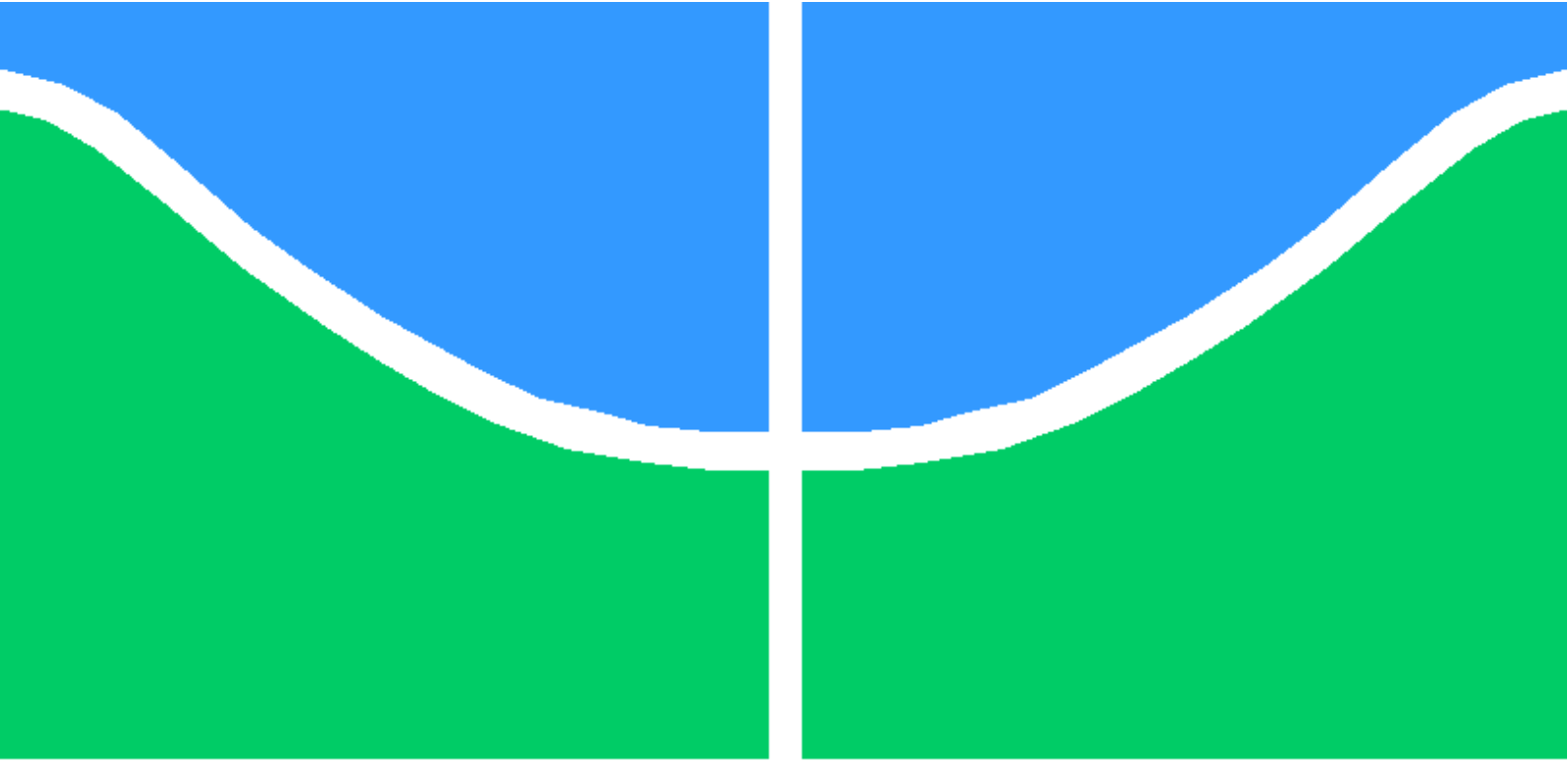

TRABALHO DE GRADUAÇÃO

# BOUNDED CHOSEN CIPHERTEXT SECURE CRYPTOSYSTEMS BASED ON COMPUTAIONAL DIFFIE-HELLMAN AND HASHED DIFFIE-HELLMAN ASSUMPTIONS

**Mayana Wanderley Pereira**

**Tobias Back Carrijo**

**Brasília, julho de 2009**

**UNIVERSIDADE DE BRASÍLIA**

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASILIA
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

# BOUNDED CHOSEN CIPHERTEXT SECURE CRYPTOSYSTEMS BASED ON COMPUTAIONAL DIFFIE-HELLMAN AND HASHED DIFFIE-HELLMAN ASSUMPTIONS

**Mayana Wanderley Pereira**

**Tobias Back Carrijo**

*Relatório submetido ao Departamento de Engenharia*

*Elétrica como requisito parcial para obtenção*

*do grau de Engenheiro Eletricista*

Banca Examinadora

Anderson C. A. Nascimento - Ph.D., UnB/ENE _____
*Orientador*

Rafael T. de Sousa Jr. - Ph.D., ENE/UnB _____
*Membro Interno*

## RESUMO

Neste trabalho propomos criptossistemas de chave pública com segurança CCA2 contra adversários que possuam um número limitado de acessos a um oráculo de decifração. Esses esquemas apresentados são baseados no esquema de cifração de ElGamal [15]. Nossas construções consideram hipóteses computacionais fracas, e o tamanho do *overhead* do texto cifrado é ótimo, uma vez que é igual ao do esquema proposto por ElGamal. A desvantagem em relação a alguns esquemas conhecidos é a limitação de acessos do adversário ao oráculo de decifração.



## ABSTRACT

We propose two constructions of chosen-ciphertext secure cryptosystems against adversaries with a bounded number of decrytion queries based on the ElGamal encryption scheme [15]. We rely our work on weak computational assumptions, and the ciphertext overhead of the resulting schemes will be only one group element which is considered optimal since it is the same as the original ElGamal. Disadvantages to known schemes are that the upper bound of the number of decryption queries has to be known before set-up phase.

# CONTENTS

# Chapter 1

# Introduction

Cryptography is a subject that encompasses data security, specially regarded with reliable communication in the presence of an adversary. It involves the the conception, the definition and the construction of computational schemes with the purpose of maintaining the security in varied systems. This aim is achieved by designing systems with the capability of protection against any kind of abuse. Such constructions must be in order to keep the system's desired functionalities, even under malicious attacks attempts intended to deviate from its original function.

To ensure security in a system, in many situations, is a task that demands a precise capacity of analysis and, therefore, it must be considered the typical state in which the system will operate, once the adversary who attacks the system always attempt to manipulate the environment in his favor. Considering the facts, analysis based based exclusively in intuition should be avoided.

In this sense, to develop cryptographic systems, is relevant to consider possible strategies that the adversary would use to modify the system functionalities. Considering this strategies, it was developed an ideal model as an equivalence to the function that is desired to achieve [1].

The use of this model make reference to the utilization of an approach known as provable security. In cryptography, a system has provable security when its security requisits can be demonstrated under a adversarial model, as opposed to a heuristic model. The adversarial model assume that the adversary as access to the system, as to computational resources. The security proof consists in the validity of the systems requisits, considering that the assumptions about the adversary's access to the system are satisfieds and the suppositions about the hardness of some computational problems are valid.

## 1.1 Public-key Cryptography

The concept of public-key cryptography was proposed by Diffie and Hellman [2] in 1976. Also known as asymmetric encryption schemes, the encryption key can be known to any adversary without compromising the security of the scheme. Evidently, encryption and decryption keys are different. In addition, to compute the decryption key from the encryption key is infeasible. The highest level of security known to public key cryptosystems is indistinguishability against adaptive chosen ciphertext attack (IND-CCA2) [3], and developing a cryptosystem with this kind of feature can be a complex task. On the other hand, in the past years, several public key encryption (PKE) schemes have been proposed with either practical or theoretical purposes, where most of its security proofs relies on number theoretic assumptions or consider the applicability of certain one-way functions.

Currently there are three known paradigms for the elaboration of IND-CCA2 PKE schemes. The first paradigm was proposed by Dwork, Dolev and Naor [4], and is an enhancement of an construction proposed by Naor and Yung [5] (which only achieved the non-adptive IND-CCA). This scheme was proven IND-CCA2 and is based on computational assumptions of theoretical value and in *non-interactive zero knowledge* techniques. Later Sahai [6] and Lindell [7] made other improvements following the same approach. The second gives rise to IND-CCA2 practical schemes making use of hash-proof systems, and was presented by Cramer and Shoup [8]. Specifically, this was the first public key scheme to be proven IND-CCA2 without the use of random oracles. The last one requires the existance of *identity-based encryption* (IBE) schemes [9], and was first introduced by Canetti, Halevi and Katz [10].

In this sense, the final goal on improving IND-CCA2 PKE systems is the possibility of increasing the efficiency of these schemes and application of weaker computational assumptions. In addition, its brought to our attention the scarcity of systems based on *computational Diffie-Hellman* (CDH) assumption and *hashed Diffie-Hellman* (HDH) assumption. Considering these facts, we developed a IND-CCA2 PKE cryptosystem (with some restrictions that will be mentioned latter) based on the CDH assumption, seldom used in this kind of construction, and weaker than many other assumptions, including *decisional Diffie-Hellman* (DDH) assumption.

## 1.2 Our Contribution

We present a modification of the construction presented in Cramer et al [11], which its security is a weaker version of IND-CCA2, q-*bounded*-CCA2 security, technically termed IND-q-CCA2. This definition guarantees IND-CCA2 as long the number of the adversary's queries to the decryption oracle is bounded on a polynomial $q$ fixed in advance, in the key-generation.

The motivation for using this security notion can be explained in two different aspects. The first one is the fact that relying PKE schemes on simpler computational assumptions usually leads to schemes with weaker security[1]. This kind of construction allow us to build systems with a strong notion of security, such as IND-q-CCA2, adapting schemes with weaker computational assumptions, as long there is a bound for the number of adaptive chosen ciphertext queries. At the moment there are no standard IND-CCA2 schemes that relies on simpler notions related do discrete logarithm[2] with a optimal ciphertext lenght. The second aspect can be explained by the fact that IND-q-CCA2 systems allow us to achieve more efficient constructions.

We emphasize that our construction has a reduced ciphertext due to certain homomorphic key properties. Besides, our scheme makes no use of NIZK techniques.

[1] In public key encryption, security against *chosen plaintext attack*
[2] By simpler notions we mean weaker than DDH, such as CDH and HDH.

# Chapter 2

# Historical Background

The most classic challenge in cryptography consists on providing secret communication over insecure media. One method to assure secrecy in communication is the use of encryption schemes. An encryption scheme consists in a pair of algorithms, one applied by the sender, the *encryption*, while the other is applied by the receiver, the *decryption*.

The security analysis plays an essential role in the study of encryption schemes. In order to comprehend the meaning of *security*, we describe two notorious methodologies.

We start discussing about the information-theoretic methodology. Its purpose is to analyze the *information* about the plaintext that is present in the ciphertext. The downside of such a high level of security approach, is that the key must be as long as the message to be encrypted. This requisite is actually a serious restriction on the applicability of encryption schemes of this nature. It is an evident obstacle when a huge quantity of information needs to be secretly communicated.

The second methodology is concerned with computational complexity. Its primary aspect is that it does not matter whether or not the ciphertext contains information about the plaintext, but rather whether or not this information can be efficiently extracted. It turns out that this approach provides secure communication even if the key is much shorter than the message to be encrypted. A possibility with this kind of approach, is to use *pseudorandom generators* to expand short keys into longer *pseudo-keys*, so that the latter are as secure as *real keys* of comparable length.

*Computational-complexity* approach led to concepts and primitives that would not exist under information theoretic approach. One of the most important concepts that relies on *computationa-complexity* is *public-key encryption* schemes. This concept was proposed by Diffie and Hellman [2] in 1976. Also known as asymmetric encryption schemes, the encryption key can be known to any adversary without compromising the security of the scheme. Evidently, encryption and decryption keys are different. In addition, to compute the decryption key from the encryption key is infeasible.

In order to formalize security related to computational complexity, specially in public-key cryptography, researchers started to list necessary requisites for these schemes. Therefore the security proof of a protocol consisted on demonstrating that it met all the necessary features. Considering the context, it was developed an ideal model to serve as a prototype to the purpose to be achieved [1]. This model refers to an approach of *provable security*, which is the use of and adversarial algorithm to demonstrate the securities requisites.

### PUBLIC-KEY CRYPTOGRAPHY SECURITY MODELS

The first demostration of *provably secure public-key encryption* scheme was made by Rabin [12]. The Rabin cryptosystem is a PKE scheme, whose security is related to the difficulty of factorization. The great advantage of the Rabin cryptosystem is that the entire recovery of a random plaintext from the ciphertext is possible only if the adversary is capable of efficiently factor the public key. For current security standards, is a very weak level of security. This cryptosystem is provably secure against chosen plaintext attacks, however, its extensions achieve stronger notions of security.

Subsequently, Goldwasser and Micali proposed the first probabilistic public-key encryption scheme [13] which is provably secure under standard cryptographic assumptions. Still, this construction does not conduct to an efficient cryptosystem, as it produces a considerable ciphertext overhead. Because encryption is performed using a probabilistic algorithm, a given message produces different ciphertexts each time it is encrypted. This has significant advantages, as it prevents an adversary from recognizing intercepted messages by comparing them to a dictionary of known ciphertexts. For the security proof of properties of the cryptosystem, Goldwasser and Micali proposed the definition of semantic security.

For a PKE scheme achieve semantic security, it must be infeasible for a computationally-bounded adversary to derive significant information about a message when given only its ciphertext

and the corresponding public encryption key. Semantic security considers only the case of a "passive" attacker, i.e., one who generates and observes ciphertexts using the public key and plaintexts of her choice. Howerver, semantic security is now considered an insufficient condition for securing a general-purpose encryption scheme.

In early 1990s it began to be established reliable and easy to use formal models of the security of an encryption scheme and also began a concernment about constructing practical and efficient provably secure public-key encryption schemes. Naor and Yung [5] presented the first scheme provably secure against chosen ciphertext attacks (CCA). This construction uses probabilistic encryption schemes and non-interactive zero-knowledge proof systems, leading to a public key system secure against chosen ciphertext attacks. In CCA model, the adversary has access to an decryption oracle, and it chooses the ciphertext or ciphertexts to decrypt in advance, and does not use the resulting plaintexts to inform their choice for more ciphertexts.

The concept of security against an adaptive chosen ciphertext attack was expound by Rackoff and Simon [3]. The basic idea comes from the possibility that an adversary can inject messages into a network messages that may be encryptions. With this behavior, the adversary may be able to extract partial information about the wanted message through its interactions with the parties in the network. Rackoff and Simon's definition models this type of attack by simply allowing an adversary to obtain decryptions of its choice, i.e., the adversary is allowed access to a decryption oracle. Now, given an encryption of a message, it is desirable that the adversary cannot obtain any partial information about the message. To achieve this, the adversary's behavior is restricted in some way, otherwise the adversary could simply submit the target ciphertext itself to the decryption oracle. The restriction proposed by Rackoff and Simon is the weakest possible, the adversary is not allowed to submit the target ciphertext itself to the oracle. However, it may submit any other ciphertext, including ciphertexts that are related to the target ciphertext.

### RANDOM ORACLE METHODOLOGY

The random oracle methodology consists on modeling hash functions as an oracle that responds to every query with a random response chosen uniformly from its output domain, except that for any specific query, it responds the same way every time it receives that query. The possibility of this modeling is due to the fact that secure hash functions share many properties with random

functions.

Random oracles are a mathematical abstraction used in cryptographic proofs. They are typically used when no known implementable function provides the mathematical properties required by the proof. A system that is proven secure using such a proof is described as being secure in the random oracle model, as opposed to secure in the standard model. Such a proof generally shows that a system or a protocol is secure by showing that an attacker must require impossible behavior from the oracle, or solve some mathematical problem believed hard, in order to break the protocol.

Schemes proven secure using the random oracle methodology are not necessarily secure when the hash function is instantiated with a given fixed hash function. There is always the possibility that the particular hash function will interact badly with the mathematics of the encryption scheme, and that the resulting system will be insecure. However, it was hoped that the universe of hash functions that would interact in an undesirable way would be small, so it would be true that a scheme proven secure using the random oracle methodology would be secure when the random oracle was replaced with almost any hash function. But, Canetti, Goldreich and Halevi [14] proposed construction of an encryption scheme that was provably secure using the random oracle methodology, which turned out to be insecure when the random oracle was instantiated with any hash function. They make an observation that in the standard model the attacker has an extra piece of information not available to the attacker in the random oracle model, which is the description of the hash function. Although the encryption scheme of Canetti, Goldreich and Halevi is completely artificial, it does act as a proof of concept, i.e., it is possible to construct a scheme that is secure in the random oracle model, but insecure in the standard model. A lot of effort has been expended by cryptographers attempting to find a non-artiÞcial scheme which is secure in the random oracle model, but insecure in practice, but so far no such scheme has been found.

### IND-CCA2 PKE CONSTRUCTION PARADIGMS

We now turn our attention to schemes that can be proven in the standard model. Currently are known three paradigms for the construction of IND-CCA2 public-key encryption schemes in the standard model.

The first was proposed by Dolev, Dwork, and Naor [4], is a derivation of Naor-Yung construction [5], and uses its technique to construct a tag-based encryption scheme and then apply a CHK

transform [5]. The downside of this scheme is its highly inefficiency. The use of Naor-Yung "double- and-add" technique means that every message bit has to be encrypted multiple times, and use an arbitrary NIZK proof system. Furthermore, the use of the CHK transform implies the need for an inexpensive signing operation. Hence, this scheme can only be considered to be of theoretical interest.

The second paradigm was proposed by Cramer and Shoup [8], and it gives rise to practical public-key encryption construction that was proven secure in the standard model. This scheme is an extension of ElGamal encryption scheme [15], and was proven secure under the assumption that the DDH problem is hard and the existence of a target collision resistant hash function is true.

The third construction paradigm was proposed by Canetti, Halevi and Katz [10], and makes use of *identity-based cryptography* [9]. The conversion from IBE scheme to IND-CCA2 is provably secure in the standard model. The resulting construction is very simple and reasonably efficient, and does not make use of any *non-interactive* proofs.

Considering the historical background, it's easy to observe that the ultimate goal in *public-key cryptography* is the creation of simpler and more efficient encryption schemes, in addition, it should be provable-secure in a strong security model. It is also desirable that the new constructions are based on weak computational assumptions.

# Chapter 3

# Preliminaries

In this chapter we present some definitions which were used in the construction of our scheme. We refer the reader to [8, 11, 16, 17, 18, 19] for more detailed explanations of these definitions.

## 3.1 Notation

Throughout this paper it will be used the subsequent notations. We denote by $x \xleftarrow{\$} \mathcal{X}$ the experiment of choosing an element of $\mathcal{X}$ according to the uniform distribution over $\mathcal{X}$. If $\mathcal{A}$ is an algorithm, $x \leftarrow \mathcal{A}$ denotes that the output of $\mathcal{A}$ is $x$. In the case where $y$ is not an finite set nor an algorithm, $x \leftarrow y$ is an assignment operation. We establish $|l|$ as the bit length if $l$ is an element or an finite set. We write $w \leftarrow \mathcal{A}^{\mathcal{O}}(x, y, ...)$ for representing an algorithm $\mathcal{A}$ having access to an oracle $\mathcal{O}$. We denote by Pr $[E]$ the probability that the event $E$ occurs.

In the remainder of this section we make a brief review of notions used in our constructions such as public-key encryption, number theoretic assumptions, one-way functions, hard-core predicate, target collision resistant hash functions, strong pseudo-random permutation, cover-free families, Yao's XOR lemma and key encapsulation mechanism.

## 3.2 Public Key Encryption

A Public Key Encryption Scheme (PKE) is defined as follows:

**Definition 3.2.1** *A public-key encryption scheme is a triplet of algorithms* (Gen, Enc, Dec) *such*

*that:*

- $\mathsf{Gen}$ is a probabilistic polynomial-time ($p.p.t$) key generation algorithm which takes as input a security parameter $1^k$ and outputs a public key $\mathsf{pk}$ and a secret key $\mathsf{sk}$. The public key specifies the message space $\mathcal{M}$ and the ciphertext space $\mathcal{C}$.
- $\mathsf{Enc}$ is a (possibly) $p.p.t.$ encryption algorithm which receives as input a public key $\mathsf{pk}$ and a message $\mathsf{M} \in \mathcal{M}$, and outputs a ciphertext $\mathsf{C} \in \mathcal{C}$.
- $\mathsf{Dec}$ is a deterministic polynomial-time decryption algorithm which takes as input a secret key $\mathsf{sk}$ and a ciphertext $\mathsf{C}$, and outputs either a message $\mathsf{M} \in \mathcal{M}$ or an error symbol $\perp$.
- (Soundness) For any pair of public and private keys generated by $\mathsf{Gen}$ and any message $\mathsf{M} \in \mathcal{M}$ it holds that $\mathsf{Dec}(\mathsf{sk},\mathsf{Enc}(\mathsf{pk},\mathsf{M}))=\mathsf{M}$ with overwhelming probability over the randomness used by $\mathsf{Gen}$ and $\mathsf{Enc}$.

**Definition 3.2.2** *(Chosen Plaintext Attack) To a two-stage adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ against PKE we associate the following experiment $\mathsf{Exp}^{cpa}_{\mathcal{A},\mathrm{PKE}}(k)$:*

$(\mathsf{pk},\mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}(1^k)$

$(\mathsf{M}_0, \mathsf{M}_1, state) \leftarrow \mathcal{A}_1(\mathsf{pk})$ s.t. $|\mathsf{M}_0| = |\mathsf{M}_1|$

$\beta \xleftarrow{\$} \{0,1\}$

$\mathsf{C}^* \leftarrow \mathsf{Enc}(\mathsf{pk},\mathsf{M}_\beta)$

$\beta' \leftarrow \mathcal{A}_2(\mathsf{C}^*, state)$

$If\ \beta = \beta'$ return 1 else return 0

We define the advantage of $\mathcal{A}$ in the experiment as

$$\mathsf{Adv}^{cpa}_{\mathcal{A},\mathrm{PKE}}(k) = |Pr[\mathsf{Exp}^{cpa}_{\mathcal{A},\mathrm{PKE}}(k) = 1] - \frac{1}{2}|$$

We say that PKE is *indistinguishable against chosen-plaintext attack* (IND-CPA) if for all $p.p.t.$ adversaries $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ the advantage of $\mathcal{A}$ in the experiment is a negligible function of k.

**Definition 3.2.3** *(Adaptive Chosen Ciphertext Attacks) To a two-stage adversary* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ *against PKE we associate the following experiment* $\mathsf{Exp}^{cca2}_{\mathcal{A},\mathrm{PKE},}(k)$*:*

$(\mathsf{pk},\mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}(1^k)$
$(\mathsf{M}_0, \mathsf{M}_1, state) \leftarrow \mathcal{A}_1^{\mathsf{Dec}(\mathsf{sk},.)}(\mathsf{pk})$ s.t. $|\mathsf{M}_0| = |\mathsf{M}_1|$
$\beta \xleftarrow{\$} \{0,1\}$
$\mathsf{C}^* \leftarrow \mathsf{Enc}(\mathsf{pk},\mathsf{M}_\beta)$
$\beta' \leftarrow \mathcal{A}_2^{\mathsf{Dec}(\mathsf{sk},.)}(\mathsf{C}^*, state)$
$If\ \beta = \beta'$ return 1 else return 0

The adversary $\mathcal{A}_2$ is not allowed to query $\mathsf{Dec}(\mathsf{sk},.)$ with $\mathsf{C}^*$. We define the advantage of $\mathcal{A}$ in the experiment as

$$\mathsf{Adv}^{cca2}_{\mathcal{A},\mathrm{PKE}}(k) = |Pr[\mathsf{Exp}^{cca2}_{\mathcal{A},\mathrm{PKE}}(k) = 1] - \frac{1}{2}|$$

We say that PKE is *indistinguishable against adaptive chosen-ciphertext attack* (IND-CCA2) if for all *p.p.t.* adversaries $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ that makes a polynomial number of oracle queries the advantage of $\mathcal{A}$ in the experiment is a negligible function of k.

**Definition 3.2.4** *(q-Bounded Chosen Ciphertext Attacks) For a function* $q(k) : \mathbb{N} \to \mathbb{N}$ *and a two stage adversary* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$*, against PKE we associate the following experiment* $\mathsf{Exp}^{ind-q-cca2}_{\mathcal{A},\mathrm{PKE}}(k)$*:*

$(\mathsf{pk},\mathsf{sk}) \xleftarrow{\$} \mathsf{Gen}(1^k)$
$(\mathsf{M}_0, \mathsf{M}_1, state) \leftarrow \mathcal{A}_1^{\mathsf{Dec}(\mathsf{sk},.)}(\mathsf{pk})$ s.t. $|\mathsf{M}_0| = |\mathsf{M}_1|$
$\beta \xleftarrow{\$} \{0,1\}$
$\mathsf{C}^* \leftarrow \mathsf{Enc}(\mathsf{pk},\mathsf{M}_\beta)$
$\beta' \leftarrow \mathcal{A}_2^{\mathsf{Dec}(\mathsf{sk},.)}(\mathsf{C}^*, state)$
$If\ \beta = \beta'$ return 1 else return 0

The adversary $\mathcal{A}$ is allowed to ask at most $q(k)$ queries to the decryption oracle $\mathsf{Dec}$ in each run of the experiment. As in the IND-CCA2 game, none of the queries of $\mathcal{A}_2$ may contain $\mathsf{C}^*$. We define

the advantage of $\mathcal{A}$ in the experiment as

$$\mathsf{Adv}^{ind-q-cca2}_{\mathcal{A},\mathrm{PKE}}(k) = |Pr[\mathsf{Exp}^{ind-q-cca2}_{\mathcal{A},\mathrm{PKE}}(k) = 1] - \frac{1}{2}|$$

We say that PKE is *indistinguishable against q-bounded adaptive chosen-ciphertext attack* (IND-q-CCA2) if for all *p.p.t.* adversaries A=($\mathrm{A}_1, \mathcal{A}_2$) that makes a polynomial number of oracle queries the advantage of $\mathcal{A}$ in the experiment is a negligible function of k.

## 3.3 Number Theoretic Assumptions

In this section we state three of the Diffie-Hellman intractability assumptions: *Computational Diffie-Helman, Decisional Diffie-Hellman* and *Hashed Diffie-Hellman.*

**Definition 3.3.1** *(CDH assumption) Let $\mathbb{G}$ be a group of order p and generator g. For all probabilistic polynomial time adversaries $\mathcal{A}$, we define the CDH advantage of $\mathcal{A}$ against $\mathbb{G}$ at a security paramerer k as*

$$\mathbf{Adv}^{cdh}_{\mathcal{A},\mathbb{G}}(k) = \Pr[c = g^{xy} : x, y \xleftarrow{\$} \mathbb{Z}_p; c \leftarrow \mathcal{A}(1^k, g^x, g^y)]$$

The CDH assumption states that for every polynomial-time adversary $\mathcal{A}$ the function $\mathbf{Adv}^{cdh}_{\mathcal{A},\mathbb{G}}$ is negligible in $k$.

**Definition 3.3.2** *(DDH assumption) Let $\mathbb{G}$ be a group of order p and generator g. We define the sets $\mathcal{D}_k$ and $\mathcal{T}_k$ for a security parameter k as follows:*

$\mathcal{D}_k := \{g^x, g^y, g^{xy}$: x,y $\in \mathbb{Z}_p$, $x \neq 0$ $\}$;
$\mathcal{T}_k := \{g^x, g^y, g^z$: x,y,z $\in \mathbb{Z}_p$, $x \neq 0$, $z \neq xy$ $\}$.

The set $\mathcal{D}_k$ is the set of *Diffie-Hellman triples* and $\mathcal{T}_k$ is the set of triples $\in \mathbb{G}^3$ different from DH triples. For $\rho \in \mathbb{G}^3$ and $\mathcal{A}$ a 0/1-valued *p.p.t.* adversarial algorithm, let $\zeta$ be $\mathcal{A}$'s guess about the triple $\rho$. For $\zeta$=1, $\mathcal{A}$ guesses that $\rho \in \mathcal{D}_k$, else $\mathcal{A}$ guesses that $\rho \in \mathcal{T}_k$. We define the DDH

advantage of $\mathcal{A}$ against $\mathbb{G}$ at a security paramerer $k$ as

$$\mathbf{Adv}^{ddh}_{\mathcal{A},\mathbb{G}}(k) = |\Pr[\zeta = 1 : \rho \xleftarrow{\$} \mathcal{D}_k; \zeta \leftarrow \mathcal{A}(1^k, \rho)] - \Pr[\zeta = 1 : \rho \xleftarrow{\$} \mathcal{T}_k; \zeta \leftarrow \mathcal{A}(1^k, \rho)]|$$

The DDH assumption states that for every polynomial-time adversary $\mathcal{A}$ the function $\mathbf{Adv}^{ddh}_{\mathcal{A},\mathbb{G}}$ is negligible in $k$.

**Definition 3.3.3** *(Hashed-DH Assumption) Let $\mathbb{G}$ be a group of order $p$ and generator $g$. Let $H$: $\{0,1\}^k \times \mathbb{G} \rightarrow \{0,1\}^n$ be a family of one-way hash functions. We define the sets $\mathcal{D}_k$ and $\mathcal{T}_k$ for a security parameter $k$ as follows:*

$\mathcal{D}_k := \{g^x, g^y, H(g^{xy})$: x,y $\in \mathbb{Z}_p$, $x \neq 0$ $\}$;
$\mathcal{T}_k := \{g^x, g^y, r \in \{0,1\}^n$: x,y $\in \mathbb{Z}_p$, $x \neq 0$, $r \neq H(g^{xy})$ $\}$.

In this weakness of DDH assumption, the set $\mathcal{D}_k$ is the set with respect of values of *Diffie-Hellman triples*. Correspondingly to DDH game definition, $\mathcal{T}_k$ is a set with a random element. For $\rho \in \mathbb{G}^3$ and $\mathcal{A}$ a 0/1-valued *p.p.t.* adversarial algorithm, let $\zeta$ be $\mathcal{A}$'s guess about the triple $\rho$. For $\zeta$=1, $\mathcal{A}$ guesses that $\rho \in \mathcal{D}_k$, else $\mathcal{A}$ guesses that $\rho \in \mathcal{T}_k$. We define the HDH advantage of $\mathcal{A}$ against $\mathbb{G}$ at a security paramerer $k$ as

$$\mathbf{Adv}^{hdh}_{\mathcal{A},\mathbb{G}}(k) = |\Pr[\zeta = 1 : \rho \xleftarrow{\$} \mathcal{D}_k; \zeta \leftarrow \mathcal{A}(1^k, \rho)] - \Pr[\zeta = 1 : \rho \xleftarrow{\$} \mathcal{T}_k; \zeta \leftarrow \mathcal{A}(1^k, \rho)]|$$

The HDH assumption states that for every polynomial-time adversary $\mathcal{A}$ the function $\mathbf{Adv}^{hdh}_{\mathcal{A},\mathbb{G}}$ is negligible in $k$.

Throughout this paper we will denote $\epsilon_{cdh} = \mathbf{Adv}^{cdh}_{\mathcal{A},\mathbb{G}}(k)$, $\epsilon_{ddh} = \mathbf{Adv}^{ddh}_{\mathcal{A},\mathbb{G}}(k)$, $\epsilon_{hdh} = \mathbf{Adv}^{hdh}_{\mathcal{A},\mathbb{G}}(k)$.

## 3.4 One-Way Functions

A collection of efficiently computable functions is defined as a pair of *p.p.t.* algorithms $\mathcal{F} = (\mathsf{F}, \mathsf{G})$. The algorithm $\mathsf{G}$ on input $1^k$ outputs a description $s \in \{0,1\}^k$ of a function $f_s$: $\{0,1\}^k$ $\{0,1\}^k$. The algorithm $\mathsf{F}$ on input $(s,x) \in \{0,1\}^k \times \{0,1\}^k$ outputs $f_s(x)$. Let $\mathsf{F}^{-1}$=$\{x \in \{0,1\}^k | y = \mathsf{F}(s,x)\}$, and $\mathcal{A}$ be an adversarial algorithm. Then, consider the following experiment:

$$\mathbf{Exp}^{owf}_{\mathcal{A},\pi}(k) : [x \xleftarrow{\$} \{0,1\}^k; s \xleftarrow{\$} \mathsf{G}(1^k); y \leftarrow \mathcal{A}(1^k, s, \mathsf{F}(s,x)); \text{return 1 if } y = \mathsf{F}^{-1}(s,x), \text{ else return 0}].$$

We define

$$\epsilon_{owf} = Pr[\mathbf{Exp}^{owf}_{\mathcal{A},\pi}(k) = 1].$$

**Definition 3.4.1** *(One-Way Function) A collection of* efficient computable functions $\mathcal{F}$ *is said to be* one-way *if for every* p.p.t. $\mathcal{A}$ *it holds that* $\epsilon_{owf}$ *is negligible.*

## 3.5 Hard-Core Predicate

Let $\mathcal{F} = (\mathsf{F}, \mathsf{G})$ be a collection of efficiently computable functions. A polynomial-time algorithm h: $\{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ is a hard-core predicate. Let $\mathcal{A}$ be an adversarial algorithm. Then, consider the following experiment:

$$\mathbf{Exp}^{hcb}_{\mathcal{A},\pi}(k) : [x \xleftarrow{\$} \{0,1\}^k; s \xleftarrow{\$} \mathsf{G}(1^k); y \leftarrow \mathcal{A}(1^k, s, \mathsf{F}(s,x)); \text{return 1 if } y = h(s,x), \text{ else return 0}].$$

We define

$$\epsilon_{hcb} = |Pr[\mathbf{Exp}^{hcb}_{\mathcal{A},\pi}(k) = 1] - \frac{1}{2}|.$$

**Definition 3.5.1** *(Hard-core Predicate) A polynomial-time algorithm h:* $\{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ *is said to be a hard-core predicate if for every* p.p.t. $\mathcal{A}$ *it holds that* $\epsilon_{hcb}$ *is negligible.*

## 3.6 Target Collision Resistant Hash Functions

*Target collision resistant hash function* is a special case of *universal one-way hash function.* Let $\mathbb{G}$ be a group, and $H$: $\{0,1\}^k \times \mathbb{G} \to \{0,1\}^n$ be a family of functions,where $k$ is the security parameter. Let TCR: $\{0,1\}^k \times \mathbb{G} \to \{0,1\}^n$ be an instance of $H$, which is indexed by $k \in \{0,1\}^k$, and $\mathcal{A}$ be an adversary. Then, consider the following experiment:

$$\mathbf{Exp}^{tcr}_{\mathcal{A},\pi}(k) : [k \xleftarrow{\$} \{0,1\}^k; x \xleftarrow{\$} \{0,1\}^l; x' \leftarrow \mathcal{A}(k,x); \text{return 1 if } TCR(x') = TCR(x), \text{else return 0}].$$

We define

$$\epsilon_{tcr} = Pr[\mathbf{Exp}^{tcr}_{\mathcal{A},\pi}(k) = 1].$$

**Definition 3.6.1** *(Target Collision Resistant Hash Function) A polynomial-time algorithm TCR:*$\{0,1\}^k \times \mathbb{G} \to \{0,1\}^n$ *is said to be a target collision resistant hash function if for every* p.p.t. $\mathcal{A}$ *it holds*

*that $\epsilon_{tcr}$ is negligible.*

## 3.7 Strong Pseudo-Random Permutation

Let $\pi$: $\{0,1\}^k \times \{0,1\}^l \to \{0,1\}^l$ be a family of permutations, and $\pi_k$: $\{0,1\}^l \to \{0,1\}^l$ be an instance of $\pi$, which is indexed by $k \in \{0,1\}^k$. Let $\mathcal{P}$ be the set of all permutations for $l$-bit strings, and $\mathcal{A}$ be an adversary. Then, consider the following experiments:

$$\mathbf{Exp}^{sprp}_{\mathcal{A},\pi}(k) : [k \xleftarrow{\$} \{0,1\}^k; \beta \leftarrow \mathcal{A}^{\pi_K, \pi_k^{-1}}; return\, \beta],$$

$$\mathbf{Exp}^{real}_{\mathcal{A},\pi}(k) : [perm \xleftarrow{\$} \mathcal{P}; \beta \leftarrow \mathcal{A}^{perm, perm^{-1}}; return\, \beta],$$

where permutations $\pi_K, \pi_k^{-1}, perm, perm^{-1}$ are given to $\mathcal{A}$ as black boxes, and $\mathcal{A}$ can observe only their outputs which correspond to $\mathcal{A}$'s inputs. We define

$$\epsilon_{sprp} = \frac{1}{2}|\Pr[\ \mathbf{Exp}^{sprp}_{\mathcal{A},\pi}(k) = 1] - \Pr[\ \mathbf{Exp}^{real}_{\mathcal{A},\pi}(k) = 1]|$$

**Definition 3.7.1** *(Strong Pseudorandom Permutation - SPRP) A polynomial-time algorithm $\pi_k$: $\{0,1\}^l \to \{0,1\}^l$ is said to be a strong pseudorandom permutation if for every* p.p.t. *$\mathcal{A}$ it holds that $\epsilon_{sprp}$ is negligible.*

## 3.8 Cover Free Families

If S, T are sets, we say that S does not cover T if S $\not\supseteq$ T . Let d, q, s be positive integers, and let $\mathcal{F} = (\mathcal{F}_i)_{1 \le i \le s}$ be a family of subsets of {1, . . . , d}. We say that family $\mathcal{F}$ is q-cover-free over {1, . . . , d}, if for each subset $\mathcal{F}_i \in \mathcal{F}$ and each S that is the union of at most q sets in ($\mathcal{F}_1$ , . . . , $\mathcal{F}_{i-1}$ , $\mathcal{F}_{i+1}$ , . . . , $\mathcal{F}_s$ ), it is the case that S does not cover $\mathcal{F}_i$ . Furthermore, we say that $\mathcal{F}$ is $l$-uniform if all subsets in the family have size $l$. We use the following fact: there is a deterministic polynomial time algorithm that on input integers s, q returns l, d, $\mathcal{F}$ where $\mathcal{F} = (\mathcal{F}_i)_{1 \le i \le s}$ is a l-uniform q-cover-free family over {1, . . . , d}, for l = $\frac{d}{4q}$ and d$\le 16q^2$log(s). In the following we let SUB denote the resulting deterministic polynomial-time algorithm that on input s, q, it returns $\mathcal{F}_i$ . We call $\mathcal{F}_i$ = SUB(s(k), q(k), i) the subset associated to index i $\in$ {1, . . . , s(k)}. For our

construction we will need a cover-free family with the parameters

$$s(k) = 2k, d(k) = 16kq^2(k), l(k) = 4kq(k).$$

## 3.9 Yao's XOR Lemma

An important Lemma of Yao [20] states that computational hardness of inverting one-way functions gets amplified if the results of several independent instances are XOR together. By hard to invert we mean that any efficient algorithm will fail to predict the predicate with probability beyond a stated bound, where the probability is taken over all possible inputs.

Let $f$: $\{0,1\}^n \rightarrow \{0,1\}^n$ be a one-way function and let $h$: $\{0,1\}^n \rightarrow \{0,1\}$ be a hard-core predicate of $f$.

In particular, Yao's XOR Lemma asserts that if the predicate $h$ is weakly-unpredictable then $\overline{h}(x_1, \ldots, x_t) := \bigoplus_{i=1}^{t} h(x_i)$ for sufficiently large t is almost unpredictable within a related complexity bound. This results can be expanded for the case where there is a hard-core bit among $h(x_i)$. The Lemma guarantees that the unpredictability of the hard-core bit is transfered to the result of the XOR operation $\bigoplus_{i=1}^{t} h(x_i)$.

**Lemma 3.9.1** (Yao's XOR Lemma)*: Suppose $\mathcal{A}_2$ is a probabilistic adversary satisfying*

$$Pr[\mathcal{A}_2(x_1, x_2) = \overline{h}(x_1, x_2) | x_1, x_2 \leftarrow \{0,1\}^n] > \frac{1}{2} + \epsilon^2$$

*Then there is a probabilistic adversary $\mathcal{A}$ whose running time is polynomial in $\mathcal{A}_2$'s running time and satisfies:*

$$Pr[\mathcal{A}(g(x_2)) = h(x_2) | x_2 \leftarrow \{0,1\}^n] > \frac{1}{2} + \epsilon$$

## 3.10 Hybrid Encryption

Our models make use of a method of hybrid encryption [21]. Such schemes uses public-key encryption techniques to encrypt a random key $\mathsf{K}$. The encrypted key $\overline{\mathsf{K}}$ is then used to encrypt a actual message using a symmetric encryption scheme.

The structure of key encapsulation mechanism is used to generate the symmetric key used in

the message encryption. This mechanism is homologous to public key encryption scheme, but instead of encrypting a message, the encryption algorithm generates the encryption $\overline{\mathsf{K}}$ of a random key $\mathsf{K}$.

### 3.10.1 Key Encapsulation Mechanism

A Key Encapsulation Mechanism (KEM) is defined as follows:

**Definition 3.10.1** *A key encapsulation mechanism is a triplet of algorithms* (KGen, KEnc, KDec) *such that:*

- KGen is a probabilistic polynomial-time ($p.p.t$) key generation algorithm which takes as input a security parameter $1^k$ and outputs a public key pk and a secret key sk. The public key specifies the key space $\mathcal{K}$ and the symmetric key space $\overline{\mathcal{K}}$.
- KEnc is a (possibly) $p.p.t.$ encryption algorithm which receives as input a public key pk, and outputs ($\mathsf{K}$ ,$\overline{\mathsf{K}}$), where $\mathsf{K} \in \mathcal{K}$ is a key, and $\overline{\mathsf{K}} \in \overline{\mathcal{K}}$ is a encapsulated symmetric key.
- KDec is a deterministic polynomial-time decryption algorithm which takes as input a secret key sk and a key K, and outputs a encapsulated symmetric key $\overline{\mathsf{K}} \in \overline{\mathcal{K}}$ or an error symbol $\perp$.
- (Soundness) For any pair of public and private keys generated by KGen and any pair ($\mathsf{K}$ ,$\overline{\mathsf{K}}$) generated by KEnc it holds that KDec(sk,K)=$\overline{\mathsf{K}}$ with overwhelming probability over the randomness used by KGen and KEnc.

**Definition 3.10.2** *(Key Encapsulation Mechanism Adaptive Chosen Ciphertext Security) To an attack algorithm $\mathcal{A}$ and against KEM we associate the following experiment* $\mathbf{Exp}^{kem}_{\mathcal{A},\mathrm{PKE}}(\mathrm{k})$*:*

$(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{KGen}(1^k)$

$(\mathsf{K}^*,\overline{\mathsf{K}}^*) \leftarrow \mathsf{KEnc}(\mathsf{pk})$

$\beta \xleftarrow{\$} \{0,1\}$

If $\beta = 0$, $\overline{\mathsf{K}}^{\diamond} \leftarrow \overline{\mathsf{K}}^*$, else $\overline{\mathsf{K}}^{\diamond}$ is random

$\beta' \leftarrow \mathcal{A}^{\mathsf{KDec}(\mathsf{sk},\cdot)}(\mathsf{K}^*,\overline{\mathsf{K}}^{\diamond})$

If $\beta' = \beta$ return 1 else return 0

The adversary $\mathcal{A}$ is not allowed to query $\mathsf{KDec}(\mathsf{sk},.)$ with $\overline{\mathsf{K}}^{\diamond}$. We define the advantage of $\mathcal{A}$ in the experiment as

$$\mathbf{Adv}^{kem}_{\mathcal{A},\mathrm{PKE}}(\mathrm{k}) = |\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\mathrm{PKE}}(\mathrm{k}) = 1] - \frac{1}{2}|$$

We say a KEM used in a PKE is *indistinguishable against adaptive chosen-ciphertext attack* (IND-CCA2) if for all *p.p.t.* adversaries $\mathcal{A}$ the advantage of $\mathcal{A}$ in the experiment is a negligible function of k. Throughout this paper, we will denote $\mathbf{Adv}^{kem}_{\mathcal{A},\mathrm{PKE}}(\mathrm{k})$ as $\epsilon_{kem}$.

# Chapter 4

# Our Model

## 4.1 IND-q-CCA2 Encryption

Our constructions will be addressed in this chapter. It will be presented the main ideas of our work and, subsequently, the description the schemes.

### 4.1.1 IND-q-CCA2 Encryption From CDH

Our construction yields a IND-q-CCA PKE scheme based on CDH assumption with optimal ciphertext length. To achieve a scheme with such features, we make use of hybrid encryption techniques. The property of key homomorphism also plays an important role in our construction, since it makes possible component ciphertexts be compressed to be one. The symmetric-key encryption scheme is constructed based on strong pseudorandom permutations, as in [19], to obtain redundancy-free property and security against chosen-ciphertext attacks.

Furthermore, we use the randomness established in the encryption phase and a TCR to define the value $t$ that will designate the q-cover-free family (q-CFF) subset of the session. The session's q-CFF subset and a hardcore function will be used to construct the symmetric key.

It can be assured, due to the property of cover-free families and the unduplicatable set selection, that at least one element of the decryption key set will remain secret, since it won't be required for responding decryption queries, if the adversary submits at most $q$ queries.

CONSTRUCTION

We assume the existence of a cyclic group $\mathbb{G}$ of prime-order $p$ where the CDH assumption is believed to hold, i.e., given $(g, g^a, g^b)$ there is no efficient way to calculate $g^{ab}$, for random $g \in \mathbb{G}$, and random $a$, $b \in \mathbb{Z}_p$. Let $H$: $\{0,1\}^k \times \mathbb{G} \rightarrow \{1, \ldots, s\}$ be a function family where the index space is $\{0,1\}^k$, $\pi$:$\{0,1\}^k \times \{0,1\} \rightarrow \{0,1\}$ be a permutation family where the index space is $\{0,1\}^k$, and $h$ : $\{0,1\}^k \times \mathbb{G} \rightarrow \{0,1\}$ be a hardcore function family where the index space is $\{0,1\}^k$. Our scheme from CDH assumption $\Pi'$ consists of the following algorithms:

Gen $(1^k)$: Define s(k) = $2^k$, d(k) = 16k$q^2$(k), l(k)=4kq(k). Run KGen. For i = 1, ..., d(k), KGen computes $X_i$=$g^{x_i}$ for $x_i \xleftarrow{\$} \mathbb{Z}_p$, and outputs pk=$(X_1, \ldots, X_{d(k)})$ and sk=$(x_1, \ldots, x_{d(k)})$. The public key is pk, and the secret key is sk.

Enc (pk, M): Run KEnc. KEnc computes r = $g^y$ for $y \xleftarrow{\$} \mathbb{Z}_p$, $j$ = TCR(r) where $\mathcal{F}_j = \{j_1, \ldots, j_l\}$ is the q-CFF subset associated to the value $j$ (which will define the set of the session's publlic/private keys), sets K = $r$ and calculates $\overline{\mathsf{K}} = (\mathrm{h}(X_{j_1}{}^y) \oplus \ldots \oplus h(X_{j_l}{}^y))$. To encrypt message M, run symmetric-key encryption to obtain the ciphertext $\psi \leftarrow \pi_{\overline{\mathsf{K}}}(\mathsf{M})$. Output C=(r, $\psi$).

Dec (sk, C): Run KDec. KDec computes j = TCR(r) to obtain the subset $\mathcal{F}_j$, and calculates the session's symmetric key $\overline{\mathsf{K}} = (h(r^{x_{j_1}}) \oplus \ldots \oplus h(r^{x_{j_l}}))$. Decrypt $\psi$ to $\mathsf{M} \leftarrow \pi^{-1}_{\overline{\mathsf{K}}}(\psi)$.

AN IND-q-CCA2 PKE SCHEME BASED ON CDH.

**Theorem 1** *The above scheme is IND-q-CCA2 if the CDH assumption holds, TCR is a target collision resistant hash function, h is a hardcore predicate, and $\pi$ is strongly pseudorandom.*

*Proof:* To prove the above theorem we follow the same approach of [19]. The proof of the above theorem is made in two parts: In the first part we prove that $\Pi'$ is secure if $\overline{\mathsf{K}}$ is totally unknown to $\mathcal{A}$ and $\pi$ is a SPRP. In the second part of the proof, using the CDH assumption, we prove that the adversary cannot distinguish the session key from a random bit.

By showing $\mathcal{A}$'s inability of extracting partial knowledge of $\overline{\mathsf{K}}$, we demonstrate the security of the session key.

**Lemma 1** *The public key scheme $\Pi'$ is IND-q-CCA2 if $\pi$ is a* strong pseudo random permutation.

*Proof:* It's constructed a simulator $\mathcal{B}$ with the capability to distinguish a real session key from a random key, or to distinguish an instance $\pi_{\overline{\mathsf{K}}^*}$ of the permutation family $\pi$ from a random permutation *perm*. In the construction of $\mathcal{B}$ it's used a subroutine $\mathcal{A}$, which breaks the security of the scheme $\Pi'$.

Preliminary to the simulation, $\mathcal{B}$ flips a coin $\mathcal{COIN} \in \{0,1\}$. If $\mathcal{COIN}$=0, $\mathcal{B}$ tries to break the security of the session key. Otherwise, $\mathcal{B}$ tries to distinguish $\pi$ from a random permutation *perm*.

If $\mathcal{COIN}$=0 the simulation occurs as follows. The simulator $\mathcal{B}$, interacting with $\mathcal{A}$, will try to break the security of the session's key in a KEM CCA2 game. As $\mathcal{B}$ receives the public key $\mathsf{pk}$ of $\Pi'$, he passes it to $\mathcal{A}$. As the game proceeds, $\mathcal{B}$ receives a challenge ($\mathsf{C}^*$, $\overline{\mathsf{K}}^*$). Subroutine $\mathcal{A}$ is allowed to decryption queries, as long it respect the bound $q$ of queries. For a decryption request $\mathsf{C} = (r\ (\neq r^*), \psi)$ from $\mathcal{A}$ , $\mathcal{B}$ answers the queries asking the symmetric key $\overline{\mathsf{K}}$ related to the value $r$ to his own oracle. $\mathcal{B}$ sends $\mathrm{M} \leftarrow \pi_{\overline{\mathsf{K}}^{-1}}(\psi)$ to $\mathcal{A}$. In addition, for $\mathcal{A}$'s decryption query $\mathsf{C}$=($r^*$, $\psi$), $\mathcal{B}$ returns $\mathrm{M} \leftarrow \pi^{-1}_{\overline{\mathsf{K}}^*}(\psi)$.

In the challenge phase, $\mathcal{A}$ submits $M_0$ and $M_1$ to $\mathcal{B}$. $\mathcal{B}$ chooses $\beta$ randomly from $\{0,1\}$ and encrypts $M_\beta$ as $\psi^* \leftarrow \pi_{\overline{\mathsf{K}}^*}(M_\beta)$. $\mathcal{B}$ sends the challenge $(r^*, \psi^*)$ to $\mathcal{A}$. $\mathcal{A}$ is allowed for more decryption queries (taking into consideration the bounded number of decryption queries).

In the final stage, $\mathcal{A}$ outputs its guess $\beta'$. If $\beta' = \beta$, $\mathcal{B}$ outputs 1 in his own challenge. Else, $\mathcal{B}$ outputs 0.

If $\mathcal{COIN} = 1$, simulator $\mathcal{B}$ will try to distinguish $\pi_{\overline{\mathsf{K}}^*}$ from a random permutation *perm*. $\mathcal{B}$ is allowed to access an oracle that answers with a instance that can be either $\pi_{\overline{\mathsf{K}}^*}$ or *perm*.

The simulation proceeds as follows. $\mathcal{B}$ generates key pair ($\mathsf{pk}$, $\mathsf{sk}$) of $\Pi'$, passing $\mathsf{pk}$ to $\mathcal{A}$. $\mathcal{B}$ randomly chooses a value $r^*$ from $\mathbb{G}$. For a decryption request $\mathsf{C} = (r\ (\neq r*), \psi)$ from $\mathcal{A}$ , $\mathcal{B}$ answers the queries by decrypting $\psi$ with $\mathsf{sk}$. Also, for a decryption request $\mathsf{C} = (r^*, \psi)$ from $\mathcal{A}$ , $\mathcal{B}$ answers the queries sending $\psi$ to his own oracle, which is $\pi^{-1}_{\overline{\mathsf{K}}^*}$ or $perm^{-1}$. $\mathcal{B}$ returns the oracle's answer to $\mathcal{A}$. When $\mathcal{A}$ submits $\mathsf{M}_0$ and $\mathsf{M}_1$ to $\mathcal{B}$, $\mathcal{B}$ chooses $\beta$ randomly from $\{0,1\}$, and sends $\mathsf{M}_\beta$ to his own oracle, which is $\pi_{\overline{\mathsf{K}}^*}$ or *perm*. $\mathcal{B}$ takes the oracle's answer $\psi^*$, and sends $\mathsf{C}^*$=($r^*, \psi^*$) as the challenge to $\mathcal{A}$.

In the final stage, $\mathcal{A}$ outputs its guess $\beta'$. If $\beta' = \beta$, $\mathcal{B}$ outputs 1 in his own challenge. Else, $\mathcal{B}$ outputs 0.

For our simulator $\mathcal{B}$, the views for the events [$\overline{\mathsf{K}}^*$ is random $\wedge$ $\mathcal{COIN} = 0$] and [given permu-

tation is $\pi_{\overline{\mathsf{K}}^*} \wedge \mathcal{COIN} = 1]$ are identical. Let

$$Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|\overline{\mathsf{K}}^* \; is \; random \wedge \mathcal{COIN} = 0] = Pr[\mathbf{Exp}^{sprp}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 1] = \frac{1}{2} + \lambda \quad (4.1)$$

The probability that $\mathcal{B}$ breaks the KEM experiment is given by:

$$Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1] = Pr[\mathcal{COIN} = 1].Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 1] +$$

$$Pr[\mathcal{COIN} = 0].Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 0]$$

$$= \frac{1}{2}.Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 0] + \frac{1}{2}.\frac{1}{2}$$

$$= \frac{1}{2}.Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 0] + \frac{1}{4} \quad (4.2)$$

Let's denote the advantage of adversary $\mathcal{A}$ by $\epsilon_{cca}$. For the probabilities:

$$Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 0] =$$

$$\frac{1}{2}.Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|K^* = K(c^*) \wedge \mathcal{COIN} = 0]$$

$$+ \frac{1}{2}.Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|K^* \; is \; random \wedge \mathcal{COIN} = 0]$$

From (4.2):

$$Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 0] \leq \frac{1}{2}.(\frac{1}{2} + \epsilon_{cca}) + \frac{1}{2}.(\frac{1}{2} - \lambda) \quad (4.3)$$

Therefore,

$$Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1] \leq \frac{1}{2} + \frac{1}{4}(\epsilon_{cca} - \lambda) \quad (4.4)$$

The probability that $\mathcal{B}$ distinguishes the permutation is given similarly to the event above:

$$Pr[\mathbf{Exp}^{sprp}_{\mathcal{B},\Pi'}(k) = 1] = Pr[\mathcal{COIN} = 0].Pr[\mathbf{Exp}^{sprp}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 0] +$$

$$Pr[\mathcal{COIN} = 1].Pr[\mathbf{Exp}^{sprp}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 1]$$

$$= \frac{1}{2}.Pr[\mathbf{Exp}^{sprp}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 1] + \frac{1}{2}.\frac{1}{2}$$

$$= \frac{1}{2}.Pr[\mathbf{Exp}^{sprp}_{\mathcal{B},\Pi'}(k) = 1|\mathcal{COIN} = 1] + \frac{1}{4} \tag{4.5}$$

And just like above, from (4.2):

$$Pr[\mathbf{Exp}^{sprp}_{\mathcal{B},\Pi'}(k) = 1] = \frac{1}{2}.(\frac{1}{2} + \lambda) + \frac{1}{4} = \frac{1}{2} + \frac{1}{2}.\lambda \tag{4.6}$$

The advantage of the adversary $\mathcal{B}$ distinguishing a *SPRP* from a random permutation, $\epsilon_{sprp,\mathcal{B}}$, can be defined as:

$$\epsilon_{sprp,\mathcal{B}} = \frac{1}{2}.|\Pr[\mathbf{Exp}^{sprp}_{\mathcal{B},\Pi'}(k) = 1] - [\Pr\mathbf{Exp}^{rnd}_{\mathcal{B},\Pi'}(k) = 1]|$$

Since $[\Pr\mathbf{Exp}^{rnd}_{\mathcal{B},\Pi'}(k) = 1] = \frac{1}{2}$, and from (4.1):

$$\epsilon_{sprp,\mathcal{B}} = \frac{1}{2}|\frac{1}{2} + \frac{1}{2}.\lambda - \frac{1}{2}| = \frac{1}{4}|\lambda|$$

From the assumptions, we have:

$$\epsilon_{kem} \leq \frac{1}{4}(\epsilon_{cca} - \lambda)$$

$$\epsilon_{sprp} \leq \frac{1}{4}|\lambda|$$

Hence:

$$\epsilon_{cca} \leq 4\epsilon_{kem} + \lambda \leq 4\epsilon_{kem} + 4\epsilon_{sprp}.$$

□

**Lemma 2** *The adversary cannot distinguish the session key from a random bit if the CDH assumption holds.*

*Proof:* Now we will show that if the CDH assumption holds, the adversary cannot distinguish a real session key from a random bit. To prove the indistinguishability of the key, we construct a adversary $\mathcal{B}$ that breaks the CDH assumption.

To achieve this goal, the adversary $\mathcal{B}$ makes use of an adversary $\mathcal{A}$ that distinguishes $h(r^{x_{j_1}}) \oplus h(r^{x_{j_2}}) \oplus \ldots \oplus h(r^{x_{j_{l-1}}}) \oplus h(r^{x_{j_l}})$ from a random bit.

We know from a property of cover free families, that at least one of the secret keys remains unknown to the adversary. Considering this, applying a hardcore function to each ${X_{j_i}}^y = r^{x_{j_i}}$ (where $j_i$ are the elements of the session's cover-free family subset) results in at least one hardcore bit. In addition, if we consider the existence of an element $h(r^{x_{j_i}})$, $1 \leq i \leq l$, which is a hardcore bit, from Yao's XOR Lemma, it states that $h(r^{x_{j_1}}) \oplus \ldots \oplus h(r^{x_{j_l}})$ is also a hardcore bit. The adversary $\mathcal{B}$ first receives a challenge $(g, g^a, g^b, z)$, where he has to decide if $z$ is the hardcore predicate of $g^{ab}$ or a random bit. To simulate the adaptive chosen ciphertext interaction between $\mathcal{B}$ and $\mathcal{A}$, first $\mathcal{B}$ sets $r^* \leftarrow g^b$. From w = TCR($r^*$), $\mathcal{B}$ defines a q-CFF subset $\mathcal{F}_w$, picks a random element $\alpha \in \mathcal{F}_w$, and sets $x_\alpha = g^a$. $\mathcal{B}$ generates pairs of public/private keys for each position in the q-CFF $\{1, \ldots, d\} \backslash \{\alpha\}$.

$\mathcal{B}$ gives $\mathcal{A}$ the set of all public keys. For $\mathcal{A}$'s decryption query $(r, \psi)$, $\mathcal{B}$ calculates j = TCR($r$), for $\{j_1, \ldots, j_l\} \in \mathcal{F}_j$, and responds as follows:

**(a)** If j = w, $\mathcal{B}$ responds with $\perp$.

**(b)** If $\alpha \in \mathcal{F}_j$, where j $\neq$ w, $\mathcal{B}$ aborts the simulation and outputs a random bit to the challenger.

**(c)** Else, $\mathcal{B}$ computes the symmetric key $\overline{\mathsf{K}} = (h(r^{x_{j_1}}) \oplus h(r^{x_{j_2}}) \oplus \ldots \oplus h(r^{x_{j_{l-1}}}) \oplus h(r^{x_{j_l}}))$. Then, sends to $\mathcal{A}$ the pair ($\overline{\mathsf{K}}$, $\mathsf{M}$).

For the challenge, $\mathcal{A}$ sends $\mathsf{M}_0, \mathsf{M}_1$ to $\mathcal{B}$. $\mathcal{B}$ chooses $\beta \in \{0,1\}$ randomly and encrypts $\mathsf{M}_\beta$ with the symmetric key

$$\overline{\mathsf{K}} = (h(r^{*x_{j_1}}) \oplus \ldots \oplus h(r^{*x_{j_{\alpha-1}}}) \oplus z \oplus h(r^{*x_{j_{\alpha+1}}}) \oplus \ldots \oplus h(r^{*x_{j_l}}))$$

$\mathcal{B}$ sends $(r^*, \psi^*)$ to $\mathcal{A}$, and $\mathcal{A}$ sends back its guess $\beta$'. If $\beta$' = $\beta$, $\mathcal{B}$ outputs 1 (the bit is the hardcore of $g^{ab}$), else $\mathcal{B}$ outputs 0.

Throughout the proof, we will estimate probabilities based on the following events:

- *real*: Real session key is given to $\mathcal{A}$;
- *random*: Random bit is given to $\mathcal{A}$;

- *fake*: A fake key is given to $\mathcal{A}$;
- *succeed*: $\mathcal{B}$ does not abort the simulation and $\alpha \in \mathcal{F}_w$;
- *collision*: $\mathcal{A}$ submits a decryption query $r(\neq r^*)$ such that $\mathrm{TCR}(r) = \mathrm{TCR}(r^*)$;
- *a.abort*: $\mathcal{B}$ outputs a random bit in the artificial abort phase.

**Artificial Abort** The artificial abort technique plays an important role in the security proof. The reason we use this technique is primarily due to the fact that we cannot calculate $\Pr[\beta' = \beta | succeed \wedge ideal]$ but $\Pr[\beta' = \beta | succeed \wedge \overline{a.abort} \wedge ideal]$. Let $r^n$ $(1 \leq n \leq q)$ be $\mathcal{A}$'s decryption queries, and $j^n$ be $\mathrm{TCR}(r^n)$. Then $\mathcal{B}$ calculates the following value t:

$$t = |\mathcal{F}_{j^*} \setminus \bigcup_{s \in \{1,\ldots,q\}} \mathcal{F}_{j^s}|$$

Given that the experiment occurred without $\mathcal{B}$ having to abort, the probability that $\mathcal{B}$ outputs $\beta$' is $\Pr[\overline{a.abort}] = \frac{1}{t}$, and $\mathcal{B}$ will output a random bit with a probability $\Pr[a.abort] = \frac{t-1}{t}$.

We specify $\epsilon_{cdh,\mathcal{B}} = \frac{1}{2}|\Pr[\mathbf{Exp}^{cdh}_{\mathcal{B},\mathbb{G}}(k) = 1] - \Pr[\mathbf{Exp}^{rnd}_{\mathcal{B},\mathbb{G}}(k) = 1]|$. This way, we can estimate the probabilities:

$$\Pr[\mathbf{Exp}^{cdh}_{\mathcal{B},\mathbb{G}}(k) = 1] \geq \Pr[\mathbf{Exp}^{cdh}_{\mathcal{B},\mathbb{G}}(k) = 1 | \overline{collision}] \Pr[\overline{collision}]$$

$$\geq \Pr[\beta' = 1 | succeed \wedge \overline{a.abort} \wedge real \wedge \overline{collision}] \cdot$$

$$\Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision}] \Pr[\overline{collision}]$$

$$+ \frac{1}{2} \Pr[\overline{succeed} \vee a.abort | real \wedge \overline{collision}] \Pr[\overline{collision}]. \tag{4.7}$$

We will represent the randomness of adversary $\mathcal{A}$ by a coin, where $\mathcal{R}$ is the set of all possible values of *coin*. Considering the simulation of a CCA interaction is perfect when $succeed \wedge \overline{a.abort}$ is true, we have for all $R \in \mathcal{R}$

$$\Pr[\beta' = 1 | succeed \wedge \overline{a.abort} \wedge real \wedge \overline{collision} \wedge coin = R] = \tag{4.8}$$

$$Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 1 | real \wedge \overline{collision} \wedge coin = R]. \tag{4.9}$$

And also

$$\Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision} \wedge coin = R] =$$

$$\Pr[\overline{a.abort}|succeed \wedge real \wedge \overline{collision} \wedge coin = R]\cdot$$

$$\Pr[succeed|real \wedge \overline{collision} \wedge coin = R]$$

By definition, $\Pr[\overline{a.abort}|succeed \wedge real \wedge \overline{collision}] = \frac{1}{t(R)}$, where the value $t$ defined above depends on the value of *coin*. If $\overline{collision}$ is true, there always exists a non-empty subset $D$ of { 1, ..., d} such that $D \subseteq \mathcal{F}_{j^*}$ and $|D \cap \cup_{s\in\{1,\ldots,q\}}\mathcal{F}_{j^s}| = 0$ due to the propertiy of the CFF. This implies that *succeed* occurs if $a \in D$, and therefore, $\Pr[succeed|real \wedge \overline{collision}] = \frac{t(R)}{d}$. We have that

$$\Pr[succeed \wedge \overline{a.abort}|real \wedge \overline{collision}] = \frac{1}{t(R)} \cdot \frac{t(R)}{d} = \frac{1}{d} \tag{4.10}$$

From (4.8), (4.9) and (4.10)

$$\Pr[\beta' = 1|succeed \wedge \overline{a.abort} \wedge real \wedge \overline{collision}]\cdot$$

$$\Pr[succeed \wedge \overline{a.abort}|real \wedge \overline{collision}] \cdot \Pr[\overline{collision}] =$$

$$\sum_{R\in\mathcal{R}} (\Pr[succeed \wedge \overline{a.abort}|real \wedge \overline{collision} \wedge coin = R])$$

$$\cdot\Pr[succeed \wedge \overline{a.abort}|real \wedge \overline{collision} \wedge coin = R]$$

$$\cdot\Pr[coin = R|real \wedge \overline{collision}]\Pr[\overline{collision}] =$$

$$\sum_{R\in\mathcal{R}} Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1|real \wedge \overline{collision} \wedge coin = R] \cdot \frac{1}{d}$$

$$\cdot\Pr[coin = R|real \wedge \overline{collision}]\Pr[\overline{collision}] =$$

$$\frac{1}{d} \cdot \Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 1|real \wedge \overline{collision}] \cdot \Pr[\overline{collision}] \tag{4.11}$$

From (4.11) we have that

$$\Pr[succeed \wedge \overline{a.abort}|real \wedge \overline{collision}] = \frac{1}{d} \tag{4.12}$$

From (4.7)

$$\Pr[\mathbf{Exp}^{cdh}_{\mathcal{B},\mathbb{G}}(k)=1] \geq \frac{1}{d}\cdot\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=1|real\wedge\overline{collision}]\cdot\Pr[\overline{collision}]$$

$$+\frac{1}{2}\cdot\frac{d-1}{d}\cdot\Pr[\overline{collision}]$$

$$\geq \frac{1}{d}\cdot(\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=1|real]-\epsilon_{tcr})+\frac{1}{2}\cdot\frac{d-1}{d}\cdot(1-\epsilon_{tcr})$$

$$=\frac{1}{d}\cdot\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=1|real]+\frac{d-1}{2d}-\frac{d+1}{2d}\cdot\epsilon_{tcr} \tag{4.13}$$

By a analogous analysis,

$$\Pr[\mathbf{Exp}^{rnd}_{\mathcal{B},\mathbb{G}}(k)=0] \geq \frac{1}{d}\cdot\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=0|fake]+\frac{d-1}{2d}-\frac{d+1}{2d}\cdot\epsilon_{tcr} \tag{4.14}$$

Suppose we can construct an adversarial algorithm that distinguishes random bits from hard-core bits using adversary $\mathcal{A}$. Since this statement can not be true, we have

$$\frac{1}{2}|\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=0|random]-\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=0|fake]|\leq\epsilon_{hcb}$$

This way we can estimate the probabilities for adversary $\mathcal{B}$:

$$\frac{1}{2}|\Pr[\mathbf{Exp}^{cdh}_{\mathcal{B},\mathbb{G}}(k)=1]-\Pr[\mathbf{Exp}^{rnd}_{\mathcal{B},\mathbb{G}}(k)=1]|$$

$$\geq\frac{1}{2}|\frac{1}{d}\cdot\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=1|real]+\frac{d-1}{2d}-\frac{d+1}{2d}\cdot\epsilon_{tcr}$$

$$-(1-\frac{1}{d}\cdot\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=0|fake]+\frac{d-1}{2d}-\frac{d+1}{2d}\cdot\epsilon_{tcr})|$$

$$\geq\frac{1}{d}\epsilon_{kem}-\frac{1}{d}\epsilon_{hcb}-\frac{d+1}{2d}\epsilon_{tcr}$$

Therefore,

$$\epsilon_{kem}\leq d\cdot\epsilon_{cdh}+\epsilon_{hcb}+\frac{d+1}{2}\epsilon_{tcr}$$

□

### 4.1.2 IND-q-CCA2 Encryption From HDH

This construction is a variation of the one presented above. It yields in a IND-q-CCA PKE scheme based on HDH assumption also with optimal ciphertext length. In this construction, we equally make use of *key encapsulation* method to construct a key to be used in a symmetric encryption. Although the HDH assumption is stronger than CDH, this scheme enables a more efficient construction, due to the length of the symmetric key.

CONSTRUCTION

Let $H$: $\{0,1\}^k \times \mathbb{G} \rightarrow \{1,\ldots,s\}$ be a function family where the index space is $\{0,1\}^k$, $\pi$:$\{0,1\}^k \times \{0,1\} \rightarrow \{0,1\}$ be a permutation family where the index space is $\{0,1\}^k$ and H: $\{0,1\}^k \times \mathbb{G} \rightarrow \{0,1\}^n$ be a function family where the index space is $\{0,1\}^k$. We assume the existence of a cyclic group $\mathbb{G}$ of prime-order $p$ where the HDH assumption is believed to hold, i.e, given $(g, g^a, g^b)$ there is no efficient way to distinguish $\mathrm{H}(g^{ab}) \in \{0,1\}^n$ from a random string of bits of size $n$, for random $g \in \mathbb{G}$, and random $a$, $b \in \mathbb{Z}_p$.

Gen $(1^k)$: Define s(k) = $2^k$, d(k) = 16k$q^2$(k), l(k)=4kq(k). Run KGen. For i = 1, ..., d(k), KGen computes $X_i$=$g^{x_i}$ for $x_i \xleftarrow{\$} \mathbb{Z}_p$, and outputs pk=$(X_1,\ldots,X_{d(k)})$ and sk=$(x_1,\ldots,x_{d(k)})$. The public key is pk, and the secret key is sk.

Enc (pk, M): Run KEnc. KEnc computes r = $g^y$ for $y \xleftarrow{\$} \mathbb{Z}_p$, $j$ = TCR(r) where $\mathcal{F}_j = \{j_1,\ldots,j_l\}$ is the q-CFF subset associated to the value $j$ (which will define the set of the session's publlic/private keys), sets K = $r$ and calculates $\overline{\mathsf{K}} = \mathrm{H}((\prod_{j_i \in F_j} X_{j_i})^y)$. To encrypt message M, run symmetric-key encryption to obtain the ciphertext $\psi \leftarrow \pi_{\overline{\mathsf{K}}}(\mathsf{M})$. Output C=(r, $\psi$).

Dec (sk, C): Run KDec. KDec computes j = TCR(r) to obtain the subset $\mathcal{F}_j$, and calculates the session's symmetric key $\overline{\mathsf{K}} = \mathrm{H}(\ r^{\sum_{j_i \in F_j} x_{j_i}}\ )$. Decrypt $\psi$ to $\mathsf{M} \leftarrow \pi^{-1}_{\overline{\mathsf{K}}}(\psi)$.

AN IND-q-CCA2 PKE SCHEME BASED ON HDH.

**Theorem 2** *The above scheme is IND-q-CCA2 if the HDH assumption holds, TCR is a target collision resistant hash function, H is a one-way hash function, and $\pi$ is strongly pseudorandom.*

*Proof:* Just as in *Theorem 1*, the proof is made of two parts: In the first part we prove that $\Pi'$

is secure if $\overline{\mathsf{K}}$ is totally unknown to $\mathcal{A}$ and $\pi$ is a SPRP. In the second part of the proof, using the HDH assumption, we prove that the adversary cannot distinguish the session key from a random string.

**Lemma 3** $\Pi'$ *is IND-q-CCA2 if* $\overline{\mathsf{K}}$ *is totally unknown to* $\mathcal{A}$ *and* $\pi$ *is a* SPRP.

*Proof:* This proof is identical to the proof in *Lemma 1.*

**Lemma 4** *The adversary cannot distinguish the session key from a random bit if the HDH assumption holds.*

*Proof:* We prove this lemma by considering the existence of an adversary $\mathcal{B}$ that breaks the HDH assumption with nonnegligible probability. We assume that this adversary $\mathcal{B}$ interacts with an adversary $\mathcal{A}$ that distinguishes $\mathrm{H}(\prod_{i\in F_j} X_{j_i}^y)$ (i.e., the session key) from a random string.

Now we give details of the simulator. The adversary $\mathcal{B}$ first receives a challenge $(g, g^a, g^b, z)$, where he has to decide if $z$ is the hash of $g^{ab}$ or a random string. To simulate the adaptive chosen ciphertext interaction between $\mathcal{B}$ and $\mathcal{A}$, first $\mathcal{B}$ sets $r^* \leftarrow g^b$. From $\mathrm{w} = \mathrm{TCR}(r^*)$, $\mathcal{B}$ defines a q-CFF subset $\mathcal{F}_w$, and picks a random element $\alpha \in \mathcal{F}_w$. $\mathcal{B}$ generates pairs of public/private keys for each position in the q-CFF $\{1, \ldots, \mathrm{d}\}\backslash\{\alpha\}$ and sets $X_\alpha = g^a \cdot (\prod_{i\in F_w\backslash\alpha} g^{x_i})^{-1}$.

$\mathcal{B}$ gives $\mathcal{A}$ the set of all public keys. For $\mathcal{A}$'s decryption query $(r, \psi)$, $\mathcal{B}$ calculates $\mathrm{j} = \mathrm{TCR}(r)$, for $\{t_1, \ldots, t_l\} \in \mathcal{F}_j$, and responds as follows:

**(a)** If $\mathrm{j} = \mathrm{w}$, $\mathcal{B}$ responds with $\perp$.

**(b)** If $\alpha \in \mathcal{F}_j$, where $\mathrm{j} \neq \mathrm{w}$, $\mathcal{B}$ aborts the simulation and outputs a random bit to the challenger.

**(c)** Else, $\mathcal{B}$ computes the symmetric key $\overline{\mathsf{K}} = \mathrm{H}(\, r^{x_{j_1}} \cdot r^{x_{j_2}} \cdot \ldots \cdot r^{x_{j_{l-1}}} \cdot r^{x_{j_l}})$. Then, sends to $\mathcal{A}$ the pair $(\overline{\mathsf{K}}, \mathsf{M})$.

For the challenge, $\mathcal{A}$ sends $\mathsf{M}_0, \mathsf{M}_1$ to $\mathcal{B}$. $\mathcal{B}$ chooses $\beta \in \{0,1\}$. Since the challenge $\mathsf{M}_\beta$ is encrypted with respect to $r^*$, $\mathcal{B}$ should compute the symmetric key as

$$\overline{\mathsf{K}} = H\Big(X_\alpha^b \cdot \big(\prod_{j_i\in F_{j^*}\backslash\{\alpha\}} g^{x_{j_i}}\big)^b\Big)$$

But calculating the product inside de hash above and considering the value of $X_\alpha$ defined

before, the key will result in

$$\overline{\mathsf{K}} = H(g^{ab})$$

$\mathcal{B}$ sends $(r^*, \psi^*)$ to $\mathcal{A}$, and $\mathcal{A}$ sends back its guess $\beta$'. If $\beta$' $= \beta$, $\mathcal{B}$ outputs 1 (the string is the hash of $g^{ab}$), else $\mathcal{B}$ outputs 0.

To estimate the probabilities of the simulation above, we will use the same approach of section 3.1., and we also define the following events:

- *real*: Real session key is given to $\mathcal{A}$;
- *random*: Random string is given to $\mathcal{A}$;
- *fake*: A fake key is given to $\mathcal{A}$;
- *succeed*: $\mathcal{B}$ does not abort the simulation and $\alpha \in \mathcal{F}_w$;
- *collision*: $\mathcal{A}$ submits a decryption query $r(\neq r^*)$ such that $\mathrm{TCR}(r) = \mathrm{TCR}(r^*)$;
- *a.abort*: $\mathcal{B}$ outputs a random bit in the artificial abort phase.

**Artificial Abort** The artificial abort technique plays an important role in the security proof. The reason we use this technique is primarily due to the fact that we cannot calculate $\Pr[\beta' = \beta | succeed \wedge ideal]$ but $\Pr[\beta' = \beta | succeed \wedge \overline{a.abort} \wedge ideal]$. Let $r^n$ $(1 \leq n \leq q)$ be $\mathcal{A}$'s decryption queries, and $j^n$ be $\mathrm{TCR}(r^n)$. Then $\mathcal{B}$ calculates the following value t:

$$t = |\mathcal{F}_{j^*} \setminus \bigcup_{s \in \{1,\ldots,q\}} \mathcal{F}_{j^s}|$$

Given that the experiment occurred without $\mathcal{B}$ having to abort, the probability that $\mathcal{B}$ outputs $\beta$' is $\Pr[\overline{a.abort}] = \frac{1}{t}$, and $\mathcal{B}$ will output a random bit with a probability $\Pr[a.abort] = \frac{t-1}{t}$.

We specify $\epsilon_{hdh,\mathcal{B}} = \frac{1}{2}|\Pr[\mathbf{Exp}^{hdh}_{\mathcal{B},\mathbb{G}}(k) = 1] - \Pr[\mathbf{Exp}^{rnd}_{\mathcal{B},\mathbb{G}}(k) = 1]|$.

This way, we can estimate the probabilities:

$$\Pr[\mathbf{Exp}^{cdh}_{\mathcal{B},\mathbb{G}}(k) = 1] \geq \Pr[\mathbf{Exp}^{cdh}_{\mathcal{B},\mathbb{G}}(k) = 1 | \overline{collision}] \Pr[\overline{collision}]$$

$$\geq \Pr[\beta' = 1 | succeed \wedge \overline{a.abort} \wedge real \wedge \overline{collision}] \cdot$$

$$\Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision}] \Pr[\overline{collision}]$$

$$+\frac{1}{2}\Pr[\overline{succeed} \vee a.abort | real \wedge \overline{collision}]\Pr[\overline{collision}]. \tag{4.15}$$

We will represent the randomness of adversary $\mathcal{A}$ by a coin, where $\mathcal{R}$ is the set of all possible values of *coin*. Considering the simulation of a CCA interaction is perfect when $succeed \wedge \overline{a.abort}$ is true, we have for all $R \in \mathcal{R}$

$$\Pr[\beta' = 1 | succeed \wedge \overline{a.abort} \wedge real \wedge \overline{collision} \wedge coin = R] = \tag{4.16}$$

$$Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 1 | real \wedge \overline{collision} \wedge coin = R]. \tag{4.17}$$

And also

$$\Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision} \wedge coin = R] =$$

$$\Pr[\overline{a.abort} | succeed \wedge real \wedge \overline{collision} \wedge coin = R] \cdot$$

$$\Pr[succeed | real \wedge \overline{collision} \wedge coin = R]$$

By definition, $\Pr[\overline{a.abort} | succeed \wedge real \wedge \overline{collision}] = \frac{1}{t(R)}$, where the value $t$ defined above depends on the value of *coin*. If $\overline{collision}$ is true, there always exists a non-empty subset $D$ of { 1, ..., d} such that $D \subseteq \mathcal{F}_{j^*}$ and $|D \cap \cup_{s \in \{1,\ldots,q\}} \mathcal{F}_{j^s}| = 0$ due to the propertiy of the CFF. This implies that *succeed* occurs if $a \in D$, and therefore, $\Pr[succeed | real \wedge \overline{collision}] = \frac{t(R)}{d}$. We have that

$$\Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision}] = \frac{1}{t(R)} \cdot \frac{t(R)}{d} = \frac{1}{d} \tag{4.18}$$

From (4.16), (4.17) and (4.18)

$$\Pr[\beta' = 1 | succeed \wedge \overline{a.abort} \wedge real \wedge \overline{collision}] \cdot$$

$$\Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision}] \cdot \Pr[\overline{collision}] =$$

$$\sum_{R \in \mathcal{R}} (\Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision} \wedge coin = R])$$

$$\cdot \Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision} \wedge coin = R]$$

$$\cdot \Pr[coin = R | real \wedge \overline{collision}] \Pr[\overline{collision}] =$$

$$\sum_{R \in \mathcal{R}} Pr[\mathbf{Exp}^{kem}_{\mathcal{B},\Pi'}(k) = 1 | real \wedge \overline{collision} \wedge coin = R] \cdot \frac{1}{d}$$

$$\cdot \Pr[coin = R | real \wedge \overline{collision}] \Pr[\overline{collision}] =$$

$$\frac{1}{d} \cdot \Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 1 | real \wedge \overline{collision}] \cdot \Pr[\overline{collision}] \tag{4.19}$$

From (4.19) we have that

$$\Pr[succeed \wedge \overline{a.abort} | real \wedge \overline{collision}] = \frac{1}{d} \tag{4.20}$$

From (4.15)

$$\Pr[\mathbf{Exp}^{cdh}_{\mathcal{B},\mathbb{G}}(k) = 1] \geq \frac{1}{d} \cdot \Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 1 | real \wedge \overline{collision}] \cdot \Pr[\overline{collision}]$$

$$+ \frac{1}{2} \cdot \frac{d-1}{d} \cdot \Pr[\overline{collision}]$$

$$\geq \frac{1}{d} \cdot (\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 1 | real] - \epsilon_{tcr}) + \frac{1}{2} \cdot \frac{d-1}{d} \cdot (1 - \epsilon_{tcr})$$

$$= \frac{1}{d} \cdot \Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 1 | real] + \frac{d-1}{2d} - \frac{d+1}{2d} \cdot \epsilon_{tcr} \tag{4.21}$$

By a analogous analysis,

$$\Pr[\mathbf{Exp}^{rnd}_{\mathcal{B},\mathbb{G}}(k) = 0] \geq \frac{1}{d} \cdot \Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 0 | fake] + \frac{d-1}{2d} - \frac{d+1}{2d} \cdot \epsilon_{tcr} \tag{4.22}$$

Suppose we can construct an adversarial algorithm that distinguishes random bits from hard-core bits using adversary $\mathcal{A}$. Since this statement can not be true, we have

$$\frac{1}{2} |\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 0 | random] - \Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k) = 0 | fake]| \leq \epsilon_{hash}$$

This way we can estimate the probabilities for adversary $\mathcal{B}$:

$$\frac{1}{2}|\Pr[\mathbf{Exp}^{hdh}_{\mathcal{B},\mathbb{G}}(k)=1]-\Pr[\mathbf{Exp}^{rnd}_{\mathcal{B},\mathbb{G}}(k)=1]|$$

$$\geq \frac{1}{2}|\frac{1}{d}\cdot\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=1|real]+\frac{d-1}{2d}-\frac{d+1}{2d}\cdot\epsilon_{tcr}$$

$$-(1-\frac{1}{d}\cdot\Pr[\mathbf{Exp}^{kem}_{\mathcal{A},\Pi'}(k)=0|fake]+\frac{d-1}{2d}-\frac{d+1}{2d}\cdot\epsilon_{tcr})|$$

$$\geq \frac{1}{d}\epsilon_{kem}-\frac{1}{d}\epsilon_{hash}-\frac{d+1}{2d}\epsilon_{tcr}$$

Therefore,

$$\epsilon_{kem}\leq d\cdot\epsilon_{hdh}+\epsilon_{hash}+\frac{d+1}{2}\epsilon_{tcr}$$

□

# Chapter 5

# Conclusion

In our work we presented two IND-q-CCA2 PKE constructions, both under weak computacional assumptions, and with a optimal ciphertext overhead of one group element.

Our first result is of theoretical interest, which is a construction that leads to a IND-q-CCA2 PKE under CDH assumption, which is belived to be weakest among Diffie-Hellman assumptions. Whereas this scheme is not efficient, it is a great result, once we achieve a IND-CCA2 scheme for a bounded number of oracle access, under CDH assumption with a ciphertext overhead of one group element. Nowadays, there is no PKE scheme with similar characteristics.

Our second result is an efficient IND-q-CCA2 scheme under the HDH assumption, and also with one group element of ciphertext overhead. This scheme is very similar to the one presented in [19], however, our proof required a construction of the simulator. As in the CDH case, there is also a lack of schemes proved for a IND-CCA2 model (even with the restriction of bounded number of acces to the oracle) and under HDH assumption. In addition we achieve a efficient scheme with optimal ciphertext overhead.

For future works, it would be interesting an enhancement of of our CDH scheme to a efficient scheme under the same conditions, and also The construction of signature schemes with similar conditions.

In conclusion,given the importance that public-key schemes have for cryptography, a great effort should be made to build new protocols towards enhancing the efficiency and weakening the computational assumptions, and proven for high level security models.

# REFERENCES


[1] GOLDWASSER, S.; WIGDERSON, S. M. ans A. How to play any mental game or a completeness theorem for protocols with honest majority. *STOC '87*, 1987.

[2] DIFFIE, W.; HELLMAN, M. E. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22, p. 644–654, 1976.

[3] RACKOFF, C.; SIMON, D. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. *Advances in Cryptology – Crypto '91*, v. 576 of Lecture Notes in Computer Science, p. 434–444, 1991.

[4] DOLEV, D.; DWORK, C.; NAOR, M. Non-malleable cryptography. *In Proc. 23rd ACM Symp. on Theory of Computing, pages 542–552*, 1991.

[5] NAOR, M.; YUNG, M. Public-key cryptosystems provably secure against chosen ciphertext attacks. *STOC 90*, 1990.

[6] SAHAI, A. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. *FOCS '99*, p. 543–553, 1999.

[7] LINDELL, Y. A simpler construction of cca2-secure public-key encryption under general assumptions. *Journal of Cryptology*, v. 19(3), n. 359-377, 2006.

[8] CRAMER, R.; SHOUP, V. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. *CRYPTO '98*, v. 1462 of LNCS, n. 13-25, 1998.

[9] BONEH, D.; FRANKLIN, M. Identity based encryption from the weil pairing. *Crypto*, 2001.

[10] CANETTI, R.; HALEVI, S.; KATZ, J. Chosen-ciphertext security from identity-based encryption. *Cryptology ePrint Archive, Report 2003/182*, 2003.

[11] CRAMER, R. et al. Bounded cca2-secure encryption. *Advances in Cryptology – ASIACRYPT 2007*, v. 4833/2008, p. 502–518, 2007.

[12] RABIN, M. O. Digitalized signatures and public-key functions as intractable as factorization. *Technical Report MIT/LCS/TR-212, MIT Laboratory for Computer Science*, 1979.

[13] GOLDWASSER, S.; MICALI, S. Probabilistic encryption. *Journal of Computer and System Science*, v. 28, n. 270-299, 1984.

[14] CANETTI, R.; GOLDREICH, O.; HALVEI, S. The random oracle model, revisited. *Proceedings of the 30th Annual ACM Symposium on the Theory of Computing*, v. 209-218, 1998.

[15] ELGAMAL, T. A public-key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, IT-31, n. 4, p. 469–472, 1985.

[16] SHOUP, V. Sequences of games: a tool for taming complexity in security proofs. *Cryptology ePrint Archive, Report 2004/332*, 2004.

[17] ROSEN, A.; SEGEV, G. Chosen-ciphertext security via correlated products. *Cryptology ePrint Archive, Report 2008/116*, 2008.

[18] HOFHEINZ, E. K. D. Secure hybrid encryption from weakened key encapsulation. *CRYPTO 2007*, p. 553–571.

[19] HANAOKA, G.; IMAI, H. A generic construction of cca-secure cryptosystems without nizkp for a bounded number of decryption queries. *Cryptology ePrint Archive, Report 2006/408*, 2006.

[20] YAO, A. C. Theory and applications of trapdoor functions. *23st FOCS*, 1982.

[21] CRAMER, R.; SHOUP, V. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, v. 33, n. 1, p. 167–226, 2003.

[22] GOLDREICH, O.; NISAN, N.; WIGDERSON, A. On yao's xor-lemma. *Technical Report TR95-050, Electronic Colloquium on Computational Complexity*, 1995.

[23] BELLARE, M.; RISTENPART, T. Simulation without the artificial abort: Simplified proof and improved concrete security for waters' ibe scheme. *Advances in Cryptology – EUROCRYPT '09*, 2009.

[24] GOLDREICH, O. *Foundations of Cryptography: Basic Tools*. [S.l.]: Cambridge University Press, 2001.

[25] GOLDREICH, O. *Foundations of Cryptography: Basic Applications*. [S.l.]: Cambridge University Press, 2004.

[26] DENT, A. A brief history of provably-secure public-key encryption. *Cryptology ePrint Archive, Report 2009/090*, 2009.

[27] HANAOKA, G.; KUROSAWA, K. Efficient chosen ciphertext secure public key encryption under the computational diffie-hellman assumption. 2008.

akechapterhead[1]@ **5. 1**

@