A Public Key Encryption In Standard Model Using Cramer-Shoup Paradigm

Mahabir Prasad Jhanwar and Rana Barua
mahabir_r, rana@isical.ac.in

Stat-Math Unit
Indian Statistical Institute
Kolkata, India

Abstract. We present a public-key encryption scheme which is provably secure against adaptive chosen ciphertext attack. The scheme is constructed using Cramer-Shoup paradigm [7]. The security of the scheme is based on the Decisional Bilinear Diffie-Hellman problem.

Keywords: IND-CCA2(adaptive chosen ciphertext security), public-key encryption, DDH- Assumption, DBDH-Assumption, Universal Projective Hashing.

1 Introduction

Indistinguishability against adaptive chosen ciphertext attack (IND-CCA2), where an adversary is given the capability to decrypt ciphertexts of his choice, with the exception of a target ciphertext, is considered to be the correct notion of security for general-purpose public-key encryption schemes. In the literature, there are a number of approaches for obtaining public key encryption schemes that are secure under this model. Much of this work, however, has been only achieved with proofs in the random oracle model [4], the most well-known being OAEP [3].

In the standard model, three main techniques have been proposed for constructing CCA-secure encryption schemes. The first approach, by Naor and Yung [5] and subsequently by Dolev, Dwork and Naor [1], builds IND-CCA2 secure schemes from any chosen-plaintext secure scheme (CPA-secure) and any non-interactive zero knowledge (NIZK) proof system. The resulting schemes, however, are too inefficient for practical use, since they use expensive NIZK proofs.

Cramer and Shoup [6] proposed the first encryption scheme that was simultaneously practical and IND-CCA2 secure in the standard model. Subsequently, Cramer and Shoup [7] generalised their encryption scheme by introducing the notion of hash proof systems (HPS) and gave a framework for constructing IND-CCA2 secure schemes using HPS constructed from a general subset membership problem. They also showed that their scheme can be regarded as a special case of their general construction.

Our contribution:

We construct a new universal$_2$ projective hash family. Consequently, using Cramer-Shoup paradigm [7], from this family we can easily get a public-key encryption scheme which is secure against adaptive chosen ciphertext attack. The security of the scheme reduces to Decisional Bilinear Diffie-Hellman problem. The underlying security assumptions of the previously constructed encryption schemes using Cramer-Shoup paradigm are the DDH assumption, Quadratic Residuosity assumption and Paillier's Decision Composite Residuosity assumption. Further we observe that our scheme can be used to derive the well known (albeit, easy) reduction 'DBDH implies DDH'. Usually hardness of the underlying assumptions are used to prove the security of a cryptographic protocol. On the contrary we describe a method which uses the security of a protocol to derive this above mention reduction.

[0] Research supported by NBHM

2 Preliminaries

2.1 Notations

- Z_q denotes the set of all congruence classes modulo q
- $|S|$ denotes the cardinality of S if S is a set.
- If S is a set, $x \in_R S$ denotes the experiment of choosing $x \in S$ at random.

2.2 Bilinear Groups

We briefly review the necessary facts about the bilinear maps and bilinear map friendly groups. We use the following notation:

1. G and $\tilde{G}$ are two cyclic groups of prime order q.
2. g is any non zero element of G.
3. $e : G \times G \to \tilde{G}$ a bilinear map.

We use additive notation for the group operation in G and mulitiplicative notation in $\tilde{G}$. Let G and $\tilde{G}$ be two groups as above. A bilinear map (pairing map) is a map $e : G \times G \to \tilde{G}$ with the following properties.

1. Bilinearity: for all $x, y \in G$ and $a, b \in Z$, we have $e(ax, by) = e(x, y)^{ab}$.
2. Non-degeneracy: $e(g, g) \neq 1$.

We say that G is a bilinear group if the group action in G can be computed efficiently and there exists a group $\tilde{G}$ and an efficiently computable bilinear map $e : G \times G \to \tilde{G}$ as above. Note that the map e is symmetric since $e(ag, bg) = e(g, g)^{ab} = e(bg, ag)$. Examples of such maps are the (modified) Weil and Tate pairings.

2.3 Public-Key Encryption Scheme

We recall the definition of a public-key encryption scheme and the notion of security against adaptive chosen ciphertext attack.

A public key encryption scheme provides three algorithms:

KeyGen(l): a probabilistic, polynomial-time key-generation algorithm that on input 1^l, where $l \geq 0$, is a security parameter, outputs a public-key/private-key pair (PK, SK). A public key PK specifies a finite message space M_{PK}.

Encrypt(PK, M): a probabilistic, polynomial-time encryption algorithm that takes as input a public-key PK, 1^l and a message M. It outputs a ciphertext.

Decrypt(SK, C): a polynomial-time decryption algorithm that takes as input a private key SK and a ciphertext C. It outputs a plaintext message or the special symbol reject.

2.4 Adaptive Chosen Ciphertext Security(IND-CCA2)

The strongest and commonly accepted notion of security for a public key encryption system is that of indistinguishability against an adaptive chosen ciphertext attack. This notion, denoted IND-CCA2, is defined using the following game between a challenger and an adversary $\mathcal{A}$. Both are given the security parameter $l \in Z^+$ as input.

Setup The challenger runs KeyGen(l) to obtain a random instance of public and private key pair (PK, SK). It gives the public key PK to the adversary.

Query phase-1 The adversary adaptively issues decryption queries C where $C \in \{0,1\}^*$. The challenger responds with Decrypt(SK, C).

Challenge The adversary outputs two (equal length) messages M_0, M_1. The challenger picks a random $b \in \{0,1\}$ and sets C^*=Encrypt(PK, M_b). It gives C^* to the adversary.

Query Phase-2 The adversary continue to issue decryption queries C as in Phase 1, with the added constraint that $C \neq C^*$. The challenger responds with Decrypt(SK, C).

Guess Adversary outputs its guess $\hat{b} \in \{0,1\}$ for b and wins the game if $b = \hat{b}$.

The above game is commonly known as the IND-CCA2 game. We define the advantage of $\mathcal{A}$ in this game as $Adv_{cca}^{\mathcal{A}}(l) = |Pr[b = \hat{b}] - \frac{1}{2}|$. An encryption system is (t, q, ϵ)-IND-CCA2 secure if there is no randomized algorithm $\mathcal{A}$ that runs in time t, makes at most q decryption queries, and has advantage at least ϵ in the IND-CCA2 game.

3 Security Assumption

Decisional Diffie-Hellman Assumption

Let G be an Abelian group of order q, where q is a large prime. Let g_1 be a generator of G. Consider the following two distributions. Let

$$\hat{\mathbf{D}} = \{(g_1, g_2, u_1 = g_1^r, u_2 = g_2^r) | r \in Z_q, g_2 = g_1^w, w \in Z_q\}$$

$$\hat{\mathbf{R}} = \{(g_1, g_2, u_1 = g_1^{r_1}, u_2 = g_2^{r_2}) | r_1, r_2 \in Z_q, r_1 \neq r_2, g_2 = g_1^w, w \in Z_q\}$$

The decisional Diffie-Hellman (DDH) assumption claims that the distributions $\hat{\mathbf{D}}$ and $\hat{\mathbf{R}}$ are indistinguishable.

Decisional Bilinear Diffie-Hellman Assumption

Let G be a bilinear group of prime order q. Let $e : G \times G \rightarrow \tilde{G}$ be the bilinear map. The decisional bilinear Diffie-Hellman problem in G is as follows:

Given $g, ag, bg, cg \in G$ and $T \in \tilde{G}$, where a, b, c are random elements in Z_q, g is random element in G, T is random element in $\tilde{G}$. We say an algorithm $\mathcal{A}$ that outputs $l \in \{0,1\}$ has advantage ϵ in solving the DBDH problem in G if

$$|Pr[A(g, ag, bg, cg, e(g,g)^{abc}) = 0] - Pr[(g, ag, bg, cg, T) = 0]| \geq \epsilon$$

We refer to the distribution on the left as $\mathbf{D}$ and the distribution on the right as $\mathbf{R}$.

4 Universal Projective Hashing

In this section we review universal projective hashing introduced by Cramer and Shoup [7].

Definition 1 *Let X and Π be finite, non empty sets. Let $H = (H_k)_{k \in K}$ be a collection of functions indexed by K, so that for every $k \in K$, H_k is a function from X into Π. Note that we may have $H_k = H_{k'}$ for $k \neq k'$. We call $\mathbf{F}$ =(H, K, X, Π) a hash family and each H_k a hash function.*

We now introduce the concept of universal projective hashing. Let $\mathbf{F}$ =(H, K, X, Π) be a hash family. Let L be a nonempty, proper subset of X. Let S be a finite, nonempty set, and let $\alpha : K \rightarrow S$ be a function. Set $\mathbf{H}$=$(H, K, X, L, \Pi, S, \alpha)$.

Definition 2 $\mathbf{H}$=$(H, K, X, L, \Pi, S, \alpha)$, *defined as above, is called a projective hash family for (X, L) if for all $k \in K$, the action of H_k on L is determined by $\alpha(k)$ [7].*

Definition 3 *Let* $\mathbf{H}$*=*$(H, K, X, L, \varPi, S, \alpha)$ *be a projective hash family , and let* $\epsilon \geq 0$ *be a real number. Consider the probability space defined by choosing* $k \in K$ *at random.*
Then H *is said to be* ϵ*-universal*$_1$ *if for all* $s \in S$*,* $x \in X \backslash L$ *and* $\pi \in \varPi$*, it holds that*

$$Pr[H_k(x) = \pi \wedge \alpha(k) = s] \leq \epsilon Pr[\alpha(k) = s].$$

$\mathbf{H}$ *is said to be* ϵ*-universal*$_2$ *if for all* $s \in S$*,* $x, x^* \in X$ *and* $\pi, \pi^* \in \varPi$ *with* x *doesnot belongs to* $L \cup \{x^*\}$*, it holds that*

$$Pr[H_k(x) = \pi \wedge H_k(x^*) = \pi^* \wedge \alpha(k) = s] \leq \epsilon Pr[H_k(x^*) = \pi^* \wedge \alpha(k) = s].$$

Definition 4 *Let* $\mathbf{H}$*=*$(H, K, X, L, \varPi, S, \alpha)$ *be a projective hash family. Consider the probability space defined by choosing* $k \in_R K$*,* $x \in_R X \backslash L$ *and* $\pi' \in_R \varPi$*. We say for* $\epsilon \geq 0$*, a real number, the above family is* ϵ*-smooth if*

$$|\Pr[H_k(x) = \pi'] - \frac{1}{|\varPi|}| \leq \epsilon$$

4.1 Diverse Group System

Let X, L and $\varPi$ be finite abelian groups, L be a proper subgroup of X.

Let $\text{Hom}(X, \varPi)$ denote the group of all homomorphisms $\phi : X \to \varPi$. This is also a finite abelian group. If we use additive notation for the group operation in X, L and $\varPi$ then the group operation in $\text{Hom}(X, \varPi)$ is as follows. For $\phi, \phi' \in \text{Hom}(X, \varPi)$, $x \in X$, and $\alpha \in Z$, we have $(\phi + \phi')(x) = \phi(x) + \phi'(x), (\phi - \phi')(x) = \phi(x) - \phi'(x)$, and $(a\phi)(x) = a\phi(x) = \phi(ax)$. The zero element of $\text{Hom}(X, \varPi)$ sends all elements of X to $0 \in \varPi$.

Definition 5 *Let* X*,* L*,* $\varPi$ *be as above. Let* H *be a subgroup of* $Hom(X, \varPi)$*. Then* $\mathbf{G}$*=*$(H, X, L, \varPi)$ *is called a group system.*

Definition 6 *Let* $\mathbf{G}$*=*$(H, X, L, \varPi)$ *be a group system. We say that* $\mathbf{G}$ *is diverse if for all* $x \in X \backslash L$*, there exists* $\phi \in H$ *such that* $\phi(L) =< 0 >$*, i.e.* ϕ *vanishes on* L *but* $\phi(x) \neq 0$*.*

4.2 Subset-Membership Problem (SMP)

A subset membership problem SMP specifies a collection $\{instance_n\}_{n \in N}$ such that for every n, $instance_n$ is a probability distribution over problem instance $\varLambda$. Each $\varLambda$ specifies the following:

Non-empty sets, X, L and W such that L is a proper subset of X.

A binary relation $R \subset X \times W$ such that $x \in L$ iff $(x, w) \in R$ for some witness $w \in W$.

SMP requires that the following Probabilistic polynomial time algorithms exist.

1. Instance sampling: samples an instance $\varLambda$ according to $instance_n$ on system parameter1^n.
2. Subset sampling: outputs a random $x \in L$ together with a witness $w \in W$ for x on input 1^n and $\varLambda[X, L, W, R]$.
3. Element sampling: outputs a random $x \in X$.

The SMP is said to be hard if $(\varLambda, x_0)$ and $(\varLambda, x_1)$ are indistinguishable for a random $x_0 \in L$ and a random $x_1 \in X \backslash L$.

4.3 Hash proof systems

Let $\mathbf{M}$ be a susbet membership problem specifying a sequence $(I_l)_{l\geq 0}$ of instance distrubutions. A hash proof system(HPS) $\mathbf{P}$ for $\mathbf{M}$ associates with each instance $\Lambda[X, L, W, R]$ of $\mathbf{M}$ a projective hash family $\mathbf{H}$=$(H, K, X, L, \Pi, S, \alpha)$ for (X, L). Additionally, $\mathbf{P}$ provides several algorithms to carry out basic operations we have defined for an associated projective hash family; namely, sampling $k \in K$ at random, computing $\alpha(k) \in S$ given $k \in K$, computing $H_k(x) \in \Pi$ given $k \in K$ and $x \in X$. This later algorithm is called private evaluation algorithm for $\mathbf{P}$. Moreover, a crucial property is that the system provides an efficient algorithm to compute $H_k(x) \in \Pi$, given $\alpha(k) \in S$, $x \in L$ and $w \in W$, where w is a witness for x. This algorithm is called public evaluation algorithm for $\mathbf{P}$.

4.4 Universal hash proof systems

Definition 7 *Let $\epsilon(l)$ be a function mapping non-negetive integers to non-negetive reals. Let $\mathbf{M}$ be a subset membership problem specifying a sequence $(I_l)_{l\geq 0}$ of instance distributions. Let $\mathbf{P}$ be an HPS for $\mathbf{M}$. We say that $\mathbf{P}$ is $\epsilon(l)$-universal (respectively , -universal$_2$, -smooth) if there exists a negligible function $\delta(l)$ such that for all $l \geq 0$ and for all $\Lambda[X, L, W, R] \in [I_l]$, the projective hash family $\mathbf{H}$=$(H, K, X, L, \Pi, S, \alpha)$ that $\mathbf{P}$ associates with Λ is $\delta(l)$-close (see [7]) to an $\epsilon(l)$-universal((respectively, -universal$_2$, -smooth) projective hash family $\mathbf{H}^*$=$(H^*, K^*, X, L, \Pi, S, \alpha^*)$.*

Moreover, if this is the case, and $\epsilon(l)$ is a negligible function, then we say that $\mathbf{P}$ is strongly universal (respectively, universal$_2$, smooth).

Definition 8 *For $l \geq 0$ and for all $\Lambda[X, L, W, R] \in [I_l]$ an extended HPS $\boldsymbol{P}$ for $\boldsymbol{M}$ associates with Λ a finite set E along with a projective hash family $\boldsymbol{H}$=$(H, K, X \times E, L \times E, \Pi, S, \alpha)$ for $(X \times E, L \times E)$. All the related properties of HPS can similarly be defined for extended HPS.*

5 The generic Cramer-Shoup scheme

We now describe the Cramer-Shoup generic mothod for constructing a secure public-key encryption scheme. The method requires

- $\mathbf{M}$, a subset membership problem
- a strongly smooth hash proof system $\mathbf{P}$ for $\mathbf{M}$
- a strongly universal$_2$ extended hash proof system $\hat{\mathbf{P}}$ for $\mathbf{M}$

To simplyfy things, we will describe the scheme with respect to a fixed problem instance $\Lambda[X, L, W, R]$. With Λ fixed, let $\mathbf{H}$=$(H, K, X, L, \Pi, S, \alpha)$ be the projective hash family that $\mathbf{P}$ associates with Λ, and let $\hat{\mathbf{H}}$=$(\hat{H}, \hat{K}, X \times \Pi, L \times \Pi, \hat{\Pi}, \hat{S}, \hat{\alpha})$ be the projective hash family that $\hat{\mathbf{P}}$ associates with Λ . It is required that Π is an abelian group, for which we use additive notation, and that elements of Π can be efficiently added and substracted. The message space is Π.

- **Key Generation** Choose $k \in_R K$ and $\hat{k} \in_R \hat{K}$. Compute $s = \alpha(k) \in S$ and $\hat{s} = \hat{\alpha}(\hat{k}) \in \hat{S}$. Note that all these operations can be efficiently performed using the algorithms provided by $\mathbf{P}$ and $\hat{\mathbf{P}}$.
  The public key is $(s, \hat{s})$.
  The private key is $(k, \hat{k})$.
- **Encryption**
  To encrypt a message $m \in \Pi$ under the public key as above, one does the following. Choose $x \in_R L$ together with a corresponding witness $w \in W$, using the subset sampling algorithm provided by $\mathbf{M}$.

Compute $\pi = H_k(x) \in \Pi$, using the public evaluation algorithm for $\mathbf{P}$ on inputs s, x, and w.
Compute $e = m + \pi \in \Pi$.
Compute $\hat{\pi} = \hat{H}_{\hat{k}}(x, e) \in \hat{\Pi}$, using the public evaluation algorithm for $\hat{\mathbf{P}}$ on inputs $\hat{s}$, x, e, and w.
The ciphertext is $(x, e, \hat{\pi})$.

- **Decryption**
  To decrypt a ciphertext $(x, e, \hat{\pi}) \in X \times \Pi \times \hat{\Pi}$ under a secret key as above, one does the following.
  Compute $\hat{\pi'} = \hat{H}_{\hat{k}}(x, e) \in \hat{\Pi}$, using the private evaluation algorithm for $\hat{\mathbf{P}}$ on inputs $\hat{k}$, x, and e.
  Check wheather $\hat{\pi} = \hat{\pi'}$; if not , then output reject and halt.
  Compute $\pi = H_k(x) \in \Pi$, using the private evaluation algorithm for $\mathbf{P}$ on inputs k and x.
  Compute $m = e - \pi \in \Pi$, and output the message m.

Theorem 1. *The above scheme is secure against adaptive chosen ciphertext attack, assuming* $\mathbf{M}$ *is a hard subset membership problem.*

Proof. See [7] □

6 The Proposed Scheme

Let G, $\tilde{G}$ be two prime order groups and $\hat{e} : G \times G \to \tilde{G}$, a bilinear map. We use additive notation for the group operation in G and multiplicative for $\tilde{G}$. Let q be the order of G, $\tilde{G}$, where q is a prime. Let

- $\Gamma : G \times \tilde{G} \times \tilde{G} \to Z_q^*$ be a collision resistant hash function.
- $f : G \to \tilde{G}$ be an efficiently computable isomorphism.

Key-Generation
Choose $g_0, g_1, g_2 \in_R G$ and $k_0, k_1, k_{00}, k_{01}, k_{10}, k_{11} \in_R Z_q^*$. Compute

$$s_0 = f(g_0)^{k_0}.\hat{e}(g_1, g_2)^{k_1}$$
$$s_1 = f(g_0)^{k_{00}}.\hat{e}(g_1, g_2)^{k_{01}}$$
$$s_2 = f(g_0)^{k_{10}}.\hat{e}(g_1, g_2)^{k_{11}}$$

The public-key pk=$(g_0, g_1, g_2, s_0, s_1, s_2)$ and secret-key sk=$(k_0, k_1, k_{00}, k_{01}, k_{10}, k_{11})$.

Encryption
To encrypt a message $m \in \tilde{G}$,

- choose $w \in_R Z_q^*$
- compute (x, y)=$(wg_0, \hat{e}(g_1, g_2)^w)$
- compute π=s_0^w
- compute e=$\pi.m$
- compute $\hat{\pi}$=$(s_1.s_2^t)^w$, where $t = \Gamma(x, y, e)$
- Ciphertext is $(x, y, e, \hat{\pi})$

Decryption
To decrypt $(x, y, e, \hat{\pi})$ using the secret-key sk=$(k_0, k_1, k_{00}, k_{01}, k_{10}, k_{11})$,

- compute $\hat{\pi}' = (f(x)^{k_{00}}.y^{k_{01}}).(f(x)^{k_{10}}.y^{k_{11}})^t$, where $t = \Gamma(x, y, e)$
- check if $\hat{\pi}$=$\hat{\pi}'$; if not, then output ‘reject’ and halt.
- compute π=$(f(x)^{k_0}.y^{k_1})$
- compute m=$e.\pi^{-1}$.

We first verify that this is an encryption scheme, in the sense that the decryption of an encryption of a message yields the message. One can see that for a valid ciphertext $(x, y, e, \hat{\pi})$ of the message m the decryption algorithm first computes $\hat{\pi}'$ which is, one can see, is equal to $\hat{\pi}$ as follows

$$\begin{aligned}
\hat{\pi}' &= (f(x)^{k_{00}}.y^{k_{01}}).(f(x)^{k_{10}}.y^{k_{11}})^t \\
&= (f(wg_0)^{k_{00}}.(\hat{e}(g_1,g_2)^w)^{k_{01}}).(f(wg_0)^{k_{10}}.(\hat{e}(g_1,g_2)^w)^{k_{11}})^t \\
&= (f(g_0)^{k_{00}}.(\hat{e}(g_1,g_2))^{k_{01}})^w.(f(g_0)^{k_{10}}.(\hat{e}(g_1,g_2))^{k_{11}})^{w.t} \\
&= ((f(g_0)^{k_{00}}.(\hat{e}(g_1,g_2))^{k_{01}}).(f(g_0)^{k_{10}}.(\hat{e}(g_1,g_2))^{k_{11}})^{.t})^w \\
&= (s_1.s_2^t)^w \\
&= \hat{\pi}
\end{aligned}$$

Then see that

$$\begin{aligned}
(f(x)^{k_0}.y^{k_1}) &= (f(wg_0)^{k_0}.(\hat{e}(g_1,g_2)^w)^{k_1}) \\
&= (f(g_0)^{k_0}.(\hat{e}(g_1,g_2))^{k_1})^w \\
&= s_0^w \\
&= \pi
\end{aligned}$$

Note: The map f in scheme can be taken as follows. For fixed non-zero element $g_0 \in G$, $f(x) = \hat{e}(g_0, x)$ for all $x \in G$. This is clearly an isomorphism from $G \to \tilde{G}$. Also $\hat{e}(g_1, g_2)$ can be taken as part of pulic key.

6.1 Security

Theorem 2. *Assuming the hardness of Decisional Bilinear Diffie-Hellman (DBDH) assumption the proposed public-key encryption scheme is secure against adaptive chosen ciphertext attack (IND-CCA2).*

Proof. To prove the security of the proposed scheme we show the scheme to be an particular instantantiation of Cramer-Shoup paradigm. We single out the universal$_1$ and universal$_2$ projective hash families which build the above scheme according to the Cramer-Shoup paradigm. □

6.2 Security Analysis of the Proposed Scheme

As we mention earlier, we will show the proposed scheme to be an particular instantantiation of the Cramer-Shoup generic scheme. To this end we describe the following objects essential to Cramer-Shoup paradigm which we use in our scheme. They are

- a subset membership problem $\mathbf{M}$
- a smooth projective hash family $\mathbf{H}$=$(H, K, X, L, \Pi, S, \alpha)$ which can be obtained given an instance $\Lambda[X, L, W, R]$ of the subset membership problem $\mathbf{M}$
- a universal$_2$ projective hash family $\hat{\mathbf{H}}$=$(\hat{H}, \hat{K}, X \times \Pi, L \times \Pi, \hat{\Pi}, \hat{S}, \hat{\alpha})$ which can be obtained given the instance $\Lambda[X, L, W, R]$

One can observe that describing an efficient way to obtain the above objects will suffices as it gives the required Hash Proof Systems for the paradigm. To this end we first describe the subset membership problem and then given an instance of the subset membership problem we will construct a diverse group system. Using this diverse group system we derive the smooth projective hash family and universal$_2$ projective hash family. This group theoretic method to construct the above families is due to Cramer-Shoup [7].

6.3 The Subset-Membership Problem

Let G be a bilinear group and $e : G \times G \to \tilde{G}$ be a bilinear map, where G and $\tilde{G}$ are groups of prime order q. We use additive notation for G and multiplicative notation for $\tilde{G}$. With G, $\tilde{G}$, q and e given, we now define an instance of a subset-membership problem as follows.

Let g_0, g_1, g_2 be randomly chosen elements of G. Define X=$G \times \tilde{G}$ and let L be the subgroup of X generated by the element $(g_0, e(g_1, g_2))$. Given an element $(x_0, x_1) \in X$ a witness for (x_0, x_1) which shows that $(x_0, x_1) \in L$ is an element $w \in Z_q$ such that $(x_0, x_1) = (wg_0, e(g_1, g_2)^w)$. Obviously one can efficiently sample a random element of L together with a witness by generating $w \in Z_q$ at random and computing $(wg_0, e(g_1, g_2)^w)$.

It is clear that this defines a subset-membership problem (distinguishing between elements of L and $X \backslash L$) and we will show that the hardness of the subset-membership problem is equivalent to the hardness of DBDH-assumption as follows.

The subset membership problem intend to distinguish tuples of the form

$$\left(g_0, g_1, g_2, rg_0, \tilde{A} = e(g_1, g_2)^r\right) \qquad \text{(in L)}$$

from the tuples of the form

$$\left(g_0, g_1, g_2, rg_0, \tilde{B} = e(g_1, g_2)^{r'}\right) \qquad \text{(in } X \backslash L)$$

where g_0, g_1, g_2 are randomly chosen from G and r, r' are randomly chosen from Z_q.

Now we show how to derive an instance for DBDH problem from the given tuples. We set

$$\left(g = g_0, A = ag = rg_0, B = bg = g_1, C = cg = g_2, Z = \tilde{A}\right)$$

and

$$\left(g = g_0, A = ag = rg_0, B = bg = g_1, C = cg = g_2, Z = \tilde{B}\right)$$

It is clear, that for Z=$\tilde{A}$ the distribution of the corresponding tuple is from **D**, as $Z = \tilde{A} = e(g_1, g_2)^r = e(bg, cg)^a = e(g, g)^{abc}$. If Z=$\tilde{B}$, then clearly the distribution of the corresponding tuple is from **R**. Conversely given a valid DBDH tuple,

$$(g, ag, bg, cg, e(g, g)^{abc})$$

by setting g_0=g, g_1=bg, g_2=cg yeilds an element of the set L as follows.

$$(g_0, g_1, g_2, rg_0 = ag, e(g_1, g_2)^r = e(g, g)^{abc})$$

Thus hardness of the subset-membership reduces to the hardness of DBDH problem.

6.4 Construction of Diverse Group-System

Now as we have the subset-membership problem, we construct a Diverse Group System as follows.

We set, for a fixed non-zero element $g_0 \in G$, $f(x) = e(g_0, x)$ for all $x \in G$. This is clearly an isomorphism. With X and L defined as above, set $K = Z_q \times Z_q$. We define for each $(k_0, k_1) \in K$ a map $H_{k_0,k_1} : G \times \tilde{G} \to \tilde{G}$ as follows.
For $(x, y) \in X = G \times \tilde{G}$,

$$H_{k_0,k_1}(x, y) = f(x)^{k_0} y^{k_1}.$$

It can be checked that the correspondence $(k_0, k_1) \to H_{k_0,k_1}$ is a bijection between K and $\mathrm{Hom}(X, \tilde{G})$, the set of all group homomorphisms from X to $\tilde{G}$ (Its clear that $\mathrm{Hom}(X, \tilde{G})$ is a group).

With H=$\mathrm{Hom}(X, \tilde{G})$, we consider the group system $\mathbf{G}$= $(H, X, L, \tilde{G})$.

Claim : $\mathbf{G}$ is a diverse group system.

Given $\mathbf{G}$= $(H, X, L, \tilde{G})$ with $g_0, g_1, g_2 \in G$, $X = G \times \tilde{G}$ and $L =< (g_0, e(g_1, g_2)) >$ a subgroup of X generated by $(g_0, e(g_1, g_2))$, we set $X' = \tilde{G} \times \tilde{G}$ and $L' =< (f(g_0), e(g_1, g_2)) >$ a subgroup of X' generated by $(f(g_0), e(g_1, g_2))$.

We define H'=$\mathrm{Hom}(X', \tilde{G})$. It follows from [7] that the map $(k_0, k_1) \to H'_{(k_0,k_1)}$, where $H'_{k_0,k_1}(x, y) = x^{k_0}y^{k_1}$ is a 1-1 correspondence between $K = Z_q \times Z_q$ and H'=$\mathrm{Hom}(X', \tilde{G})$. So now we have the following two group systems.

$$\mathbf{G} = (H, X, L, \tilde{G}) \text{ and } \mathbf{G}' = (H', X', L', \tilde{G})$$

We know (by [7]) that $\mathbf{G}'$=$(H', X', L', \tilde{G})$ is a diverse group system. We will show that $\mathbf{G} = (H, X, L, \tilde{G})$ is also diverse group system.

The map $(k_0, k_1) \to H_{(k_0,k_1)}$, where $H_{k_0,k_1}(x, y) = f(x)^{k_0}y^{k_1}$ is a 1-1 correspondence between $Z_q \times Z_q$ and H

Now let (x', y') be an element which is in $X \backslash L$. Then $(x', y') = (ag_0, e(g_1, g_2)^b)$ for some $a \neq b$, $a, b \in Z_q$. We have to show that there exists $(k_0, k_1) \in Z_q \times Z_q$ such that $H_{(k_0,k_1)}(x', y') = f(x')^{k_0}.y'^{k_1} \neq 0$ but $H_{(k_0,k_1)}$ vanishes on L. Now $(f(x'), y') = (f(g_0)^a, e(g_1, g_2)^b)$, clearly $(f(x'), y') \in X' \backslash L'$. As $\mathbf{G}'$ is a diverse group system there exists a tuple $(k_0, k_1) \in Z_q \times Z_q$ such that the corresponding homomorphism H'_{k_0,k_1} vanishes on L' but $H'_{k_0,k_1}(f(x'), y') = f(x')^{k_0}y'^{k_1} \neq 0$. Since $H'_{k_0,k_1}(f(x), y) = H_{k_0,k_1}(x, y)$ for all $(x, y) \in X$, $H_{(k_0,k_1)}$ vanishes on L and $H_{(k_0,k_1)}(x', y') \neq 0$.

This clearly shows that $\mathbf{G}$=$(H, X, L, \tilde{G})$ is a diverse group system.

6.5 Construction of Smooth Projective Hash Family and Universal$_2$ Projective Hash Family

First we construct a $\frac{1}{q}$-universal$_1$ projective hash family $\mathbf{H}$=$(H, K, X, L, \mathit{\Pi}, S, \alpha)$ given the above diverse group system $\mathbf{G}$=$(H, X, L, \tilde{G})$ as follows.

With H, K, X, L as above we set $\mathit{\Pi} = \tilde{G}$ and $S = \tilde{G}$. The map $\alpha : K \to S$ is defined as follows :

$$\alpha(k_0, k_1) = H_{k_0,k_1}(g_0, e(g_1, g_2)) = f(g_0)^{k_0}e(g_1, g_2)^{k_1}$$

where $(k_0, k_1) \in K$. This resulting family is a $\frac{1}{q}$-universal$_1$ projective hash family due to the following thoerem.

Theorem 3. *[7] Let* $\mathbf{G}$=$(H, X, L, \mathit{\Pi})$ *be a diverse group system. A projective hash family derived from* $\mathbf{G}$ *in the above fashion will be* ϵ*-universal, where* $\epsilon = 1/\tilde{p}$*, and* $\tilde{p}$ *is the smallest prime dividing* $|X/L|$.

In our case $\tilde{p}$ is actually q as $|X|$=q^2 and $|L|$=q. Further one can observe that this family is 0-smooth. We now construct a $\frac{1}{q}$ universal$_2$ projective hash family using the above family.
Let $\mathit{\Gamma} : X \times \tilde{G} \to Z_q$ be a collision resistant hash function (CRHF).

Set $\hat{\mathbf{H}}$=$(\hat{H}, K \times K, X \times \tilde{G}, L \times \tilde{G}, \tilde{G}, S \times S, \hat{\alpha})$.
For $((k_0, k_1), (k_0', k_1')) \in K \times K$ define $\hat{H}_{((k_0,k_1),(k_0',k_1'))} \in \hat{H}$ as follows,

$$\hat{H}_{((k_0,k_1),(k_0',k_1'))}((x, y), e) = H_{(k_0,k_1)}(x, y).(H_{(k_0',k_1')}(x, y))^t$$

where $(x, y) \in X$, $e \in \tilde{G}$ and $\mathit{\Gamma}((x, y), e) = t$. Define the map $\hat{\alpha}$ as follows:

$$\hat{\alpha}((k_0, k_1), (k_0', k_1')) = (\alpha(k_0, k_1), \alpha((k_0', k_1'))$$

where $((k_0, k_1), (k_0', k_1')) \in K \times K$.
Theorem-3 of [7] assures that this family to be a $\frac{1}{q}$ universal$_2$ projective hash family.
Now one can observe that these two families are used in the proposed scheme according to the Cramer-Shoup paradigm. Hence by Theorem 1 the proposed scheme is IND-CCA2 secure provided Decisional Bilinear Diffie-Hellman problem is hard.

7 Relation between DBDH and DDH

By saying DBDH implies DDH we mean that if there is a polynomially-bounded algorithm A_{ddh} solving the problem DDH then we can build another polynomially-bounded algorithm A_{dbdh} with polynomially-bounded access to A_{ddh} which solves the problem DBDH.

Let $e : G \times G \to \tilde{G}$ be a bilinear map. It is easy to see that if DDH is easy in $\tilde{G}$ then the corresponding DBDH is also easy. However, we shall obtain this, albeit easy, reduction in an indirect fashion. This is just meant as an illustration of obtaining a reduction between two hardness assumptions. In fact, given a DDH oracle O_{ddh} in $\tilde{G}$, we shall simulate an adversary to mount an IND-CCA2 attack against our proposed scheme with non-negligible advantage. By Theorem 2, this will show that, there is a distinguishing algorithm for $DBDH$ with non-negligible advantage.

7.1 Reduction

Let O_{ddh} be an oracle for DDH.
We build a simulator(for an IND-CCA2 adversary), who in turn use the above DDH oracle O_{ddh}, to mount an IND-CCA2 attack against the proposed scheme with non-negligible advantage.
Let U, the user of the scheme, runs the Key-Generation algorithm as follows. It chooses $k_0, k_1, k_{00}, k_{01}, k_{10}, k_{11} \in_R Z_p^*$. It computes,

$$s_0 = f(g_0)^{k_0}.e(g_1, g_2)^{k_1}$$
$$s_1 = f(g_0)^{k_{00}}.e(g_1, g_2)^{k_{01}}$$
$$s_2 = f(g_0)^{k_{10}}.e(g_1, g_2)^{k_{11}}$$

It keeps the secret key $(k_0, k_1, k_{00}, k_{01}, k_{10}, k_{11})$ to itself and issues the public-key $(g_0, g_1, g_2, s_0, s_1, s_2)$ to the simulator.
With this the simulator directly moves to the Challenge phase without carrying out phase-1 decryption-queries.
Challenge-Phase The simulator chooses $r_0, r_1 \in_R Z_p^*$ and then computes,

$$\begin{aligned} m_0 &= (f(g_0)^{k_0}.e(g_1, g_2)^{k_1})^{r_0} \\ &= s_0^{r_0} \\ m_1 &= (f(g_0)^{k_0}.e(g_1, g_2)^{k_1})^{r_1} \\ &= s_0^{r_1} \end{aligned}$$

and gives m_0 and m_1 to the U.
Then U chooses $b \in_R \{0, 1\}$ and encrypts the message m_b as follows.

- choose $w \in_R Z_p^*$
- compute (x, y)=$(wg_0, e(g_1, g_2)^w)$
- compute π=s_0^w

- compute $e{=}\pi.m_b$
- compute $\hat{\pi}{=}(s_1.s_2^t)^w$, where $t = \Gamma(x, y, e)$
- Ciphertext is $(x, y, e, \hat{\pi})$

WIth the challenge ciphertext in hand the simulator once again avoids Phase-2. It moves to Guess Phase. **Guess-Phase** The simulator chooses $\hat{b} \in \{0, 1\}$ and computes

$$\begin{aligned} T_1 &= (f(x).y).(f(g_0).e(g_1, g_2))^{r_{\hat{b}}} \\ &= (f(wg_0).e(g_1, g_2)^w).(f(g_0).e(g_1, g_2))^{r_{\hat{b}}} \\ &= (f(g_0).e(g_1, g_2))^{w+r_{\hat{b}}} \end{aligned}$$

Now the simulator submits the following tuple

$$((f(g_0).e(g_1, g_2)), s_0, T_1, T_2 = e)$$

as a DDH query to the oracle O_{ddh}.
Now we show that the answer DDH oracle gives will clearly determine which message was encrypted. Looking at the query tuple carefully we find,

$$\begin{aligned} e &= \pi.m_b \\ &= s_0^w.m_b \\ &= (f(wg_0)^{k_0}.e(g_1, g_2)^{wk_1}).m_b \\ &= (f(g_0)^{k_0}.e(g_1, g_2)^{k_1})^w.(f(g_0)^{k_0}.e(g_1, g_2)^{k_1})^{r_b} \\ &= (f(g_0)^{k_0}.e(g_1, g_2)^{k_1})^{w+r_b} \end{aligned}$$

So the tuple is essentially

$$(\ \ (f(g_0).e(g_1, g_2)), (f(g_0)^{k_0}.e(g_1, g_2)^{k_1}), (f(g_0).e(g_1, g_2))^{w+r_{\hat{b}}}, (f(g_0)^{k_0}.e(g_1, g_2)^{k_1})^{w+r_b}\ \ )$$

Clearly , this is a valid DDH tuple iff

$$w + r_{\hat{b}} = w + r_b \text{ i.e. iff } b = \hat{b}$$

Thus the DDH oracle O_{ddh} enables the simulator to win the game ($\hat{b}$ was chosen by simulator) with non-negligible probability.

8 Conclusions

We presented a public-key encryption scheme, which is provably secure against adaptive chosen ciphertext attack. The security of the scheme is based on the Decisional Bilinear Diffie-Hellman problem. We further illustrate that given a DDH oracle O_{ddh}, we simulate an IND-CCA2 game in which an adversary (using an O_{ddh}) has non-negligible advantage of winning the game. This shows that the DBDH assumption implies the DDH assumption.

References

1. D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. In STOC. ACM Press, 1991.

2. K. Kurosawa and Y. Desmedt. A new paradigm of hybrid encryption scheme. In M. Franklin editor, Advances in Cryptology- Crypto 2004, volume 3152 of Lecture Notes in Computer Science, pages 426-442. Springer-Verlag, 2004.
3. M. Bellare and P. Rogaway. Optimal Asymmetric Encryption - How to encrypt with RSA. In EUROCRYPT '94, volume 839 of LNCS, pages 93-111. Springer-Verlag, 1994.
4. M. Bellare and P. Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In First ACM *Conference on Computer and Communications Security*, pages 62-73, 1993.
5. M. Naor and M. Yung. Public-Key Cryptosystems Provably Secure Against Chosen Ciphertext Attacks. In 22nd Annual ACM Symposium on Theory of Computing, pages 427-437, 1990.
6. R. Cramer and V. Shoup. A Practical Publlic-Key Cryptosystem Provably Secure Against Addaptive Chosen Cip hertext Attacks. In Proc. CRYPTO '98, Springer Verlag LNCS, 1998.
7. R. Cramer and V. Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. Cryptology ePrint Archive. Available at `http://eprint.iacr.org/2001/085.pdf`.