Goldwasser [BG10] generalized the circular secure encryption of Boneh, Halevi, Hamburg and Ostrovsky [BHHO08] using the same framework. This series of results indicates a close relationship between the DDH, DCR and QR assumptions.

The second result of this work is a development of the connection between the DDH, DCR and QR assumptions. One of the most useful features of projective hash proof systems is that they provide a framework for converting cryptographic schemes designed under the DDH assumption into cryptographic schemes that are provably secure under the DCR or QR assumptions. While projective hash proof systems showed a close connection between the DDH, DCR and QR assumptions, generality of projective hash proof systems makes this connection difficult to see. To make the connection between these three hardness assumptions clearer, we introduce the EDDH assumption and show how it can be realized under the DCR and QR assumptions. The benefit of the EDDH assumption is that it is semantically very similar to the DDH assumption, so many existing constructions whose security rests on the DDH assumption (including the construction of LTFs by Peikert and Waters) can immediately be instantiated under the DCR or QR assumptions. In particular, we note that the proof of [PW08] can be instantiated using the EDDH assumption. This gives a novel construction of LTFs from the DCR assumption and the first construction of LTFs from the QR assumption.

1.2 Our Contributions

In this work, we show that smooth homomorphic hash proof systems imply lossy trapdoor functions (LTFs). It was shown in [BFO08] that lossy trapdoor functions imply deterministic encryption, so our results give the first construction of deterministic encryption from smooth homomorphic hash proof systems.

In [RS09], Rosen and Segev introduced correlated product secure functions, and showed that lossy trapdoor functions are correlated product secure. Applying their results to our construction, we have a construction of correlated product secure functions from smooth homomorphic hash proof systems. Finally, combining our results with the black-box separations of Rosen and Segev [RS09], we find that there is a black-box separation between one-way trapdoor permutations and smooth homomorphic hash proof systems.

Our primary results are summarized as follows:

Theorem. Smooth Homomorphic Projective Hash Proof Systems imply Lossy Trapdoor Functions.

This theorem has a number of immediate Corollaries. Since Boldyreva et al. [BFO08] showed that LTFs imply deterministic encryption (as defined in [BBO07]), we have

Corollary. Smooth Homomorphic Projective Hash Proof Systems imply deterministic encryption.

Since Rosen and Segev [RS09] showed that LTFs imply correlated product secure encryption, and a black-box separation between one-way trapdoor permutations and lossy trapdoor functions, we have

Corollary. Smooth Homomorphic Projective Hash Proof Systems imply correlated product secure functions.

Corollary. There is a black-box separation between Smooth Homomorphic Projective Hash Proof Systems and one-way trapdoor permutations, i.e. there exists an oracle, relative to which the latter exists but the former does not.

In addition to the new constructions outlined above, in Section 4 we introduce the Extended Decisional Diffie Hellman (EDDH) assumption, which provides a simple way to achieve a DDH-like property under the DCR and QR assumptions. This serves to unify many of the previous constructions (e.g. [NP01] and [Kal05, HK07], [BHHO08] and [BG10]), and provides a more familiar alternative to projective hash proof systems.

Applying these results yields lossy trapdoor functions from the DDH, DCR and QR assumptions. When applied to DDH, the construction achieved in this way is identical to the construction of LTFs given by Peikert and Waters in [PW08], however the constructions from the DCR and QR assumptions are new. While our construction of LTFs from the DCR assumption is less efficient than that given by [BFO08] and [RS08], our results provide the first construction of lossy trapdoor functions from the QR assumption.

2 Preliminaries

2.1 Notation

If A is a Probabilistic Polynomial Time (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. If R is a set, we use $r \xleftarrow{\$} R$ to denote sampling uniformly from R.

We use the notation

$$\Pr[r \xleftarrow{\$} R; x \xleftarrow{\$} X : A(x, r) = c],$$

to denote the probability that A outputs c when x is sampled uniformly from X and r is sampled uniformly from R. We define the statistical distance between two distributions X, Y to be

$$\Delta(X,Y) = \frac{1}{2}\sum_x |\Pr[X=x] - \Pr[Y=x]|$$

If X and Y are families of distributions indexed by a security parameter λ, 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 λ, we have $\Delta(X,Y) < \frac{1}{p(\lambda)}$. We use $X \approx_c Y$ to mean X and Y are computationally close, *i.e.*, for all PPT adversaries A, for all polynomials p, then for all sufficiently large λ, we have $|\Pr[A^X = 1] - \Pr[A^Y = 1]| < 1/p(\lambda)$.

2.2 Lossy Trapdoor Functions

We briefly recall the definition of lossy trapdoor functions given in [PW08].

A tuple $(S_{\text{ltdf}}, F_{\text{ltdf}}, F^{-1}_{\text{ltdf}})$ of PPT algorithms is called a family of (n,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 $\{0,1\}^n$, and $F^{-1}_{\text{ltdf}}(t, F_{\text{ltdf}}(s,x)) = x$ 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 $\{0,1\}^n$, where the image of $F_{\text{ltdf}}(s,\cdot)$ has size at most 2^{n-k}.
- **Indistinguishability:** The first outputs of $S_{\text{ltdf}}(1^\lambda, 0)$ and $S_{\text{ltdf}}(1^\lambda, 1)$ are computationally indistinguishable.

2.3 Subset Membership Problems

In this section we recall the definition of of a subset membership problem as formalized in [CS02]. Roughly, given sets $L \subset X$, we want L and X to be computationally indistinguishable.

Formally, given a family of sets (X, L, W) indexed by a security parameter λ, we require $L \subset X$, and there is a binary relation $\mathcal{R} : X \times W \to \{0,1\}$. If $\mathcal{R}(x,w) = 1$, we say that w is a witness for x. In this work, we will restrict our attention to relations $\mathcal{R}$ such that for all $x \in L$, there exists a $w \in W$ such that $\mathcal{R}(x,w) = 1$, and for all $x \notin L$, and all $w \in W$, $\mathcal{R}(x,w) = 0$.

We also need the following efficient sampling algorithms.

- **Instance Sampling:** Given a security parameter λ, we can sample (X, L, W) and $\mathcal{R}$.

- **Sampling Without Witness:** Given (X, L, W) we can sample (statistically-close to) uniformly on X.
- **Sampling With Witness:** Given (X, L, W) we can sample x (statistically-close to) uniformly on L, along with a witness w such that $\mathcal{R}(x, w) = 1$.

Definition 1. A subset membership problem is called *hard* if for all PPT distinguishers,

$$|\Pr[x \xleftarrow{\$} X : D(x) = 1] - \Pr[x \xleftarrow{\$} L : D(x) = 1]| < \nu(\lambda),$$

for some negligible function ν.

As in [CS02], the security of all of our constructions will rely on the security of some underlying hard subset membership problem. In fact, the hardness assumptions DDH, DCR and QR all have natural formulations in terms of hard subset membership problems [CS02].

We briefly review them here. Full descriptions can be found in [CS02].

- **DDH:**
  If G is a cyclic group (written multiplicatively), then the DDH assumption is equivalent to stating that (X, L, W) is a hard subset membership problem where $X = G \times G$, and $L = \{(g^w, h^w)\}$, where g, h generate G and are specified by the instance description.
- **DCR:**
  If $N = pq$ is a product of two safe primes, then the DCR assumption is equivalent to stating that (X, L, W) is a hard subset membership problem where $X = (\mathbb{Z}/N^2\mathbb{Z})^*$, and $L = \{(g^N)^w \mod N^2\}$, where $g \in X$ is specified by the instance description.
- **QR:**
  If $N = pq$ is a product of two safe primes, then the QR assumption is equivalent to stating that (X, L, W) is a hard subset membership problem where $X = \left\{x \in \mathbb{Z}_N^* : \left(\frac{x}{N}\right) = 1\right\}$, and $L = \{(g^2)^w \mod N\}$ where $g \in X$ is specified by the instance description.

2.4 Smooth Hash Proof Systems

We briefly recall the notion of *smooth projective hash* families as defined by Cramer and Shoup in [CS02]. Let H be a function family indexed by keys in the a keyspace K, *i.e.* for each $k \in K$, $H_k : X \to \Pi$. Let $L \subset X$ and a "projection" $\alpha : K \to S$. We require efficient evaluation algorithms such that, for any $x \in X$, $H_k(x)$ is efficiently computable using $k \in K$. Using the terminology of [CS02], this is called the *private evaluation algorithm*. Finally we require efficient sampling algorithms to sample uniformly from X, uniformly from K, and uniformly from L *along with a witness*.

The security properties of the system will follow from the indistinguishability of X and L.

Definition 2. The set $\mathsf{HPS} = (H, K, X, L, \Pi, S, \alpha)$ is a *projective hash family* if, for all $k \in K$, the action of H_k on the subset L is completely determined by $\alpha(k)$.

For a projective hash family, $\alpha(k)$ determines the output of H_k on L. Additionally, if $x \in L$ and a witness w for $x \in L$ is known, then we require that $H_k(x)$ is efficiently computable given $x, w, \alpha(k)$. This is called the *public evaluation algorithm*. A *smooth* projective hash family is one in which α does not encode any information about the action of H_k on $X \setminus L$.

Definition 3. Let $(H, K, X, L, \Pi, S, \alpha)$ be a projective hash family, and define two distributions Z_1, Z_2 taking values on the set $X \setminus L \times S \times \Pi$. For Z_1, we sample $k \xleftarrow{\$} K$, $x \xleftarrow{\$} X \setminus L$, and set $s = \alpha(k)$, $\pi = H_k(x)$, for Z_2 we sample $k \xleftarrow{\$} K$, $x \xleftarrow{\$} X \setminus L$, and $\pi \xleftarrow{\$} \Pi$, and set $s = \alpha(k)$. The projective hash family is called *ν-smooth* if $\Delta(Z_1, Z_2) < \nu$.

This means that, given $\alpha(k)$ and $x \in X \setminus L$, $H_k(x)$ is statistically close to uniform on Π.

In [CS02], they showed that smooth projective hash families immediately imply IND-CPA secure encryption by taking $sk = k$, $pk = \alpha(k)$, and to encrypt a message $m \in \Pi$, we sample $x \in L$ along with randomness and output $E(m) = (x, H_k(x) + m)$.

We extend the definition of smooth projective hash proof systems slightly

Definition 4. If $\mathsf{HPS} = (H, K, X, L, \Pi, S, \alpha)$ is a projective hash family, we say that HPS is a *homomorphic projective hash family* if X is a group, and for all $k \in K$, and $x_1, x_2 \in X$, we have $H_k(x_1) + H_k(x_2) = H_k(x_1 + x_2)$, that is to say H_k is a homomorphism for each k.

In [CS02] Cramer and Shoup provide smooth homomorphic projective hash families based on the DDH, DCR and QR assumptions.

3 Lossy Trapdoor Functions from Smooth Homomorphic Hash Proof Systems

Peikert and Waters [PW08] gave a construction of lossy trapdoor functions from the Decisional Diffie-Hellman (DDH) assumption. In this section, we show that a similar construction goes through with smooth homomorphic hash proof systems. This extends the intuition given in [CS02] that projective hashing provides a good generalization of the DDH assumption. We note, however, that although our construction is very similar that of [PW08], the proofs of security are quite different.

Let (X, L, W) be a hard subset membership problem. For notational convenience, we suppress the dependence on the security parameter λ. Let $\mathbf{H} =$

$(H, K, X, L, \Pi, S, \alpha)$ be an associated smooth homomorphic projective hash family.

- **Key Generation:**
  Pick $x_1, \ldots, x_n \in L$.
  Fix $b \in \Pi \setminus \{0\}$.
  Generate the matrix $B = (B_{ij}) \subset \Pi^{n \times n}$, where $B_{ij} = 0$ if $i \neq j$, and
  In lossy mode $B_{ii} = 0$ for all i.
  In injective mode $B_{ii} = b$.

  Sample $k_1, \ldots, k_n \leftarrow K$, and output

$$R = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \qquad A = \begin{pmatrix} H_{k_1}(x_1) + B_{11} & \cdots & H_{k_1}(x_n) + B_{1n} \\ \vdots & \ddots & \vdots \\ H_{k_n}(x_1) + B_{n1} & \cdots & H_{k_n}(x_n) + B_{nn} \end{pmatrix}$$

  The trapdoor will be $(k_1, \ldots, k_n)$.

- **Evaluation:**
  Given a message $z = z_1, \ldots, z_n \in \{0,1\}^n$
  Given a function index R, A, calculate

$$F_{R,A}(z) = (Rz, Az) = \left(\sum_{i=1}^{n} z_i x_i, \begin{pmatrix} \sum_{i=1}^{n} z_i (H_{k_1}(x_i) + B_{1i}) \\ \vdots \\ \sum_{i=1}^{n} z_i (H_{k_n}(x_i) + B_{ni}) \end{pmatrix} \right).$$

- **Trapdoor:**
  Given a value (Rz, Az), and a trapdoor $(k_1, \ldots, k_n)$, we begin by noting that the homomorphic property of H_k guarantees that

$$\begin{aligned} F_{R,A}(z) = (Rz, Az) &= \left(\sum_{i=1}^{n} z_i x_i, \begin{pmatrix} \sum_{i=1}^{n} z_i (H_{k_1}(x_i) + B_{1i}) \\ \vdots \\ \sum_{i=1}^{n} z_i (H_{k_n}(x_i) + B_{ni}) \end{pmatrix} \right) \\ &= \left(\sum_{i=1}^{n} z_i x_i, \begin{pmatrix} H_{k_1}\left(\sum_{i=1}^{n} z_i x_i\right) + \sum_{i=1}^{n} z_i B_{1i}) \\ \vdots \\ H_{k_n}\left(\sum_{i=1}^{n} z_i x_i\right) + \sum_{i=1}^{n} z_i B_{ni}) \end{pmatrix} \right) \end{aligned}$$

  Since $\sum_{i=1}^{n} z_i x_i$, and k_i is known, we can calculate $H_{k_i}\left(\sum_{i=1}^{n} z_i x_i\right)$ and subtract it from each component to recover the vector

$$\left(\sum_{i=1}^{n} z_i B_{1i}, \cdots, \sum_{i=1}^{n} z_i B_{ni} \right)^t.$$

Now, in injective mode, $B_{ij} = 0 \in \Pi$ for $i \neq j$, and $B_{ij} = b$ for $i = j$, so

$$\left(\sum_{i=1}^{n} z_i B_{1i}, \cdots, \sum_{i=1}^{n} z_i B_{ni} \right)^t = (z_1 b, \cdots, z_n b) .$$

Since the $z_i \in \{0, 1\}$, and since b is known, we can recover the z_i by inspection.

Remark: Notice that we do not make use of the projection α in our construction, it will appear, however, in the proof of security. Unlike in [CS02], we do not require that α be efficiently computable, merely that it exists.

We now examine the security of this construction.

Lemma 1. In Lossy Mode, the image of F has size at most $|X|$.

Proof. Notice that in Lossy Mode, since $B_{ij} = 0$ for all i, j,

$$F_{R,A}(z) = \left(\sum_{i=1}^{n} z_i x_i, \begin{pmatrix} H_{k_1}\left(\sum_{i=1}^{n} z_i x_i\right) \\ \vdots \\ H_{k_n}\left(\sum_{i=1}^{n} z_i x_i\right) \end{pmatrix} \right)$$

which depends only on the sum $\sum_{i=1}^{n} z_i x_i \in X$. Thus the size of the image is bounded by $|X|$. □

Thus by taking $n > \log(|X|)$, we can make the lossy mode of F as lossy as desired.

Lemma 2. The Injective and Lossy Modes are computationally indistinguishable.

Proof. To prove the indistinguishability, we proceed via hybrid argument, on the columns of A. Let D_i be the distribution on function indices, where the first i columns of B are the diagonal matrix with b_j along the diagonal, and the last $n - i$ columns of B are the zero matrix. Thus D_n is the distribution output by the Injective Sampling Algorithm, and D_0 is the distribution output by the Lossy Sampling algorithm. To show that D_0 and D_n are indistinguishable, it suffices to show that D_{i-1} and D_i are indistinguishable for each $i \in \{1, \ldots, n\}$.

We define two new distributions, distribution D'_i, D'_{i-1}, which are identical to D_i and D_{i-1} respectively except that (in both cases) instead of setting $x_i \leftarrow L$, we sample $x_i \leftarrow X \setminus L$, and all the other x_i sampled from L as before. Clearly D'_i and D_i are computationally indistinguishable assuming it is hard to distinguish L from $X \setminus L$. Thus to show that D_{i-1} and D_i are computationally indistinguishable, it suffices to show that D'_{i-1} and D'_i are computationally indistinguishable. In fact, we will show that D'_{i-1} and D'_i are *statistically* indistinguishable. The only difference in the distributions D'_{i-1} and D'_i are in the ith columns, so it is enough to consider the distributions of the ith columns conditioned on all the rest of the values. In particular, we condition on the values of $x_1, \ldots, x_n$, and the values $H_{k_\ell}(x_j)$ for

$j \neq i$, and $\ell \in [n]$. The smoothness of the hash proof system guarantees that for $x_i \notin L$, $H_k(x_i)$ will be uniform conditioned on x_i and $\alpha(k)$. In our construction, we must condition on $x_i, H_k(x_j)$ for $j \neq i$. Since $x_j \in L$ for $j \neq i$, the value of $H_{k_\ell}(x_j)$ is completely determined by $\alpha(k_\ell)$. In particular, there will still be entropy in $H_k(x_i)$ even conditioned on all the other values. We make this explicit below. We begin by noticing that if $x_i \leftarrow X \setminus L$, and $k \leftarrow K$, then the distributions

$$\Lambda_1 = \{x_i, \alpha(k), H_k(x_i) + b_i\} \qquad \Lambda_2 = \{x_i, \alpha(k), H_k(x_i)\},$$

are statistically close, i.e. $\Delta(\Lambda_1, \Lambda_2) = \nu$. In particular, this implies that the distributions

$$\{x_i, H_{k_i}(x_1), \ldots, \widehat{H_{k_i}(x_i)}, \ldots, H_{k_i}(x_n), H_{k_i}(x_i)+b_i\} \qquad \{x_i, H_{k_i}(x_1), \ldots, \widehat{H_{k_i}(x_i)}, \ldots, H_{k_i}(x_n), H_{k_i}(x_i)\}$$

are statistically close. Since each of the k_j are independent, this shows that $\Delta(D'_{i-1}, D'_i) < \nu$. Putting it all together, we have

$$D_{i-1} \approx_c D'_{i-1} \approx_s D'_i \approx_c D_i.$$

□

We remark that this construction *does not* make use of the projection α. The projective property *is* used, however, since we condition on $H_k(x)$ for $x \in L$, which leaves at least as much entropy in k as conditioning on $\alpha(k)$, since $\alpha(k)$ determines $H_k(x)$.

A similar construction and proof goes through for Diverse Group Systems (see Appendix D for a discussion). Thus we arrive at

Theorem 1. Smooth Homomorphic Projective Hash Proof Systems imply Lossy Trapdoor Functions, and Diverse Group Systems imply Lossy Trapdoor Functions.

This theorem has a number of immediate Corollaries. Since Boldyreva et al. [BFO08] showed that LTFs imply deterministic encryption (as defined in [BBO07]), we have Corollary 1. Since Rosen and Segev [RS09] showed that LTFs imply correlated product secure encryption, we have Corollary 2. Since Rosen and Segev showed a black-box separation between one-way trapdoor permutations and lossy trapdoor functions, we have Corollary 3.

Corollary 1. Smooth Homomorphic Projective Hash Proof Systems imply deterministic encryption.

Corollary 2. Smooth Homomorphic Projective Hash Proof Systems imply correlated product secure functions.

Corollary 3. There is a black-box separation between Smooth Homomorphic Projective Hash Proof Systems and one-way trapdoor permutations, i.e. there exists an oracle, relative to which the latter exists but the former does not.

4 The Extended DDH Assumption

In this section, we introduce the Extended Decisional Diffie Hellman (EDDH) assumption. Let $\mathbb{G}$ be commutative group (written multiplicatively). The DDH assumption states that

Definition 5 (The DDH Assumption)**.** Assume $\mathbb{G}$ is a group with an efficient sampling algorithm, and $K = \{1, \ldots, |\mathbb{G}|\}$. Then the DDH assumption states that

$$\{(g, g^a, g^b, g^{ab}) : g \xleftarrow{\$} G, a, b \xleftarrow{\$} K\} \approx_c \{(g, g^a, g^b, g^c) : g \xleftarrow{\$} G, a, b, c \xleftarrow{\$} K, \}$$

When $\mathbb{G}$ is a cyclic group, this can be rephrased as

$$\{(g, g^a, g^b, g^{ab}) : g \xleftarrow{\$} G, a, b \xleftarrow{\$} K\} \approx_c \{(g, g^a, g^b, g^{ab}h) : g \xleftarrow{\$} G, a, b \xleftarrow{\$} K, h \xleftarrow{\$} \mathbb{G}\}$$

We introduce a slight modification of the DDH assumption, called the *Extended Decisional Diffie Hellman (EDDH)* assumption.

Definition 6 (The EDDH Assumption)**.** For a group $\mathbb{G}$, and a (samplable) subgroup $\mathbb{H} \triangleleft \mathbb{G}$, the *extended decisional diffie hellman (EDDH)* problem is said to be hard if there exists a samplable set $G \subset \mathbb{G}$ and samplable sets $K \subset \mathbb{Z}$ such that the following two distributions are computationally indistinguishable:

$$\{(g, g^a, g^b, g^{ab}) : g \xleftarrow{\$} G, a, b \xleftarrow{\$} K\} \approx_c \{(g, g^a, g^b, g^{ab}h) : g \xleftarrow{\$} G, a, b \xleftarrow{\$} K, h \xleftarrow{\$} \mathbb{H}\}$$

It is not hard to see:

Lemma 3. If $K = \{1, \ldots, |\mathbb{G}|\}$, and $\mathbb{H} = \mathbb{G}$, then the EDDH assumption is just the DDH assumption in the group $\mathbb{G}$.

The utility of this assumption is that it extracts the essential properties of the DDH assumption, yet it can be instantiated under the QR assumption and the DCR assumption. See Appendix E.1 for example applications of the EDDH assumption.

We begin by showing that the DCR assumption [Pai99] implies the EDDH assumption.

Theorem 2 (DCR implies EDDH)**.** Let p, q be safe primes[5] and define:

- $N = pq$,
- $\mathbb{G} = \{x : x \xleftarrow{\$} \mathbb{Z}^*_{N^2}, \left(\frac{x}{N}\right) = 1\}$,
- $G = \{g^{2N} \mod N^2 : g \xleftarrow{\$} \mathbb{Z}_{N^2}\}$,

[5]Choosing p, q safe primes makes the analysis slightly simpler. See Appendix A for basic facts about $\mathbb{Z}^*_N$ when p and q are safe primes.

- $K = \{0, \ldots, \lfloor N^2/4 \rfloor\} = \{0, \ldots, (N^2-1)/4\}$,
- $\mathbb{H} = \{(1+aN) : a \in \mathbb{Z}_N\} = \{(1+N)^a \mod N^2 : a \in \mathbb{Z}_N\}$.

Then under the DCR assumption the EDDH assumption is hard in the group $\mathbb{G}$.

Proof. Define the following distributions Let $\hat{G} = \{g^{2N}(1+N) \mod N^2 : g \xleftarrow{\$} \mathbb{Z}_{N^2}\}$.

$$\begin{aligned}
\Lambda_1 &= \{(g, g^a, g^b, g^{ab}) : g \xleftarrow{\$} G, a \xleftarrow{\$} K, b \xleftarrow{\$} K\} \\
\Lambda_2 &= \{(g, x, g^b, x^b) : g \xleftarrow{\$} G, x \xleftarrow{\$} \hat{G}, b \xleftarrow{\$} K\} \\
\Lambda_3 &= \{(g, x, g^b, x^b h) : g \xleftarrow{\$} G, x \xleftarrow{\$} \hat{G}, b \xleftarrow{\$} K, h \xleftarrow{\$} \mathbb{H}\} \\
\Lambda_4 &= \{(g, g^a, g^b, g^b h) : g \xleftarrow{\$} G, a \xleftarrow{\$} K, b \xleftarrow{\$} K, h \xleftarrow{\$} \mathbb{H}\}
\end{aligned}$$

1. The DCR assumption says $\{g^2 \mod N^2 : g \xleftarrow{\$} \mathbb{Z}_{N^2}\} \approx_c \{g^{2N} \mod N^2 : g \xleftarrow{\$} \mathbb{Z}_{N^2}\}$. Thus

$$\begin{aligned}
G &= \{g^{2N} \mod N^2 : g \xleftarrow{\$} \mathbb{Z}_{N^2}\} \\
&\approx_c \{g^2 \mod N^2 : g \xleftarrow{\$} \mathbb{Z}_{N^2}\} \\
&= \{g^2(1+N) \mod N^2 : g \xleftarrow{\$} \mathbb{Z}_{N^2}\} \\
&\approx_c \{g^{2N}(1+N) \mod N^2 : g \xleftarrow{\$} \mathbb{Z}_{N^2}\} \\
&= \hat{G}.
\end{aligned}$$

   Now, notice that for a fixed generator g of G,

$$\{g^a \mod N^2 : a \xleftarrow{\$} K\} \approx_s \{g^a \mod N^2 : a \xleftarrow{\$} \{0, 1, \ldots, \varphi(N)/4\}\} \approx_s G$$

   (See Corollary 6 in Appendix B for a rigorous proof of this fact). We also know that with all but negligible probability a uniformly chosen element $g \xleftarrow{\$} G$ will be a generator for G, so this implies $\Lambda_1 \approx_c \Lambda_2$.

2. If $x = g_1^{2N}(1+N)$, then $x^b = g_1^{2Nb}(1+N)^b = g_1^{2N(b \mod N\varphi(N)/4)}(1+N)^{b \mod N} \mod N^2$. Since the distribution of b is statistically close to uniform modulo $N\varphi(N)/4$, we have that b is statistically close to uniform modulo N even conditioned on any value of b modulo $\varphi(N)/4$. Since the order of g is $\varphi(N)/4$, the distribution of b modulo N is statistically close to uniform conditioned on g^b. Thus, even conditioned on g^b, the distribution of x^b is statistically close to $g_1 h$ where $g_1 \xleftarrow{\$} G$, and $h \xleftarrow{\$} \mathbb{H}$, which shows $\{(g, x, g^b, x^b)\} \approx_s \{(g, x, g^b, x^b h)\}$. Thus $\Lambda_2 \approx_s \Lambda_3$.

3. We have already observed that $G \approx_c \hat{G}$, so $\Lambda_3 \approx_c \Lambda_4$.

□

It is standard to conserve randomness by sampling $a \xleftarrow{\$} \{0, \ldots, (N-1)/4\}$, and $b \xleftarrow{\$} \{0, \ldots, (N^2-1)/4\}$. It is easy to see that security is preserved in this case as well. Since the exposition is cleaner if they are sampled from the same space, and a few DDH applications require it, our scheme samples them from the same larger space.

Next, we show that the QR assumption implies the EDDH assumption.

Theorem 3 (QR Implies EDDH)**.** Let p, q be safe primes with $p = q = 3 \mod 4$, and define:

- $N = pq$,
- $\mathbb{G} = \{x : x \xleftarrow{\$} \mathbb{Z}_N^*, \left(\frac{x}{N}\right) = 1\}$,
- $G = \{g^2 \mod N : g \xleftarrow{\$} \mathbb{Z}_N\}$,
- $K = \{0, \ldots, \lfloor N/2 \rfloor\}$,
- $\mathbb{H} = \{\pm 1\}$.

Then under the QR assumption the EDDH assumption is hard in the group $\mathbb{G}$.

Proof. Since $p = q = 3 \mod 4$, -1 is a quadratic non-residue modulo N with jacobi symbol 1.

Define the following distributions

$$\Lambda_1 = \{(g, g^a, g^b, g^{ab}) : g \xleftarrow{\$} G, a \xleftarrow{\$} K, b \xleftarrow{\$} K\}$$
$$\Lambda_2 = \{(g, x, g^b, x^b) : g \xleftarrow{\$} G, x \xleftarrow{\$} \mathbb{G}, b \xleftarrow{\$} K\}$$
$$\Lambda_3 = \{(g, x, g^b, x^b h) : g \xleftarrow{\$} G, x \xleftarrow{\$} \mathbb{G}, b \xleftarrow{\$} K, h \xleftarrow{\$} \mathbb{H}\}$$
$$\Lambda_4 = \{(g, g^a, g^b, g^b h) : g \xleftarrow{\$} G, a \xleftarrow{\$} K, b \xleftarrow{\$} K, h \xleftarrow{\$} \mathbb{H}\}$$

1. The QR assumption says

$$\mathbb{G} = \{x : x \xleftarrow{\$} \mathbb{Z}_N^*, \left(\frac{x}{N}\right) = 1\} \approx_c \{g^2 \mod N : g \xleftarrow{\$} \mathbb{Z}_N\} = G$$

Now, notice that for a fixed generator g of G,

$$\{g^a \mod N : a \xleftarrow{\$} K\} \approx_s \{g^a \mod N : a \xleftarrow{\$} \{0, 1, \ldots, \varphi(N)/4\}\} \approx_s G$$

(See Corollary 5 in Appendix B for a rigorous proof of this fact.) We also know that with all but negligible probability a uniformly chosen element $g \xleftarrow{\$} G$ will be a generator for G, so this implies $\Lambda_1 \approx_c \Lambda_2$.

2. If $x = -g_1^2$, then $x^b = g_1^{2b}(-1)^b = g_1^{2(b \mod \varphi(N)/4)}(-1)^{b \mod 2} \mod N$. Since the distribution of b is statistically close to uniform modulo $\varphi(N)/2$, we have that b is statistically close to uniform modulo 2 even conditioned on any value of b modulo $\varphi(N)/4$. Since the order of g is $\varphi(N)/4$, the distribution of b modulo 2 is statistically close to uniform conditioned on g^b. Thus, even conditioned on g^b, the distribution of x^b is statistically close to $g_1 h$ where $g_1 \xleftarrow{\$} G$, and $h \xleftarrow{\$} \{\pm 1\}$, which shows $\{(g, x, g^b, x^b)\} \approx_s \{(g, x, g^b, x^b h)\}$. Thus $\Lambda_2 \approx_s \Lambda_3$.

3. We have already observed that $G \approx_c \mathbb{G}$, so $\Lambda_3 \approx_c \Lambda_4$.

□

As in the case of the DCR based schemes, it is standard to conserve randomness by sampling a from a smaller space than b. In particular, we can sample $a \xleftarrow{\$} \{0, \ldots, (N-1)/4\}$, and $b \xleftarrow{\$} \{0, \ldots, (N^2-1)/4\}$. For the reasons outlined above we present this simpler (though slightly less efficient) variant.

4.1 Lossy Trapdoor Functions from EDDH

Peikert and Waters [PW08] gave a construction of lossy trapdoor functions from the Decisional Diffie-Hellman (DDH) assumption. We show that the same construction goes through under the EDDH assumption. This immediately gives new constructions of LTFs based on the QR assumption and the DCR assumption.

This provides the first construction of full LTFs from the QR assumption, and a novel construction of LTFs from the DCR assumption. For completeness, Appendices F and H give examples of the general construction in this section when instantiated with the QR and DCR assumptions.

Fix an overlying group $\mathbb{G}$, and $\mathbb{H}, G, K$ where the EDDH assumption holds.

- **Key Generation:**
  Pick $g \xleftarrow{\$} G$, $r_1, \ldots, r_n \xleftarrow{\$} K$. Fix $b \in \mathbb{H} \setminus \{1\}$.
  Generate the matrix $B = (b_{ij}) \subset \mathbb{H}^{n \times n}$, where $b_{ij} = 1$ if $i \neq j$, and
  In lossy mode $b_{ii} = 1$ for all i.
  In injective mode $b_{ii} = b$.

Sample $k_1, \ldots, k_n \xleftarrow{\$} K$, and output

$$R = \begin{pmatrix} g^{r_1} \\ \vdots \\ g^{r_n} \end{pmatrix} \qquad A = \begin{pmatrix} (g^{r_1})^{k_1} b_{11} & \cdots & (g^{r_n})^{k_1} b_{1n} \\ \vdots & \ddots & \vdots \\ (g^{r_1})^{k_n} b_{n1} & \cdots & (g^{r_n})^{k_n} b_{nn} \end{pmatrix}$$

The trapdoor will be $(k_1, \ldots, k_n)$.

- **Evaluation:**
  Given a message $x = x_1, \ldots, x_n \in \{0,1\}^n$
  Given a function index R, A, calculate

$$F_{R,A}(x) = (Rx, Ax) = \left(\prod_{i=1}^{n} (g^{a_i})^{x_i}, \begin{pmatrix} \prod_{i=1}^{n} ((g^{a_i})^{k_1})^{x_i} b_{1i}^{x_i} \\ \vdots \\ \prod_{i=1}^{n} ((g^{a_i})^{k_n})^{x_i} b_{ni}^{x_i} \end{pmatrix} \right)$$
$$= \left(g^{\sum_{i=1}^{n} a_i x_i}, \begin{pmatrix} g^{k_1 \sum_{i=1}^{n} a_i x_i} \prod_{i=1}^{n} b_{1i}^{x_i}) \\ \vdots \\ g^{k_n \sum_{i=1}^{n} a_i x_i} \prod_{i=1}^{n} b_{ni}^{x_i} \end{pmatrix} \right).$$

- **Trapdoor:**
  Given a value (Rx, Ax), and a trapdoor $(k_1, \ldots, k_n)$, we decrypt as in El-Gamal.

  Multiplying $(Rx)^{-k_i}$ by the ith component of Ax gives $\prod_{j=1}^{n} b_{ij}^{x_j} = b_{ii}^{x_i}$. In injective mode $b_{ii} = b \neq 1$, and $x_i \in \{0,1\}$, so since b is known, we can recover the x_i by inspection.

Lemma 4. In Lossy Mode, the image of F has size at most $|\mathbb{G}|$.

Proof. Notice that in Lossy Mode, since $b_{ij} = 1$ for all i, j,

$$F_{R,A}(x) = \left(g^{\sum_{i=1}^{n} a_i x_i}, \begin{pmatrix} g^{k_1 \sum_{i=1}^{n} a_i x_i} \\ \vdots \\ g^{k_n \sum_{i=1}^{n} a_i x_i} \end{pmatrix} \right).$$

which depends only on the group element $g^{\sum_{i=1}^{n} a_i x_i}$. Thus the size of the image is bounded by the order of g which is bounded by the order of the group $\mathbb{G}$ and in particular is independent of n. □

Thus by taking $n > \log(|\mathbb{G}|)$, we can make the lossy mode of F as lossy as desired.

Lemma 5. The Injective and Lossy Modes are computationally indistinguishable.

Proof. This proof is essentially identical to the original Peikert-Waters proof in [PW08].
To prove the indistinguishability, we proceed via hybrid argument, on the columns of A. Let D_i be the distribution on function indices, where the first i columns of B are the diagonal matrix with $b \neq 1$ along the diagonal, and the last $n - i$ columns of B have 1s along the diagonal. Thus D_n is the distribution output by the Injective Sampling Algorithm, and D_0 is the distribution output by the Lossy Sampling algorithm. To show that D_0 and D_n are indistinguishable, it suffices to show that D_{i-1} and D_i are indistinguishable for each $i \in \{1, \ldots, n\}$.

Given a distinguisher that distinguishes D_{i-1} from D_i we create a distinguisher for the EDDH El-Gamal cryptosystem based on EDDH (See Figure 2).

Suppose $(g^{r_*}, g^{k_*}, g^{r_* k_*} b_*)$ is received from the EDDH challenger. Generate $r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_n \xleftarrow{\$} K$ and $k_1, \ldots, k_{i-1}, k_{i+1}, \ldots, k_n \xleftarrow{\$} K$. Now, set $b_{jj} = b$ for $j < i$, and $b_{jj} = 1$ for $j > i$. Now, set

$$R = \begin{pmatrix} g^{r_1} \\ \vdots \\ g^{r_{i-1}} \\ g^{r_*} \\ g^{r_{i+1}} \\ \vdots \\ g^{r_n} \end{pmatrix}, \quad A = \begin{pmatrix} (g^{r_1})^{k_1} b_{11} & \cdots & (g^{r_{i-1}})^{k_1} b_{1,i-1} & (g^{r_*})^{k_1} b_{1i} & (g^{r_{i+1}})^{k_1} b_{1,i+1} & \cdots & (g^{r_n})^{k_1} b_{1n} \\ & & & \vdots & & & \\ (g^{k_*})^{r_1} b_{11} & \cdots & (g^{k_*})^{r_{i-1}} b_{i,i-1} & (g^{r_*})^{k_*} b_* & (g^{k_*})^{r_{i+1}} b_{i,i+1} & \cdots & (g^{k_*})^{r_n} b_{in} \\ & & & \vdots & & & \\ (g^{r_1})^{k_n} b_{n1} & \cdots & (g^{r_{i-1}})^{k_n} b_{n,i-1} & (g^{r_*})^{k_n} b_{ni} & (g^{r_{i+1}})^{k_n} b_{n,i+1} & \cdots & (g^{r_n})^{k_n} b_{nn} \end{pmatrix}$$

It is not hard to see that (R, A) can be generated using the information given by the EDDH challenger, and if $b_* = b$, then the distribution is identical to D_i and if $b_* = 1$ the distribution is identical to D_{i-1}. Thus any distinguisher that can distinguish D_i from D_{i-1} immediately breaks the EDDH assumption. □

5 Conclusion

In this work, we showed that the intuition that hash proof systems are a natural generalization of the Decisional Diffie-Hellman (DDH) assumption holds in the case of lossy trapdoor functions as well. In particular, we showed that the construction of lossy trapdoor functions from DDH given in [PW08] can be made to work with any smooth homomorphic projective hash (or any diverse group system). This shows an interesting connection between these two powerful primitives and provides the first generic[6] construction of lossy trapdoor functions from *any* primitive.

When applied to the results of [BFO08], we obtain the first construction of deterministic encryption from smooth homomorphic hash proof systems. Combining our

[6] i.e. not based on specific number theoretic assumptions

work with the negative results of [RS09], we obtain a black-box separation between one-way trapdoor permutations and smooth homomorphic hash proof systems.

To reinforce the intuition that the DCR and QR assumptions can be used to replace the DDH assumption, we introduced the Extended Decisional Diffie Hellman (EDDH) assumption and showed that the DCR and QR assumptions imply the EDDH assumption. This provides a simple method for converting most DDH-based protocols into protocols whose security can be based on either the DCR or QR assumptions. In particular, this framework gives novel constructions of LTFs from the DCR assumption, and the first known constructions of fully lossy trapdoor functions from the QR assumption.

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.

[BG10] Zvika Brakerski and Shafi Goldwasser. Circular and leakage resilient public-key encryption under subgroup indistinguishability. In *Proceedings of the 30th annual conference on Advances in cryptology*, CRYPTO'10, pages 1–20, Berlin, Heidelberg, 2010. Springer-Verlag.

[BHHO08] Dan Boneh, Shai Halevi, Mike Hamburg, and Rafail Ostrovsky. Circular-secure encryption from decision diffie-hellman. In *CRYPTO '08*, 2008.

[BHK11] Mark Braverman, Avinatan Hassidim, and Yael Tauman Kalai. Leaky pseudo-entropy functions. In *ICS '11*, 2011.

[CS98] Ronald Cramer and Victor Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *CRYPTO 1998*, volume 1462 of *Lecture Notes in Computer Science*, pages 13–25, 1998.

[CS02] Ronald Cramer and Victor Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In - *EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 45–64, 2002. Full version available at `http://eprint.iacr.org` Cryptology ePrint Archive, Report 2001/085.

[FGK+10] David Mandell Freeman, Oded Goldreich, Eike Kiltz, Alon Rosen, and Gil Segev. More constructions of lossy and correlation-secure trapdoor functions. In *PKC '10*, 2010.

[GL89] Oded Goldreich and L. Levin. A hard-core predicate for all one-way functions. In *STOC '89*, pages 25–32. ACM, 1989.

[HK07] Shai Halevi and Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. Cryptology ePrint Archive, Report 2007/118, 2007. http://eprint.iacr.org/2007/118.

[Kal05] Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. In *EUROCRYPT '05*, pages 78–95, 2005.

[MY09] Petros Mol and Scott Yilek. Chosen-ciphertext security from slightly lossy trapdoor functions. http://eprint.iacr.org/2009/524, 2009.

[NP01] Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In *SODA '01*, pages 448–457. ACM/SIAM, 2001.

[Pai99] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238. Springer Berlin / Heidelberg, 1999.

[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.

[PVW08] Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A framework for efficient and composable oblivious transfer. In David Wagner, editor, *CRYPTO '08*, volume 5157 of *Lecture Notes in Computer Science*, pages 554–571. Springer, 2008.

[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.

[RS91] Charles Rackoff and Daniel Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *CRYPTO '91*, pages 433–444, 1991.

[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.

Appendix

A Safe Primes

A safe prime is a prime number p such that $p = 2p' + 1$ for some prime p'. The prime p' is called a Sophie Germain prime.

By choosing our moduli as a product of safe primes, we can ensure that the group of quadratic residues has nice structure, in particular it is cyclic, and with high probability a uniformly chosen element will generate the group. This is not strictly necessary, but it eliminates the need to test if the elements are generators and makes the exposition somewhat simpler.

Lemma 6. If p, q are safe primes and $N = pq$, then the group

$$\mathbb{G} = \{x \mod N : \left(\frac{x}{n}\right) = 1\}$$

is cyclic with order $2p'q'$, and $\mathbb{G}$ has $(p'-1)(q'-1)$ generators.

Proof. $\varphi(N) = (p-1)(q-1) = 4p'q'$. Now, $\mathbb{G}$ has index 2 in $\mathbb{Z}_N^*$, so $|\mathbb{G}| = 2p'q'$. By the fundamental theorem of finite abelian groups $\mathbb{G} \simeq \mathbb{Z}_2 \times \mathbb{Z}_{p'} \times \mathbb{Z}_{q'}$, since these groups are all cyclic and their orders are relatively prime, their cartesian product is cyclic.

An element $g = (g_1, g_2, g_3) \in G_1 \times G_1 \times G_3$ will be a generator if and only if each g_i generates G_i. Since $\mathbb{Z}_m$ has $\varphi(m)$ generators, we conclude that $\mathbb{G}$ has $\varphi(2)\varphi(p')\varphi(q') = (p'-1)(q'-1)$ generators. □

Lemma 7. If p, q are safe primes and $N = pq$, then the group of quadratic residues

$$\mathbb{G} = \{x \mod N : \left(\frac{x}{n}\right) = 1\}$$

is cyclic with order $p'q'$, and $\mathbb{G}$ has $(p'-1)(q'-1)$ generators.

Proof. $\varphi(N) = (p-1)(q-1) = 4p'q'$. Now, $\mathbb{G}$ has index 4 in $\mathbb{Z}_N^*$, so $|\mathbb{G}| = p'q'$. By the fundamental theorem of finite abelian groups $\mathbb{G} \simeq \mathbb{Z}_{p'} \times \mathbb{Z}_{q'}$, since these groups are all cyclic and their orders are relatively prime, their cartesian product is cyclic.

An element $g = (g_1, g_2, g_3) \in G_1 \times G_1 \times G_3$ will be a generator if and only if each g_i generates G_i. Since $\mathbb{Z}_m$ has $\varphi(m)$ generators, we conclude that $\mathbb{G}$ has $\varphi(p')\varphi(q') = (p'-1)(q'-1)$ generators. □

Lemma 8. If p, q are safe primes and $N = pq$, then the group of quadratic residues

$$\mathbb{G} = \{x \mod N : \left(\frac{x}{n}\right) = 1\}$$

with all but negligible probability, a uniformly chosen $g \xleftarrow{\$} \mathbb{G}$ will generate $\mathbb{G}$.

Proof. By Lemma 7 the group, $\mathbb{G}$, has $(p'-1)(q'-1)$ generators, so the probability that a uniformly chosen element is a generator is

$$\Pr_{g \xleftarrow{\$} \mathbb{G}}[\ g \text{ generates } \mathbb{G}] = \frac{(p'-1)(q'-1)}{p'q'} \geq 1 - \frac{p'+q'}{p'q'} = 1 - \frac{1}{p'} - \frac{1}{q'}.$$

Notice that in practice $p' \approx q' \approx \frac{1}{2}\sqrt{N}$. □

Similar arguments show that if $\mathbb{G} = \{x^{2N} \mod N^2\}$, then $\mathbb{G}$ is cyclic of order $\varphi(N)/4$, and a uniformly chosen element will be a generator with all but negligible probability.

B Statistical Distance

Lemma 9. Assume $N > m$, and define the distributions

$$\begin{aligned} \Lambda_1 &= \{x : x \xleftarrow{\$} \mathbb{Z}_m\} \\ \Lambda_2 &= \{x \mod m : x \xleftarrow{\$} [N]\} \\ \Lambda_3 &= \{x : x \xleftarrow{\$} [N]\} \end{aligned}$$

Then if $N \mod m = r$, we have

$$\Delta(\Lambda_1, \Lambda_2) = \frac{r(m-r)}{Nm} \leq \frac{\min(r, m-r)}{N} \leq \frac{m}{N}$$

and

$$\Delta(\Lambda_1, \Lambda_3) = 1 - \frac{m}{N}.$$

Proof. Let $N = \ell m + r$, with $r < m$. Then under the distribution Λ_1 each element in $\mathbb{Z}_m$ appears with probability $\frac{1}{m}$, and under the distribution of Λ_2 we have r elements that appear with probability $\frac{\ell+1}{N}$ and $(m-r)$ elements that appear with probability $\frac{\ell}{N}$.

$$\begin{aligned} \Delta(\Lambda_1, \Lambda_2) &= \frac{1}{2}\left[r\left(\frac{\ell+1}{N} - \frac{1}{m}\right) + (m-r)\left(\frac{1}{m} - \frac{\ell}{N}\right)\right] \\ &= \frac{1}{2}\left[r\frac{(\ell+1)m - N}{Nm} + (m-r)\frac{N-\ell m}{Nm}\right] \\ &= \frac{1}{2}\left[\frac{r(m-r)}{Nm} + \frac{(m-r)r}{Nm}\right] \\ &= \frac{r(m-r)}{Nm}. \end{aligned}$$

To show the second equality, we observe that

$$\begin{aligned}\Delta(\Lambda_1, \Lambda_3) &= \frac{1}{2}\left[\sum_{i=1}^{m}\left(\frac{1}{m} - \frac{1}{N}\right) + \sum_{i=m+1}^{N}\frac{1}{N}\right] \\ &= \frac{1}{2}\left[m\left(\frac{1}{m} - \frac{1}{N}\right) + (N-m)\frac{1}{N}\right] \\ &= \frac{1}{2}\left[m\frac{N-m}{Nm} + \frac{N-m}{N}\right] \\ &= \frac{N-m}{N} \\ &= 1 - \frac{m}{N}\end{aligned}$$

□

Corollary 4. If $N = pq$, then the uniform distribution on $\{0, \ldots, \lfloor N/2 \rfloor\}$ is statistically close to the uniform distribution modulo $\varphi(N)/2$.

Proof. By Lemma 9 the statistical distance between the two distributions is equal to $1 - \frac{\varphi(N)/2}{\lfloor N/2 \rfloor}$. Using the fact that $\varphi(N) = (p-1)(q-1) = pq - p - q + 1 = N - p - q + 1$, and $\lfloor N/2 \rfloor = (N-1)/2$, we have

$$1 - \frac{\varphi(N)/2}{(N-1)/2} = 1 - \frac{N-p-q+1}{N-1} = \frac{p+q-2}{N-1} \approx 2N^{-\frac{1}{2}}.$$

□

Corollary 5. If $N = pq$, and $g = g_1^2 \mod N$, then

$$\{g^a \mod N : a \xleftarrow{\$} [\varphi(N)/4]\} \approx_s \{g^a \mod N : a \xleftarrow{\$} [\lfloor N/2 \rfloor]\}.$$

Proof. Since g is a quadratic residue modulo N we know $g^{\varphi(N)/4} = 1 \mod N$, thus the distribution of $g^a \mod N$, only depends on $a \mod \varphi(N)/4$. Let $\Lambda_1 = \{a \mod \varphi(N)/4 : a \xleftarrow{\$} [\varphi(N)/4]\}$, and $\Lambda_2 = \{a \mod \varphi(N)/4 : a \xleftarrow{\$} [\lfloor N/2 \rfloor]\}$, Now, $(N-1)/2 = 2\varphi(N)/4 + (p+q-2)/2$, so by Lemma 9

$$\Delta(\Lambda_1, \Lambda_2) \leq \frac{p+q-2}{N-1} \approx 2N^{-\frac{1}{2}}.$$

□

Corollary 6. If $N = pq$, and $g = g_1^{2N} \mod N^2$, then

$$\{g^a \mod N : a \xleftarrow{\$} [\varphi(N)/4]\} \approx_s \{g^a \mod N : a \xleftarrow{\$} [\lfloor N^2/4 \rfloor]\}.$$

Proof. We know $g^{\varphi(N)/4} = 1 \mod N^2$, thus the distribution of $g^a \mod N$, only depends on $a \mod \varphi(N)/4$. Let $\Lambda_1 = \{a \mod \varphi(N)/4 : a \xleftarrow{\$} [\varphi(N)/4]\}$, and $\Lambda_2 = \{a \mod \varphi(N)/4 : a \xleftarrow{\$} [\lfloor N^2/4 \rfloor]\}$, By Lemma 9

$$\Delta(\Lambda_1, \Lambda_2) \leq \frac{\varphi(N)/4}{(N^2-1)/4} < \frac{1}{N}.$$

□

C Diverse Group Systems

In [CS02], Cramer and Shoup defined diverse group systems and used them as a foundation for all their constructions of Projective Hash Proof Systems. We review these definitions here.

Let X, L, Π be finite abelian groups written additively, with $L \subsetneq X$. Let $\mathrm{Hom}(X, \Pi)$ be the group of homomorphisms, $\varphi : X \to \Pi$. This is also clearly an abelian group under the operation $(\varphi_1 + \varphi_2)(x) = \varphi_1(x) + \varphi_2(x)$.

Definition 7. Let X, L, Π be finite abelian groups with $L \subsetneq X$. Let $\mathcal{H} \subset \mathrm{Hom}(X, \Pi)$, We call

$$\mathcal{G} = (\mathcal{H}, X, L, \Pi),$$

a *group system*.

We will require that the groups $X, L, \mathcal{H}$ are efficiently samplable, and that the homomorphisms $\varphi \in \mathcal{H}$ are efficiently computable.

Definition 8. We call a group system $\mathcal{G} = (\mathcal{H}, X, L, \Pi)$ *diverse* if for all $x \in X \setminus L$, there exists $\varphi \in \mathcal{H}$ such that $\varphi(\ell) = 0$ for all $\ell \in L$, but $\varphi(x) \neq 0$.

Now, we review some of the basic algebra that underlies group systems.

Definition 9. Let $\mathcal{G} = (\mathcal{H}, X, L, \Pi)$ be a group system. For $Y \subset X$, define $\mathcal{A}(Y) = \mathrm{Ann}(Y) \cup \mathcal{H}$, i.e.

$$\mathcal{A}(Y) = \{\varphi \in \mathcal{H} : \varphi(y) = 0 \ \forall y \in Y\}.$$

It is easy to see that $\mathcal{G}$ is diverse if and only if for all $x \in X \setminus L$, $\mathcal{A}(L \cup \{x\}) \subsetneq \mathcal{A}(L)$.

We also define

Definition 10. Let $\mathcal{G}$ be a group system. For $x \in X$, define $\mathcal{I}(x)$ to be the image of the homomorphisms in $\mathcal{A}(L)$ applied to x, i.e.

$$\mathcal{I}(x) = \{\pi \in \Pi : \exists \varphi \in \mathcal{A}(L) \text{ s.t. } \varphi(x) = \pi\}.$$

Lemma 10. Let $\mathcal{G} = (\mathcal{H}, X, L, \Pi)$ be a diverse group system, and suppose p is the smallest prime dividing $|X/L|$, then $p \leq |\mathcal{I}(x)|$ for all $x \in X \setminus L$.

Proof. Fix $x \in X \setminus L$, and let

$$\begin{aligned} \mathcal{E} : \mathcal{A}(L) &\to \Pi \\ \varphi &\mapsto \varphi(x). \end{aligned}$$

Then $\mathrm{Ker}(\mathcal{E}) = \mathcal{A}(L \cup \{x\})$, and $\Im(\mathcal{E}) = \mathcal{I}(x)$, so the first isomorphism theorem tells us that $\mathcal{A}(L)/\mathcal{A}(L \cup \{x\}) \simeq \mathcal{I}(x)$, in particular, $\mathcal{I}(x) > 1$, and $|\mathcal{I}(x)| \mid |\mathcal{A}(L)|$. Let q be a prime that divides $|\mathcal{I}(x)|$, then $q \mid |\mathcal{A}(L)|$. It remains to show that $q \mid |X/L|$. Let $d = |X/L|$, then for all $x \in X$, $dx \in L$. Since $q \mid |\mathcal{A}(L)|$, $\mathcal{A}(L)$ contains an element of order q, call it φ. But $(d\varphi)(x) = \varphi(dx) = 0$ for all $x \in X$, so $q \mid d$. Thus any prime divisor of $|\mathcal{I}(x)|$ is a prime divisor of $|X/L|$, so it must be at least p. □

In particular, Lemma 10 gives a minimum size for $\mathcal{I}(x)$.

Now, suppose $\varphi \leftarrow \mathcal{H}$. If the action of φ on L is completely specified, then φ is fixed up to an element in $\mathcal{A}(L)$. Thus for $x \in X \setminus L$, the value of $\varphi(x)$ is known up to an element in $\mathcal{I}(x)$. In particular, only the coset of $\mathcal{I}(x)$ in $\Pi/\mathcal{I}(x)$ is fixed by the action of φ on L.

In [CS02] Cramer and Shoup show a natural method for constructing universal hash proof systems from Diverse Group Systems.

Definition 11. Let $\mathcal{G} = (\mathcal{H}, X, L, \Pi)$ be a diverse group system, and let $g_1, \ldots, g_d \in L$ be a set of generators for L. We define the associated hash proof system $\mathbf{H} = (H, K, X, L, \Pi, S, \alpha)$,

- For uniformly chosen $k \in K$, H_k is uniform on $\mathcal{H}$.
  Without loss of generality, we may assume $K = \mathcal{H}$, and $k = \varphi \in \mathcal{H}$.
  We maintain universal hash proof notation to emphasize that $H_k(\cdot)$ that someone who can calculate $H_k(\cdot)$ on elements of L may not know the underlying homomorphism φ.
- $S = \Pi^d$, and

$$\begin{aligned} \alpha : K &\to S \\ k &\mapsto (H_k(g_1), \ldots, H_k(g_d)). \end{aligned}$$

Although it was not required as a general property in the Cramer Shoup constructions, we note that for Projective Hash Proof Systems derived from Diverse Group Systems, $H_k(x_1) + H_k(x_2) = H_k(x_1 + x_2)$ since the H_k are in $\mathrm{Hom}(X, \Pi)$. We require this property for our construction of Lossy Trapdoor Functions. We emphasize, however, that this is the only additional property of a Smooth Projective Hash that we require. In particular, our construction will work for any Smooth Projective Hash Family that satisfies $H_k(x_1) + H_k(x_2) = H_k(x_1 + x_2)$, even if it was not derived from a Diverse Group System.

We note too, that Projective Hash Proofs derived from Diverse Group systems may have the additional property that they are homomorphic over the keys, i.e.

$H_{k_1}(x) + H_{k_2}(x) = H_{k_1+k_2}(x)$. We will *not* make use of this property, but this additionally homomorphic property may have value in future constructions.

D Lossy Trapdoor Functions from Diverse Group Systems

Let $\mathcal{G} = (\mathcal{H}, X, L, \Pi)$ be a diverse group system based on a hard subset membership problem and let $\mathbf{H} = (H, K, X, L, \Pi, S, \alpha)$ be its associated projective hash family.

We slightly modify the construction of Section 3 to work in this context. The only change will be to the key generation algorithm, where the diagonal elements b_i can no longer be fixed arbitrarily, but must depend on x_i.

- **Key Generation:**
  Pick $x_1, \ldots, x_n \in L$.
  For each x_i, choose $b_i \leftarrow \mathcal{I}(x_i) \setminus \{0\}$.
  Generate the matrix $B = (B_{ij}) \subset \Pi^{n \times n}$, where $B_{ij} = 0$ if $i \neq j$, and
  In lossy mode $B_{ii} = 0$ for all i.
  In injective mode $B_{ii} = b_i$.

  $k_1, \ldots, k_n \leftarrow K$, and output

$$R = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \qquad A = \begin{pmatrix} H_{k_1}(x_1) + B_{11} & \cdots & H_{k_1}(x_n) + B_{1n} \\ \vdots & \ddots & \vdots \\ H_{k_n}(x_1) + B_{n1} & \cdots & H_{k_n}(x_n) + B_{nn} \end{pmatrix}$$

  The trapdoor will be $(k_1, \ldots, k_n)$.

- **Evaluation:**
  Given a message $z = z_1, \ldots, z_n \in \{0,1\}^n$
  Given a function index R, A, calculate

$$F_{R,A}(z) = (Rz, Az) = \left(\sum_{i=1}^{n} z_i x_i, \begin{pmatrix} \sum_{i=1}^{n} z_i (H_{k_1}(x_i) + B_{1i}) \\ \vdots \\ \sum_{i=1}^{n} z_i (H_{k_n}(x_i) + B_{ni}) \end{pmatrix} \right).$$

- **Trapdoor:**
  Given a value (Rz, Az), and a trapdoor $(k_1, \ldots, k_n)$, we begin by noting that the homomorphic property of H_k guarantees that

$$\begin{aligned} F_{R,A}(z) &= (Rz, Az) \\ &= \left(\sum_{i=1}^{n} z_i x_i, \begin{pmatrix} \sum_{i=1}^{n} z_i (H_{k_1}(x_i) + B_{1i}) \\ \vdots \\ \sum_{i=1}^{n} z_i (H_{k_n}(x_i) + B_{ni}) \end{pmatrix} \right) \end{aligned}$$

$$= \left(\sum_{i=1}^{n} z_i x_i, \begin{pmatrix} H_{k_1}\left(\sum_{i=1}^{n} z_i x_i\right) + \sum_{i=1}^{n} z_i B_{1i}) \\ \vdots \\ H_{k_n}\left(\sum_{i=1}^{n} z_i x_i\right) + \sum_{i=1}^{n} z_i B_{ni}) \end{pmatrix} \right)$$

Since $\sum_{i=1}^{n} z_i x_i$, and k_i is known, we can calculate $H_{k_i}\left(\sum_{i=1}^{n} z_i x_i\right)$ and subtract it from each component to recover the vector

$$\left(\sum_{i=1}^{n} z_i B_{1i}, \cdots, \sum_{i=1}^{n} z_i B_{ni} \right)^t.$$

Now, in injective mode, $B_{ij} = 0 \in \Pi$ for $i \neq j$, and $B_{ij} = b_i$ for $i = j$, so

$$\left(\sum_{i=1}^{n} z_i B_{1i}, \cdots, \sum_{i=1}^{n} z_i B_{ni} \right)^t = (z_1 b_1, \cdots, z_n b_n).$$

Since the $z_i \in \{0, 1\}$, and the b_i are known, we can recover the z_i by inspection.

The proof that the two modes are indistinguishable is almost identical to the proof of Lemma 2.

E Using the EDDH Assumption

E.1 El-Gamal under EDDH

It is simple to create an El-Gamal type cryptosystem under EDDH instead of DDH

An EDDH group consists of a group $\mathbb{G}$, a subgroup $\mathbb{H}$ and a set of integers $K \subset \mathbb{Z}$.

Remark: Instantiating an El-Gamal type cryptosystem based on the DCR or the QR assumptions yields a cryptosystem where the factorization *is not part of the secret key.*

An extremely useful property of El-Gamal is its homomorphic property. To make this scheme additively homomorphic, we follow the standard technique to modify the scheme as follows

Notice that decryption now requires solving the discrete log problem in $\langle h \rangle$. In the EDDH schemes under the DCR assumption, $h = (1 + N)$ and, as Paillier observed, the discrete-log problem is easy in this group. Under the QR assumption, $h = -1$, so $|\langle h \rangle| = 2$ and the discrete log problem is easy in this setting as well. It is only when instantiating the El-Gamal cryptosystem using the standard DDH assumption that we end up in a situation where we cannot solve the discrete-log

- **Key Generation:**
  $g \xleftarrow{\$} G$.
  $k \xleftarrow{\$} K$. The public key will be $f = g^k$, and the secret key will be k.
- **Encryption:**
  To encrypt a message $m \in \mathbb{H}$, choose a random $r \xleftarrow{\$} K$ and set
  $$c = (g^r, f^r m)$$
- **Decryption:**
  Given a ciphertext (g_1, g_2), calculate $m = g_2(g_1^{-k})$.

Figure 1: El-Gamal from the EDDH Assumption

- **Key Generation:**
  $g \xleftarrow{\$} G$.
  $k \xleftarrow{\$} K$. fix an element $h \in \mathbb{H}$. The public key will be $f = g^k, h$, and the secret key will be k.
- **Encryption:**
  To encrypt a message $m \in [L]$ for $L < |\langle h \rangle|$, choose a random $r \xleftarrow{\$} K$ and set
  $$c = (g^r, f^r h^m)$$
- **Decryption:**
  Given a ciphertext (g_1, g_2), calculate $h^m = g_2(g_1^{-k})$ Solve the discrete log problem in $\langle h \rangle$ to recover m.

Figure 2: Additively Homomorphic El-Gamal from the EDDH Assumption

problem in the plaintext group, and we are forced to restrict the plaintext to be small integers so we can solve the discrete log problem by brute force. This limits the applicability of the homomorphism. This provides some evidence that El-Gamal type cryptosystems are actually more natural under the DCR and QR assumptions than under the original DDH assumption.

F Lossy Trapdoor Functions from the QR Assumption

For completeness, we show how the general construction of LTFs from the EDDH assumption (given in Section 4.1) looks when instantiated with the QR assumption.

We begin by recalling the QR assumption

F.1 The Quadratic Residuosity Assumption

We briefly review the definition of the quadratic residuosity assumption. Let $N = pq$ be the product of two primes. Let $J \subset \mathbb{Z}_N^*$ be the subset of elements with Jacobi symbol 1, i.e.

$$J = \left\{ x \in \mathbb{Z}_N^* : \left(\frac{x}{N}\right) = 1 \right\}.$$

Let $QR \subset X$ be the set of quadratic residues modulo N,

$$QR = \{x \in \mathbb{Z}_N^* : \exists y \in \mathbb{Z}_N^* \text{ s.t. } y^2 = x \mod N\}.$$

Definition 12 (The Quadratic Residuosity (QR) Assumption)**.** The *Quadratic Residuosity* assumption states that the sets QR and $J \setminus QR$ are computationally indistinguishable.

F.2 Lossy Trapdoor Functions from the QR Assumption

In this section we show how to construct Lossy Trapdoor Functions (LTFs) from the Quadratic Residuosity (QR) assumption. This construction is just a slight modification of our general construction in Section 3 when applied to the construction of universal hash proof systems from quadratic residuosity in [CS02] (variation 2). Let $N = pq$ be the product of two safe primes, i.e. $p = 2p' + 1$, and $q = 2q' + 1$, for primes p', q'. Then $|J| = 2p'q'$, and $|QR| = p'q'$. Thus QR is a cyclic group with only two proper subgroups, a subgroup of order p' and one of order q'. Thus QR has $|QR| - p' - q' + 1$ generators, so with all but negligible probability a randomly chosen element of QR will generate QR. The public parameters consist of a uniformly chosen $\mu \leftarrow \mathbb{Z}_N^*$, and $g = \mu^2 \mod N$. It is not hard to see that the distribution of g is statistically close to uniform over the generators of the cyclic group QR. We are now ready to describe our construction.

- **Sampling Injective Functions:**
  Let $B = (b_{ij})$ be the $n \times n$ identity matrix.
  Sample $w_1, \ldots, w_n \leftarrow W$.
  Sample $k_1, \ldots, k_n \leftarrow K$.
  Set $h_i = g^{w_i} \mod N$,
  Let

$$R = \begin{pmatrix} h_1 \\ \vdots \\ h_n \end{pmatrix} \qquad A = \begin{pmatrix} (-1)^{b_{11}} h_1^{k_1} & \cdots & (-1)^{b_{1n}} h_n^{k_1} \\ \vdots & \ddots & \vdots \\ (-1)^{b_{n1}} h_1^{k_n} & \cdots & (-1)^{b_{nn}} h_n^{k_n} \end{pmatrix}.$$

Where all the operations are done in the multiplicative group $\mathbb{Z}_N^*$. The function index will be (R, A), and the trapdoor will be $(k_1, \ldots, k_n)$.

- **Sampling Lossy Functions:**
  This is identical to sampling the Injective Functions, only $B = (b_{ij})$ is set to be the $n \times n$ zero matrix.
- **Evaluation:**
  Given a message $z = z_1 \cdots z_n \in \{0,1\}^n$, and a function index (R, A), output Rz, Az, where

$$Rz = \prod_{i=1}^{n} h_i^{z_i} \mod N = g^{\sum_{i=1}^{n} w_i z_i} \mod N,$$

  and

$$Az = \begin{pmatrix} \prod_{j=1}^{n} A_{1j}^{z_j} \mod N \\ \vdots \\ \prod_{j=1}^{n} A_{nj}^{z_j} \mod N \end{pmatrix} = \begin{pmatrix} (-1)^{\sum_{i=1}^{n} b_{1i} z_i} g^{k_1 \sum_{i=1}^{n} w_i z_i} \mod N \\ \vdots \\ (-1)^{\sum_{i=1}^{n} b_{ni} z_i} g^{k_n \sum_{i=1}^{n} w_i z_i} \mod N \end{pmatrix}.$$

  In particular, Rz and Az are the standard matrix products (written in multiplicative notation, instead of additive notation).
- **Trapdoor:**
  Given a value $Rz = r$, and $Az = (a_1, \ldots, a_n)$, set

$$\begin{aligned} m_i' &= a_i r^{-k_i} \mod N \\ &= (-1)^{\sum_{j=1}^{n} b_{ij} z_j} g^{k_i \sum_{j=1}^{n} w_j z_j} \left(\prod_{j=1}^{n} h_j^{z_j} \right)^{-k_i} \mod N \\ &= (-1)^{z_i} \end{aligned}$$

  Then set $m_i = 0$ if $m_i' = 1$, and $m_i = 1$ if $m_i' = -1$.

Remarks: Our construction does not use the factorization of N as a trapdoor. Instead, the trapdoor information is actually a set of discrete logarithms of elements in the cyclic group QR, and given the trapdoor information factoring N still remains infeasible.

We note also that our construction is completely different from the construction of slightly lossy trapdoor functions from the QR assumption given in [FGK+10]. Their construction shows that if inputs are taken from a well chosen subgroup of $\mathbb{Z}_N^*$, then the function $f(x) = x^2 \mod N$ will be a slightly lossy trapdoor function. In their construction the lossy branch loses less than 1 bit of entropy, by the results of Mol and Yilek, [MY09], this is enough to achieve correlated product security. Our results, however, can lose any polynomial fraction of the input bits.

G Slightly Lossy Functions from the QR Assumption

While the constructions from LTFs in [PW08] require the lossy branch to lose many bits, in [MY09], Mol and Yilek considered LTDFs that lose only a fraction of a single bit. The called these *Slightly Lossy Trapdoor Functions*. As a warmup, before constructing full lossy trapdoor functions from the Quadratic Residuosity (QR) assumption, we give a simple, intuitive construction of slightly lossy functions from the QR assumption. In particular, the lossy branch of this family loses only a single bit of information, and the family has no trapdoor.

- **Sampling Injective Functions:**
  Generate safe primes $p, q \leftarrow \mathcal{PRIMES}(()\lambda)$, i.e. $p = 2p' + 1$, and $q = 2q' + 1$ for primes p' and q', and set $N = pq$. Let g be a generator of the cyclic group J. Note $|J| = 2p'q'$.
  The function index will be (g, N) and the trapdoor will be (p, q).
- **Sampling Lossy Functions:**
  Generate safe primes $p, q \leftarrow \mathcal{PRIMES}(()\lambda)$, i.e. $p = 2p' + 1$, and $q = 2q' + 1$ for primes p' and q', and set $N = pq$. Let g be a generator of the cyclic group QR. Note $|QR| = p'q'$.
  The function index will be (g, N) and the trapdoor will be $\perp$.
- **Evaluation:**
  Given a message $x \in [\lfloor N/2 \rfloor]$,
  let $F((g, N), x) = g^x \mod N$.

The indistinguishability of branches is exactly the QR Assumption. To see that the lossy branch is actually lossy notice that the uniform distribution on the set $\{0, 1, \ldots, N/2\}$ is only negligibly far from uniform on $\{0, 1, \ldots, |J|\}$, we have that in injective mode, F will be injective with all but negligible probability, while in lossy mode, the output of F only depends on $x \mod |QR|$. Since $|J| = 2|QR|$, we have that in lossy mode, the family loses 1 bit of information.

H Lossy Trapdoor Functions from the DCR Assumption

In this section, we show how the general construction of Section 4.1 looks when instantiated with the DCR assumption.

We begin by reviewing the definition of Paillier's [Pai99] decisional composite residuosity assumption.

Let $N = pq$ be the product of two safe primes. Then the DCR assumption roughly says that the set of Nth powers modulo N^2 is computationally indistinguishable from the uniform distribution modulo N^2. Let $\xi = 1 + N$. Then $\xi^a = 1 + aN \mod N^2$, so ξ has order N in $\mathbb{Z}^*_{N^2}$.

Definition 13 (The Decisional Composite Residuosity (DCR) Assumption)**.** The Decisional Composite Residuosity (DCR) assumption states that

$$\{x^N \mod N^2 : x \in \mathbb{Z}^*_{N^2}\} \approx_c \{x : x \in \mathbb{Z}^*_{N^2}\}.$$

Let $L = \{x^{2N} \mod N^2 : x \in \mathbb{Z}^*_{N^2}\}$, and $X = \{x^2 \mod N^2 : x \in \mathbb{Z}^*_{N^2}\}$. It is immediate that $L \subset X$ is a hard subset membership problem under the DCR assumption. We choose to work with squares because this makes L and X cyclic groups, which simplifies the exposition somewhat.

Let $N = pq$ be the product of two safe primes, i.e. $p = 2p'+1$, and $q = 2q'+1$, for primes p', q'. Choose $\mu \leftarrow \mathbb{Z}^*_{N^2}$ uniformly, and let $g = \mu^{2N} \mod N^2$. It is not hard to see that the distribution of g is statistically close to uniform over the generators of the cyclic group L. See Appendix B for details.

The construction is as follows:

- **Sampling Injective Functions:**
  Let $B = (b_{ij})$ be the $n \times n$ identity matrix.
  Sample $w_1, \ldots, w_n \leftarrow W$.
  Sample $k_1, \ldots, k_n \leftarrow K$.
  Set $h_i = g^{w_i} \mod N^2$,
  Let
  $$R = \begin{pmatrix} h_1 \\ \vdots \\ h_n \end{pmatrix} \qquad A = \begin{pmatrix} \xi^{b_{11}} h_1^{k_1} & \cdots & \xi^{b_{1n}} h_n^{k_1} \\ \vdots & \ddots & \vdots \\ \xi^{b_{n1}} h_1^{k_n} & \cdots & \xi^{b_{nn}} h_n^{k_n} \end{pmatrix}.$$
  Where all the operations are done in the multiplicative group $\mathbb{Z}^*_{N^2}$. The function index will be (R, A), and the trapdoor will be $(\{w_i\}, \{k_i\})$.
- **Sampling Lossy Functions:**
  This is identical to sampling the Injective Functions, only $B = (b_{ij})$ is set to be the $n \times n$ zero matrix.
- **Evaluation:**
  Given a message $z = z_1 \cdots z_n \in [N]^n$, and a function index (R, A), output Rz, Az, where
  $$Rz = \prod_{i=1}^{n} h_i^{z_i} \mod N^2 = g^{\sum_{i=1}^n w_i z_i} \mod N^2,$$
  and
  $$Az = \begin{pmatrix} \prod_{j=1}^n A_{1j}^{z_j} \mod N^2 \\ \vdots \\ \prod_{j=1}^n A_{nj}^{z_j} \mod N^2 \end{pmatrix} = \begin{pmatrix} \xi^{\sum_{i=1}^n b_{1i} z_i} g^{k_1 \sum_{i=1}^n w_i z_i} \mod N^2 \\ \vdots \\ \xi^{\sum_{i=1}^n b_{ni} z_i} g^{k_n \sum_{i=1}^n w_i z_i} \mod N^2 \end{pmatrix}.$$
  In particular, Rz and Az are the standard matrix products (written in multiplicative notation, instead of additive notation).

- **Trapdoor:**
  Given a value $Rz = r$, and $Az = (a_1, \ldots, a_n)$, set

$$\begin{aligned} m'_i &= a_i r^{-k_i} \mod N^2 \\ &= \xi^{\sum_{j=1}^n b_{ij} z_j} g^{k_i \sum_{j=1}^n w_j z_j} \left(\prod_{j=1}^n w_j^{z_j} \right)^{-k_i} \mod N^2 \\ &= \xi^{z_i} \end{aligned}$$

  Then, given $\xi^{z_i} = (1+N)^{z_i} = (1 + z_i N) \mod N^2$, we can set $m_i = \frac{\xi^{z_i} - 1}{N}$.

Notice that this construction is more efficient than the construction based on quadratic residuosity, because the input can be taken from $[N]^n$ and not $\{0,1\}^n$. We remark, however, that this construction is still less efficient than the construction of LTFs based on the DCR assumption given in [BFO08] and [RS08].

I IND-CCA Security

We review the notion of security against a chosen-ciphertext attack (IND-CCA) given in [RS91].

We imagine a game played between a challenger and an adversary. The challenger has a public key cryptosystem (G, E, D) and runs the key generation algorithm to generate a public key and secret key $(pk, sk) \leftarrow G(1^\lambda)$, the adversary then sends pk to the adversary $\mathcal{A}$.

Initially we set the target ciphertext $c^* = \perp$.

- **Challenge Query:** The adversary sends two messages m_0, m_1 to the challenger. The challenger chooses $b \leftarrow \{0,1\}$, and randomness r and returns an encryption $c = E(pk, m_b, r)$ to the adversary. The challenger then sets the target ciphertext $c^* = c$.
- **Decryption Query:** The adversary sends a ciphertext c to the challenger. If $c \neq c^*$, the challenger runs $m = D(sk, c)$ and returns m to the adversary.

After a polynomial number of queries, exactly one of which is a challenge query the adversary outputs $b^* \in \{0,1\}$. We define increasing levels of security depending on the restrictions placed on the adversary's use of decryption queries.

Definition 14. A public key cryptosystem is IND-CPA secure if every efficient adversary $\mathcal{A}$ playing the above game never makes any decryption queries, and

$$\left| \Pr[\mathcal{A} = b] - \frac{1}{2} \right| < \nu(\lambda),$$

for some negligible function ν.

Definition 15. A public key cryptosystem is IND-CCA1 secure if every efficient adversary $\mathcal{A}$ playing the above game never makes a decryption query after the challenge query, and

$$\left|\Pr[\mathcal{A} = b] - \frac{1}{2}\right| < \nu(\lambda),$$

for some negligible function ν.

Definition 16. A public key cryptosystem is IND-CCA2 secure if every efficient adversary $\mathcal{A}$ playing the above game

$$\left|\Pr[\mathcal{A} = b] - \frac{1}{2}\right| < \nu(\lambda),$$

for some negligible function ν.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de