CRYPTANALYSIS, PROVABLE SECURITY AND IMPLEMENTATION OF FUZZY IDENTITY-BASED CRYPTOGRAPHY

TAN SYH YUAN

DOCTOR OF PHILOSOPHY IN ENGINEERING

FACULTY OF ENGINEERING AND SCIENCE
UNIVERSITI TUNKU ABDUL RAHMAN
NOVEMBER 2014

CRYPTANALYSIS, PROVABLE SECURITY AND IMPLEMENTATION OF FUZZY IDENTITY-BASED CRYPTOGRAPHY

By

TAN SYH YUAN

A thesis submitted to the Department of Mechatronics and BioMedical Engineering,
Faculty of Engineering and Science,
Universiti Tunku Abdul Rahman,
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Engineering
November 2014

To my parents

ABSTRACT

CRYPTANALYSIS, PROVABLE SECURITY AND IMPLEMENTATION OF FUZZY IDENTITY-BASED CRYPTOGRAPHY

Tan Syh Yuan

We study the ways of achieving authentication using a hybrid of biometrics and cryptography, namely, Fuzzy Identity-Based Cryptography (FIBC) that allows a user to answer both the questions *who you are* and *what you have* without deteriorating the security of either side. We point out some implementation issues in the variant of Fuzzy Identity-Based Encryption (FIBE), namely, biometric-based IBE (Bio-IBE) schemes and show the workarounds. Besides, from cryptanalysing two FIBS schemes, we identify the insecure constructions of key generation algorithms in FIBC. Since FIBC is closely related to its underlying IBC schemes and there is only one Fuzzy Identity-Based Identification (FIBI) scheme in the literature to date, we perform analysis on IBI schemes instead of FIBI. As a result, we unearth a subtle flaw in the security proofs of an IBI scheme in the standard model and fix it neatly. Furthermore, we ascertain for IBI schemes in the random oracle model, two proving techniques in reducing the generally acknowledged security parameter of length k^2 bits to only k bits in achieving the same security level. Compiling the cryptanalysis outputs, FIBI turns out to be the optimum solution for our research goal. Thus, as a proof of concept, we implement an efficient FIBI scheme which supports the use of discretised biometrics as the public key.

ACKNOWLEDGEMENTS

I would like to express my sincere appreciations to my supervisors Prof. Dr. Goi Bok Min, Prof. Dr. Raphael Phan Chung Wei and Prof. Dr. Heng Swee Huay for their precious guidance and enlightenment throughout these years. Their wisdom and kindness will be engraved into my memory.

Besides, I would like to thank Prof. Andrew Teoh Beng Jin, Jin Zhe, Chin Ji Jian, Yau Wei Chuen, Yap Wun She, Khoh Wee How, Yap Hui Yen, Rouzbeh Behnia, Chong Zan Kai, Wong Chee Siang, Tan Yar Ling, Harlem PY Shake and others, who have been playing at least one of the roles as my co-author, friend, colleague and comrade.

Finally, special thanks to my family members for their consideration and support for my road not taken.

APPROVAL SHEET

This thesis entitled "Cryptanalysis, Provable Security and Implementation of Fuzzy Identity-Based Cryptography" was prepared by TAN SYH YUAN and submitted as partial fulfillment of the requirements for the degree of Doctor of Philosophy in Engineering at Universiti Tunku Abdul Rahman.

Approved by:

(Prof. Dr. Goi Bok Min) Date: 20 October 2014 . . . . .
Supervisor
Department of Mechatronics and BioMedical Engineering
Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

(Prof. Dr. Raphael Phan Chung Wei) Date: 20 October 2014 . . . . .
Co-Supervisor
Faculty of Engineering
Multimedia University

FACULTY OF ENGINEERING AND SCIENCE

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 20 October 2014

SUBMISSION OF THESIS

It is hereby certified that **TAN SYH YUAN** (ID No: **11UED06189**) has completed this thesis entitled "**Cryptanalysis, Provable Security and Implementation of Fuzzy Identity-Based Cryptography**" under the supervision of Prof. Dr. Goi Bok Min (Supervisor) from the Mechatronics and BioMedical Engineering, Faculty of Engineering and Science, and Prof. Dr. Raphael Phan Chung Wei (Co-Supervisor) from the Faculty of Engineering, Multimedia University.

I understand that University will upload softcopy of my thesis in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

(Tan Syh Yuan)

DECLARATION

I hereby declare that the dissertation is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTAR or other institutions.

Name: Tan Syh Yuan

Date: 20 October 2014

TABLE OF CONTENTS

CHAPTERS **Page**

CHAPTERS **Page**

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

This chapter introduces the motivations for this project and briefs its results. History of fuzzy identity-based cryptography is briefly discussed starting from the public key cryptography and identity-based cryptography. It also explains the contribution of this project to the cryptography community particularly on authentication services.

1.1 Background

Authentication is one of the fundamental security goals in information security. It is widely needed in many electronic applications (verifier) that need to authenticate a user (prover), i.e., to make sure the user is genuine. Authentication is normally achieved in an information system by asking provers for their credentials which are formed by one or more of the following:

- What you know: Password, pass phrase, secret questions, cryptographic key, etc.
- Who you are: Biometrics (e.g., fingerprint, face, voice, signature, etc.)
- What you have: Security dongle/token, smart card, smart phone, etc.

Authenticate a prover based on his password is the most common practice nowadays but such authentication service requires the prover to trust the verifier which stores the password. The similar requirement is applied to authentication services provided by symmetric key cryptography, where both prover and verifier share the same secret key and the authentication services can be proven secure mathematically. In certain cases where stringent security is desired, provers do not trust the verifier and it is considered insecure to share provers' sensitive information with the server. For instance, the verifier can impersonate prover using the sensitive information on hand. The solutions for this stringent requirement can be found in public key cryptography as well as its successor such as identity-based cryptography and fuzzy identity-based cryptography.

As cryptography key is normally in the form of an unreadable long random string, it is stored in security devices such as security dongle, security token, smart card, smart phone and so on. Security devices cannot escape from physical security problem such as cryptography key-lost problem, where an impersonator can steal the security devices and perform unauthorised authentications. The alternative authentication approach to overcome the cryptography key-lost problem is biometrics, where one's biometrics will never lost and there is no need to memorise it. However, biometrics cannot be proven secure as in cryptography, as its security is based on empirical evidence which is closely related to the quality of biometric input such as fingerprint, face, iris, voice, hand signature and so on.

This project works towards the stringent security direction and explores the secure combination of cryptography and biometrics in providing authentication service: provide both provable security (from cryptography) and physical security (from biometrics). Note that an authentication approach which combines these two authentication ingredients naively is not necessary secure. Let's consider the scenario where a prover wishes to authenticate himself/herself to a verifier which is an electronic vault. Assume that the prover's biometric data and cryptographic key are stored in a handheld device, the verifier can verify the prover's biometrics either before or after allowing the prover to be authenticated cryptographically. For instance, a prover must present his fresh fingerprint reading before he can interact cryptographically with the vault. If the fingerprint reading matches the record stored inside the handheld device, the prover is allowed to proceed to interact with the verifier cryptographically. This two-factor authentication mechanism can prevent cryptography key-lost problem as anyone other than the owner who holds the handheld device cannot get himself authenticated simply because they can't provide a genuine biometrics reading.

However, there is a problem in the scenario above where there exists no linkage between the prover's biometrics and the prover's cryptographic key. If an impersonator can get hold of the handheld device, he can replace[1] the prover's biometric with that of his own. It is obvious that the impersonator will always succeed in authenticating himself to the verifier and so the security boils down to the provable security provided by cryptography only. Such se-

[1]Cryptography key of the handheld devices cannot be changed or the authentication service such as the challenge-respond protocol of public key encryption scheme will fail.

curity loophole can be prevented only if the biometric is stored in the verifier's database for matching purposes but the downsides are the need:

1. of prover to trust the verifier;
2. for verifier to deal with database management and security issues; and
3. to increase the cost on verifier's end.

The results of this project suggests the feasible solutions in solving the cryptographic key-lost problem and also easing the cost burden on verifier's end. The results consist of cryptanalysis of several bio-crypto schemes, new proving techniques with tight security reduction and the proof of concept of a provably secure database-less bio-crypto authentication solution.

1.2 Related Technologies

Before describing in details the motivation and contribution, we briefly discuss a few technologies which have been identified as the potential tools in integrating biometrics and cryptography to provide a physically and provably secure authentication service.

1.2.1 Identity-Based Cryptography

Diffie and Hellman (1976) popularised the thought of public key cryptography (PKC) and solved the key distribution problem in symmetric key cryptography. However, the short of PKC is that it requires a Certification Authority

(CA) to generate a certificate in order to guarantee the validity of a user public key. This leads to the storage and key management problems of the certificates and public keys.

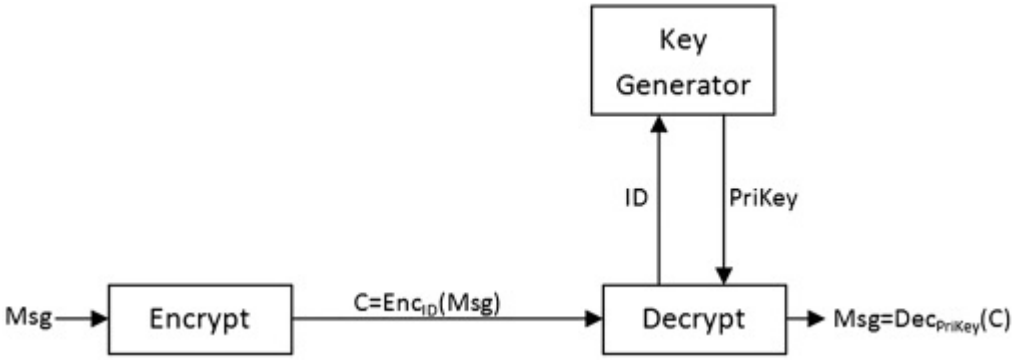


Figure 1.1: Concept of Identity-Based Cryptography

The design of a secure and efficient cryptographic scheme without certificate becomes the goal of many cryptographers and this leads to the idea of identity-based cryptography (IBC) (Shamir, 1985). In IBC, the public key is the user's publicly verifiable identity (e.g. name, ID number, email, etc.) as depicted in Figure 1.1. A trusted third party (TTP), namely, private key generator (PKG) is required to generate the user private key (upk) for every user based on their public key and this rules out the need of the storage of certificates and public keys. Some cryptosystems were proposed under the setting of identity-based but they are facing the problem of identity uniqueness in practice despite the security of the schemes are provable. Particularly, the system administrator of IBC needs to select from each user, an identity which suits their organisation the best. Otherwise, each user needs to register a new public identity as the public key such as matrix number, office room number, company email, company phone number etc., where troublesome procedures and documents are involved. Besides, there will be cases where these user public keys are expired or revoked.

1.2.2 Fuzzy Identity-Based Cryptography

The solution for the unique identity problem is the marriage of IBC and biometrics technology which uses the user public biometric identity that can be obtained easily. This solution was coined as fuzzy identity-based cryptography (FIBC) (Sahai and Waters, 2005) in order to solve the identity registration and key expiry problem in IBC. Sahai and Waters (2005) outlined the concept of FIBC by presenting one of the primitives of IBC, namely fuzzy identity-based encryption (FIBE) scheme which will be discussed later.

FIBC can be viewed as an extension to IBC where public identity in IBC is now a set of descriptive attributes. Therefore, IBC is actually a special case of FIBC where there is only one value in the public identity. FIBC was created to serve biometric-based encryption which is having advantage on the uniqueness of the biometric identity. Moreover, since biometric identity is linked to human naturally, FIBC can overcome the key expiry problem of IBC and PKC.

1.2.2.1 Fuzzy Identity-Based Encryption

FIBE as shown in Figure 1.2 allows a user who holds an genuine biometric identity (ID) to decrypt a ciphertext encrypted with another set of biometric identity (ID'), if and only if the user identity sets ID and ID' are less than a predefined Hamming distance[2]. Some may argue that made public the biometric

[2]Hamming distance is the minimum number of bits needed to be flipped in ID in order for ID to be the same as ID'. Please refer to Definition 2.13 for formal description.

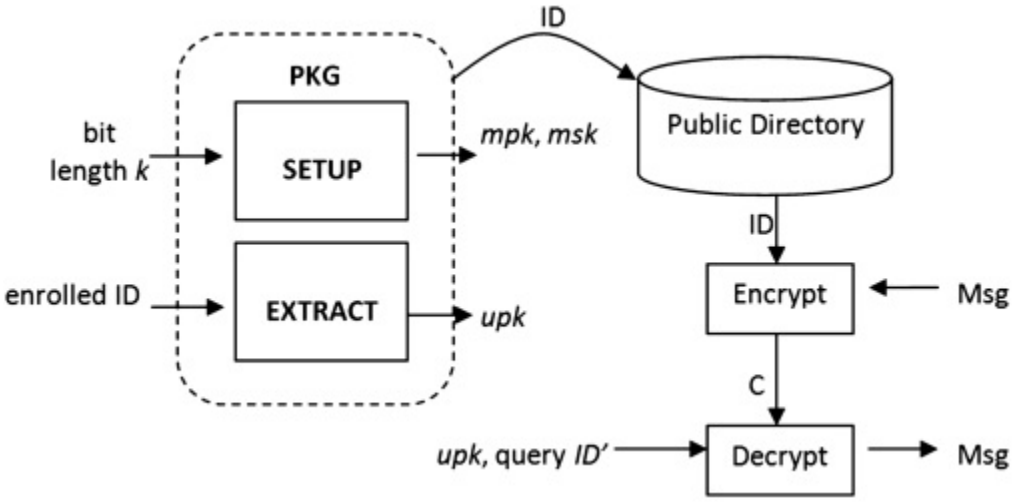


Figure 1.2: Model of Fuzzy Identity-Based Encryption

data violates user privacy but this is resolvable using biometric template protection techniques such as biometric salting, non-invertible transform, key binding and so on (Teoh et al., 2006, Jain et al., 2008).

As suggested by Sahai and Waters (2005), a naive way of constructing FIBE is to apply multiple user public identities (multi-*ID*) setting in IBE. In multi-*ID* IBE, in addition of generating IBE's system parameters such as public and private keys, the **Setup** algorithm specifies the threshold value d. So, a user will get his secret key upk_i for each of his identity ID_i as a result of running key extraction algorithm for $1 \leq i \leq n$. During encryption, encrypter encrypts the plaintext with multiple identities. Decryption is possible only when the decrypter has at least d out of n upk_i corresponding to the identities in the ciphertext. It is clear that the multi-*ID* setting can be easily adopted by any IBE while preserving the existing security properties. However, this setting creates a security problem, namely *collusion attack* (Sahai and Waters, 2005).

For an example, a multi-*ID* IBE encryption algorithm fixes a threshold value $d = 3$ and a ciphertext is generated for user C. Assume that the extracted biometric data of user A is $ID_A = \{1,2,3,4,5\}$, user B is $ID_B = \{6,7,8,9,10\}$ and user C is $ID_C = \{4,5,6,11,12\}$. Since each upk_i is bond to the ID_i independently, users A and B can combine their upk to generate a new set of secret key corresponding to the identity $ID_{A\cup B} = \{1,2,3,4,5,6,7,8,9,10\}$. As $|ID_{A\cup B} \cap ID_C| = 3 = d$, user A and B can decrypt user C's ciphertext just by colluding their user secret keys. In order to avoid the collusion attack, SW-FIBE uses Lagrange Polynomial (LP) to bind elements $\mu_i \in ID$ in key extraction to a randomly chosen secret polynomial (in upk). With this protection, when A and B collude their upk, they cannot decrypt the ciphertext as the upk obtained from polynomial interpolation is not the same as that of C's.

Only a few pairing-based FIBE schemes (Sahai and Waters, 2005, Baek et al., 2007, Ren et al., 2010, Shi et al., 2010) and lattice-based FIBE scheme (Agrawal et al., 2012) appeared in the literature and FIBE swiftly evolved to attribute-based encryption (ABE) (Goyal et al., 2006, Bethencourt et al., 2007) when it is discovered that FIBE cannot[3] really use biometrics as the public identity. Sahai and Waters (2005) claimed that FIBE is also an ABE but their FIBE can only be considered as a general framework of ABE (Goyal et al., 2006, Bethencourt et al., 2007) such that a FIBE scheme is an IBE scheme with a special predicate f, namely, the k-out-of-n threshold function. To date, there is no concrete implementation example given on using biometrics identity/attribute in

[3]Please refer to Chapter 5 for the problems in detail.

the predicate f of ABE as well.

1.2.2.2 Fuzzy Identity-Based Signature

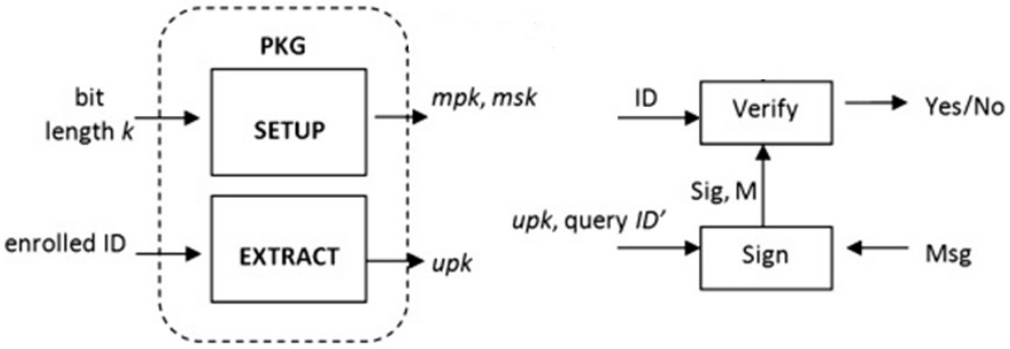


Figure 1.3: Model of Fuzzy Identity-Based Signature

On the other hand, identity-based signature (IBS) scheme is extended into fuzzy IBS (FIBS) scheme (Yang, Cao and Dong, 2011, Chen, Zhu, Cao and Geng, 2009, Wang et al., 2009, Wang and Kim, 2009, Wang, 2012, Yao and Li, 2014, Yang et al., 2014, Xiong et al., 2014). The first FIBS in the literature was proposed by Yang, Cao and Dong (2011) by adopting the key extraction technique of Sahai and Waters' FIBE and the signature is generated by using the query public biometric identity. The signature of FIBS can be verified successfully if and only if ID and ID' are overlapped for certain distance metric where ID is the enrolled public biometric identity that is used by PKG during key extraction algorithm as depicted in Figure 1.3. The most efficient FIBS scheme is the FIBS by Wang and Kim (2009), which is claimed to be existentially unforgeable under the chosen message attack and fuzzy identity attack in the random oracle model assuming the discrete logarithm problem is computa-

tionally hard. On the other hand, the most flexible FIBS scheme is the FIBS by Chen, Zhu, Cao and Geng (2009) which is proven secure against unforgeability in the standard model if the multi-sequence of Diffie-Hellman exponents problem is computationally hard. Some post-quantum FIBS schemes (Yao and Li, 2014, Yang et al., 2014) were also proposed based on hard problems in lattices but more works have to be done for them to be used in practice.

1.2.2.3 Fuzzy Identity-Based Identification

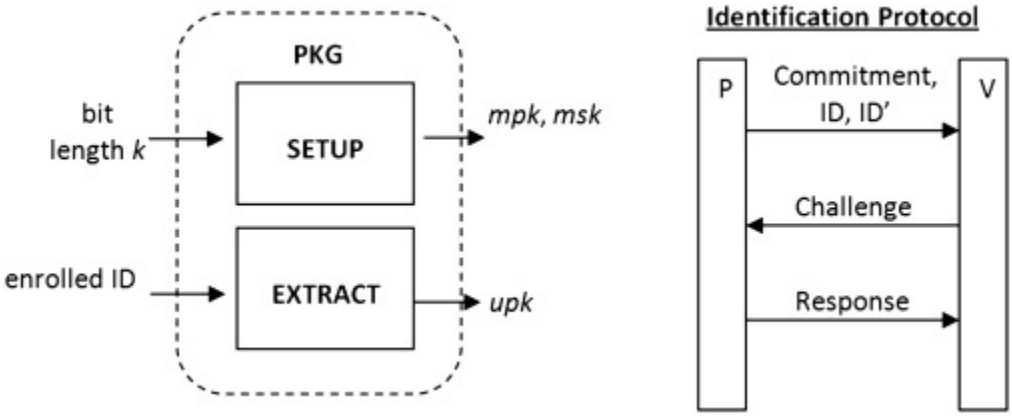


Figure 1.4: Model of Fuzzy Identity-Based Identification

Fiat and Shamir (1987) proposed identification scheme which allows a prover who holds a user private key to authenticate himself to a verifier who holds the corresponding public key. An identification scheme can provide repudiation for prover as verifier learns nothing more than the fact that the prover owns a valid user private key at the end of identification protocol. In precise, an identification scheme can provide authentication service with the following properties (Menezes et al., 1996):

1. Completeness: If both prover and verifier are honest, Bob can complete the identification protocol to accept prover's identity as genuine.
2. Soundness: The probability for Bob to accept prover's identity is negligible in the case where an impersonator tries to impersonate prover by running the identification protocol with verifier.
3. Zero knowledge:

   - verifier cannot reuse the past communication with prover to impersonate the prover to a third party.

   - verifier cannot convince a third party that a prover communicated him because the communication record can be simulated and is indistinguishable to authentic record.

If an identification scheme is proven secure, the properties will remain true even:

1. a polynomially large number of identification protocol of prover and verifier have been observed.
2. an impersonator participated in pervious communication with either prover or verifier, or both of them.
3. when identification protocol is initiated by impersonator in parallel.

Identification scheme has also been fuzzified, namely Fuzzy Identity-based Identification (FIBI) (Tan et al., 2009) by using the similar technique of FIBE and FIBS. In FIBI, a user who holds the enrolled public biometric identity ID will be verified successfully by a verifier which holds the query biometric identity

ID' if ID' is a genuine identity and at least d elements of the user private key is confirmed to be valid, ie. $|ID \cap ID'| \geq d$. Therefore, IBI is a special case of FIBI where the public identity in IBI is a singleton. The advantage of FIBI against FIBE is that it does not need a public directory to keep the enrolled ID because the authentication process is done in real time[4]; although FIBS also does not need a public directory, it cannot provide repudiation as FIBI does. Up to date, there is only one FIBI scheme appeared in the literature and no implementation is given (Tan et al., 2009). We summarize the similarities of these primitives in Table 1.1.

Table 1.1: Similarities of FIBC Primitives

	FIBE	FIBS	FIBI
Setup	Same	Same	Same
Extract	Same	Same	Same
Encrypt	Encrypt using ID	-	-
Decrypt	Decrypt using ID'	-	-
Sign	-	Sign using ID'	-
Verify	-	Verify using ID	-
Identification Protocol	-	-	Authenticate using ID'
Require Public Directory	Yes	No	No
Repudiation	No	No	Yes

[4]Prover can send the enrolled ID from smart card and the query ID' from a biometric reader.

1.2.3 Comparisons of Related Technologies

Although FIBE and FIBS provide confidentiality and integrity respectively but not authentication, these two primitives can actually be used to provide authentication service (Menezes et al., 1996) which are very similar to FIBI scheme. For instance, FIBE can be used to construct an authentication service in such a way that:

1. Prover signals verifier to start an authentication process.
2. Verifier encrypts a random nonce and sends the ciphertext to prover.
3. Prover decrypts the ciphertext and returns the nounce to verifier.
4. Verifier authenticates prover if the decrypted nonce is the same as the original's, rejects otherwise.

While FIBS also can be used in a similar way:

1. Prover signals verifier to start an authentication process.
2. Verifier sends random nonce to prover.
3. Prover signs the nonce and returns the signature to verifier.
4. Verifier authenticates prover if the signature is verified, rejects otherwise.

However, both constructions cannot provide repudiation property to prover. Repudiation indicates the ability to deny an action and it can only be found in FIBI. This repudiation property protects user privacy where a user can deny that he runs an authentication service with a verifier previously. For instance, several

cabinet members are asked by president to attend a secret meeting in a meeting room secured by FIBI. The cabinet members run the authentication service provided by FIBI and entered the room successfully. After a period of time, media realised this secret meeting and somehow obtained the FIBI log of the meeting room. However, due to the zero knowledge property of FIBI, nobody can prove that the listed cabinet members attended the secret meeting, or the log record is simulated. We hereby compare the pros and cons of existing authentication services in Table 1.2.

Table 1.2: Properties of Authentication Methods

Method	Properties					
	Database-less	What you know	Who you are	Repudiation	Authentication	Provable Security
Username + Password	×	√	×	√	√	×
Passphrase	×	√	×	√	√	×
Secret Questions	×	√	×	√	√	×
Biometrics	×	×	√	×	√	×
Symmetric Key	×	√	×	√	√	√
IBE	√	√	×	×	×	√
IBS	√	√	×	×	√	√
IBI	√	√	×	√	√	√
FIBE	×	√	√	×	√	√
FIBS	√	√	√	×	√	√
FIBI	√	√	√	√	√	√

1.3 Motivation

We are particularly interested in using FIBC for authentication purposes as it is the only primitive which can answer both "what you know" and "who you are", besides having provable security. However, FIBC is well known for having high complexity, high memory consumption and non-tight security reduction which are remain unsolved to date. Moreover, to the best of our knowledge, no implementation of FIBC using biometrics has been reported. The main challenge of implementing FIBC is to combine the fuzzy biometrics input with cryptographic operations in finite field. Even when this step is done, it is unsure if cryptography can be complimented by biometrics or getting worse in term of provable security.

1.4 Objectives and Contributions

The objectives of this project are to:

1. examine the practicality of fuzzy identity-based cryptosystems with respect to authentication service.
2. cryptanalyse the existing fuzzy identity-based encryption, signature and identification schemes.
3. provide tight reduction proofs for fuzzy identity-based cryptosystems.
4. implement a proof of concept for fuzzy identity-based cryptosystems.

The contribution of the project are as follows. Firstly, the project presents the analysis results and the suggested improvements on the models of FIBE schemes and its variants, namely, biometric-identity-based encryption (Bio-IBE) schemes.

Secondly, the project illustrates the cryptanalysis of two FIBS schemes and shows that certain sensitive criteria need to be taken care during scheme designing. Since there is only one FIBI (Tan et al., 2009) was proposed to date, the cryptanalysis on identity-based identification schemes is focused[5] instead. The outcome consist of a cryptanalysis and the corresponding solution of an IBI scheme in the standard model where the patched IBI is now provably secure and more efficient than the original version. Based on the cryptanalysis result, FIBI appears to be the most suitable candidate for authentication application as shown in Table 1.1.

Thirdly, in order to improve the efficiency of the only provably secure FIBI scheme, this project proposes two proving techniques to reduce for half, the existing security parameter of the underlying IBI scheme but achieve the same security level. Above and beyond, these two proving techniques indicate that several IBI schemes are practically secure with the shorter security parameter.

Lastly, this project claims the first realization of FIBC by providing an implementation of a FIBI scheme. Although other primitives (Sahai and Waters, 2005, Baek et al., 2007, Yang, Cao and Dong, 2011) are of different natures, they

[5]More results will be shown in Chapter 3 on the relation of IBI and FIBI.

can adopt the proposed technique for the **Setup** and **Extract** algorithms.

1.4.1 Thesis Outline

This project is outlined as follows:

- Chapter 2 provides the preliminaries and related mathematical notations. It briefly describes the hard mathematical problems and assumptions, as well as the security notions for FIBC.
- Chapter 3 presents the cryptanalysis of FIBE, FIBS and IBI schemes. It identifies the crucial points in designing and proving the security of FIBC schemes.
- Chapter 4 explains the techniques of upgrading the security of FIBI schemes by digging deep into the fundamentals of provable security. It confirms the security strength of IBI schemes in practice is as portrayed by the mathematical proofs.
- Chapter 5 presents a proof of concept for a FIBI scheme. It discusses the implementation issues and the constrains of a FIBI scheme based on the bio-crypto techniques to date.
- Chapter 6 concludes the findings in this project and briefly discusses the remaining open problems.

CHAPTER 2

PRELIMINARIES

This chapter introduces the concept of the provable security in cryptography. The definition of some mathematical notations and hard problems are also provided. Throughout the thesis, we will use q to denote the order and p to denote the modulus of a group.

2.1 Provable Security

In order to provide security assurance for a cryptographic scheme, a sound security proof is necessary. Security proofs are normally designed in such a way that a hard problem (e.g., discrete logarithm (DL), factorisation, computational Diffie-Hellman (CDH), etc.) can be linked to the scheme. Subsequently, the proof designer shows if the scheme can be broken, then the underlying hard problem can be solved. However, as it is assumed that the selected hard problem is intractable, thus this is a contradiction and the scheme is secure. Usually the proof designer needs to make some assumptions to complete the proof, such as the existence of a random oracle.

2.1.1 Random Oracle Model

In security proof, the random oracle is used to represent what is often considered in real settings as a hash function that consists of a source of true randomness with a customizable list of input and its corresponding output. More precisely, in addition to the indistinguishability of the outputs generated by a random oracle from those by a hash function, one can program the random oracle, which is not possible for any hash function in practice. Schemes that rely on such assumptions are termed as provably secure in the random oracle model (ROM) (Bellare and Rogaway, 1993). ROM is an impractical assumption, some works (Canetti et al., 2004, Goldwasser and Kalai, 2003, Fujisaki et al., 2009) have shown that some carefully crafted schemes proven secure in ROM are not necessarily secure when instantiated using concrete constructions such as hash functions. However, ROM is still widely accepted as an assurance of security for to date, no scheme reported to have serious flaws if it is properly proven secure in the ROM.

2.1.2 Standard Model

If one would like to have a better security assurance, doing away with the random oracle is required and such a model is known as the standard model. As a trade-off, schemes that are proven secure under the standard model normally suffer a significant increase in the algorithm complexity. This is because the normal hash functions are now replaced with some complex programmable hash

functions (Hofheinz and Kiltz, 2012). We note that it is not easy to prove the security of a scheme in the standard model as no assumption can be made on the hash function. In fact, some schemes may not even be possible to be proven secure in the standard model due to the nature of their construction (Nielsen, 2002, Bellare et al., 2004).

2.2 Mathematical Background

This section briefly states some hard mathematical problems together with the relevance notations. Throughout the thesis, we use $\mathbb{G}$ to denote a cyclic group of order q modulo p.

2.2.1 Bilinear Pairings

Bilinear pairing was first appeared as an attack tool in the cryptography world where it is used in breaking elliptic curve cryptosystems (ECC) (Menezes et al., 1991) by reducing the logarithm problem in the elliptic curve to the discrete logarithm in the multiplicative group of the underlying finite field. Ten years later, cryptographers found that the invasive properties of bilinear pairing that were used in attack, can actually be used to construct the identity-based cryptosystems and solved the open problems proposed by Shamir (1985).

Definition 2.1. *Let $\mathbb{G}$ and $\mathbb{G}_T$ be two distinct cyclic groups of prime order q and let g_1, g_2 be two distinct generators in $\mathbb{G}$. The bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is said to be an admissible map if it fulfils the following conditions:*

1. *Bilinearity:*

   - The placements of the scalar multipliers of the two points to e does not affect the mapping result.

   - Example: $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab} = e(g_1^b, g_2^a)$ *for all* $a, b \in \mathbb{Z}_q$.

2. *Non-degeneracy:*

   - The result of the mapping is uniquely correspondence to any two points.

   - Example: $e(g_1, g_1) \neq e(g_2, g_2) \neq e(g_1, g_2) \neq 1$.

3. *Efficiently Computable: e can be easily computed given any two points* g_1 *and* g_2.

2.2.2 Discrete Logarithm Assumption

A discrete logarithm (DL) adversary is a randomized polynomial time algorithm $\mathcal{A}$ that gets input of a cyclic group $\mathbb{G}$ of prime order q from a DL key generator $\mathcal{K}_{dl}$ and (g, g^a) where $a \in \mathbb{Z}_q^*$ and $g \in \mathbb{G}$ are chosen at random. The discrete logarithm problem is to output a given g^a.

Definition 2.2. *The discrete logarithm assumption states that* $\mathcal{A}$ *wins if it can solve the discrete logarithm problem of* g^a. *The advantage of* $\mathcal{A}$ *denoted* $\mathbf{Adv}^{dl}_{\mathcal{K}_{dl},\mathcal{A}}$ *is the probability that* $\mathcal{A}$ *wins, taking into account the coins of* $\mathcal{A}$ *and* $\mathcal{K}_{dl}$ *throughout its invocation such that:*

$$\mathbf{Adv}^{dl}_{\mathcal{K}_{dl},\mathcal{A}} = \Pr[\mathcal{A}(\mathbb{G}, g, g^a) = a]$$

We say that $\mathcal{K}_{dl}$ *is secure if* $\mathbf{Adv}^{dl}_{\mathcal{K}_{dl},\mathcal{A}}$ *is negligible.*

2.2.3 One-More Discrete Logarithm Assumption

An one-more discrete logarithm (OMDL) adversary is a randomized polynomial time algorithm $\mathcal{A}$ that gets input of a cyclic group $\mathbb{G}$ of prime order q from a DL key generator $\mathcal{K}_{dl}$ and (g, g^a) where $a \in \mathbb{Z}_q^*$ and $g \in \mathbb{G}$ are chosen at random. $\mathcal{A}$ has access to two oracles, namely, an inverse discrete logarithm oracle $I\mathcal{NV}_{\mathcal{DL}}$ that given g^a returns a, and a challenge oracle $\mathcal{CHALL}$ that each time it is invoked, returns a random challenge value $R \in \mathbb{G}$. Let $R_1, \ldots, R_n$ denote the challenges returned by $\mathcal{CHALL}$.

Definition 2.3. *The one-more discrete logarithm assumption states that $\mathcal{A}$ wins if it can solve the discrete logarithm problem of every value R returned by $\mathcal{CHALL}$, with the restriction that the number of queries made by $\mathcal{A}$ to $I\mathcal{NV}_{\mathcal{DL}}$ oracle is strictly less than n. The advantage of $\mathcal{A}$ denoted $\mathbf{Adv}^{omdl}_{\mathcal{K}_{dl},\mathcal{A}}$ is the probability that $\mathcal{A}$ wins, taking into account the coins of $\mathcal{A}, \mathcal{K}_{dl}$ and $\mathcal{CHALL}$ throughout its invocation such that:*

$$\mathbf{Adv}^{omdl}_{\mathcal{K}_{dl},\mathcal{A}} = \Pr[\mathcal{A}(\mathbb{G}, g, g^a, \mathcal{CHALL}^n, I\mathcal{NV}_{\mathcal{DL}}{}^n, a_1, \ldots, a_n) = a]$$

We say that $\mathcal{K}_{dl}$ is secure if $\mathbf{Adv}^{omdl}_{\mathcal{K}_{dl},\mathcal{A}}$ is negligible.

2.2.4 RSA Assumption

A RSA adversary is a randomized polynomial time algorithm $\mathcal{A}$ that gets input of (N, e) from a RSA key generator $\mathcal{K}_{rsa}$ and $W \in \mathbb{Z}_N^*$ where W is a randomly chosen generator and $e \in \mathbb{Z}^*_{\phi(N)}$ is a random exponent. It is required

that GCD$(e,\phi(N))$=1 where ϕ is the Euler's totient function, or the solution $d = e^{-}1 mod \phi(N)$ of RSA problem does not exist.

Definition 2.4. *The RSA assumption states that $\mathcal{A}$ wins if it can solve the RSA problem of W. The advantage of $\mathcal{A}$ denoted $\mathbf{Adv}^{rsa}_{\mathcal{K}_{rsa},\mathcal{A}}$ is the probability that $\mathcal{A}$ wins, including the coins of $\mathcal{A}$ and $\mathcal{K}_{rsa}$ throughout its invocation such that:*

$$\mathbf{Adv}^{rsa}_{\mathcal{K}_{rsa},\mathcal{A}} = \Pr[\mathcal{A}(N,e,W) = W^d]$$

We say that $\mathcal{K}_{rsa}$ is secure if $\mathbf{Adv}^{rsa}_{\mathcal{K}_{rsa},\mathcal{A}}$ is negligible.

2.2.5 One-More RSA Assumption

An one-more RSA (OMRSA) adversary is a randomized polynomial time algorithm $\mathcal{A}$ that gets input of (N,e) from a RSA key generator $\mathcal{K}_{rsa}$ and has access to two oracles, namely, an inverse RAS oracle $I\mathcal{NV}_{\mathcal{RSA}}$ that given $W \in \mathbb{Z}^*_N$ returns W^d mod N, and a challenge oracle $\mathcal{CHALL}$ that each time it is invoked, returns a random challenge value $W \in \mathbb{Z}^*_N$. Let $W_1,\ldots,W_n$ denote the challenges returned by $\mathcal{CHALL}$.

Definition 2.5. *The one-more RSA assumption states that $\mathcal{A}$ wins if it can solve the RSA problem of every value W returned by $\mathcal{CHALL}$, with the restriction that the number of queries made to $I\mathcal{NV}_{\mathcal{RSA}}$ oracle is strictly less than n. The advantage of $\mathcal{A}$ denoted $\mathbf{Adv}^{omrsa}_{\mathcal{K}_{rsa},\mathcal{A}}$ is the probability that $\mathcal{A}$ wins, taking into account the coins of $\mathcal{A}$, $\mathcal{K}_{rsa}$ and $\mathcal{CHALL}$ throughout its invocation such that:*

$$\mathbf{Adv}^{omrsa}_{\mathcal{K}_{rsa},\mathcal{A}} = \Pr[\mathcal{A}(N,e,W,\mathcal{CHALL}^n, I\mathcal{NV}_{\mathcal{RSA}}{}^n, W_1,\ldots,W_n) = W^d]$$

We say that $\mathcal{K}_{rsa}$ is secure if $\mathbf{Adv}^{omrsa}_{\mathcal{K}_{rsa},\mathcal{A}}$ is negligible.

2.2.6 Computational Diffie-Hellman Assumption

An computational Diffie-Hellman (CDH) adversary is a randomized polynomial time algorithm $\mathcal{A}$ that gets input of a cyclic group $\mathbb{G}$ of prime order q from a CDH key generator $\mathcal{K}_{cdh}$ and $(g, g^a, h = g^b)$ where $a, b \in \mathbb{Z}_q^*$ and $g \in \mathbb{G}$ are chosen at random. The computational Diffie-Hellman problem is to output g^{ab} given (g, g^a, h).

Definition 2.6. *The computational Diffie-Hellman assumption states that $\mathcal{A}$ wins if it can solve the computational Diffie-Hellman problem of h. The advantage of $\mathcal{A}$ denoted $\mathbf{Adv}^{cdh}_{\mathcal{K}_{cdh},\mathcal{A}}$ is the probability that $\mathcal{A}$ wins, taking into account the coins of $\mathcal{A}$and$\mathcal{K}_{cdh}$ throughout its invocation such that:*

$$\mathbf{Adv}^{cdh}_{\mathcal{K}_{cdh},\mathcal{A}} = \Pr[\mathcal{A}(\mathbb{G}, g, g^a, h) = h^a]$$

We say that $\mathcal{K}_{cdh}$ is secure if $\mathbf{Adv}^{cdh}_{\mathcal{K}_{cdh},\mathcal{A}}$ is negligible.

2.2.7 One-More Computational Diffie-Hellman Assumption

An one-more computational Diffie-Hellman (OMCDH) adversary is a randomized polynomial time algorithm $\mathcal{A}$ that gets input of a cyclic group $\mathbb{G}$ of prime order q from a CDH key generator $\mathcal{K}_{cdh}$ and (g, g^a) where $a \in \mathbb{Z}_q^*$ and $g \in \mathbb{G}$ are chosen at random. $\mathcal{A}$ has access to two oracles, namely, an inverse computational Diffie-Hellman oracle $I\mathcal{NV}_{\mathcal{CDH}}$ that given g returns g^a, and a challenge oracle $\mathcal{CHALL}$ that each time it is invoked, returns a random challenge point $h \in \mathbb{G}$. Let $h_1, \ldots, h_n$ denote the challenges returned by $\mathcal{CHALL}$.

Definition 2.7. *The one-more computational Diffie-Hellman assumption states that $\mathcal{A}$ wins if it can solve the computational Diffie-Hellman problem of every value h returned by $\mathcal{CHALL}$, with the restriction that the number of queries made by $\mathcal{A}$ to its $\mathcal{INV}_{\mathcal{CDH}}$ oracle is strictly less than n. The advantage of $\mathcal{A}$ denoted $\mathbf{Adv}^{omcdh}_{\mathcal{K}_{cdh},\mathcal{A}}$ is the probability that $\mathcal{A}$ wins, taking into account the coins of $\mathcal{A}$, $\mathcal{K}_{cdh}$ and $\mathcal{CHALL}$ throughout its invocation such that:*

$$\mathbf{Adv}^{omcdh}_{\mathcal{K}_{cdh},\mathcal{A}} = \Pr[\mathcal{A}(\mathbb{G}, g, g^a, h, \mathcal{CHALL}^n, \mathcal{INV}_{\mathcal{CDH}}{}^n, h_1^a, \ldots, h_n^a) = h^a]$$

We say that $\mathcal{K}_{cdh}$ is secure if $\mathbf{Adv}^{omcdh}_{\mathcal{K}_{cdh},\mathcal{A}}$ is negligible.

2.2.8 Lagrange Coefficient

Let $q(\cdot)$ be a random $(d-1)$-degree polynomial and $q(i) = s_i$, the polynomial $q(\cdot)$ can be reconstructed by having the knowledge of d-pair values of (i_η, s_{i_η}):

$$q(\cdot) = \sum_{\eta=0}^{d-1} s_{i_\eta} \triangle_{i_\eta,S}(\cdot)$$

where $S = \{i_0, i_1, \ldots, i_{d-1}\}$.

Definition 2.8. *The Lagrange coefficient $\triangle_{i,S}$ is defined as:*

$$\triangle_{i,S}(x) = \prod_{j \in S, j \neq i} \frac{x-j}{i-j}$$

where i and j are elements in the set S.

2.2.9 Fuzzy Extractor

Dodis et al. (2004) introduced fuzzy extractor to extract a cryptographic key from fuzzy input such as images, videos, digital signals, analog signals and so on. As biometrics inputs are always fuzzy, fuzzy extractor frequently appears in biometrics and bio-crypto algorithms. Let b be the discretised value of a biometric feature ω and b' be the discretised value of the corresponding query biometric feature ω', the definition of fuzzy extractor is as follows.

Definition 2.9. *A fuzzy extractor consists of two randomized algorithms (***Gen**, **Rep***) ("generate" and "reproduce") (Dodis et al., 2004):*

1. **Gen** *takes enrolled biometric feature $\omega \in \mathbb{M}$ as input, where $\mathbb{M}$ is a metric space with distance function $dis(\cdot)$ whose threshold value is $(n-d)$, where n is the size of ω and d is the number of matched elements. It outputs a random string $R \in \{0,1\}^l$ and a helper string $PAR = b \oplus C_e(R)$ where C_e is an one to one encoding function.*

2. **Rep** *takes a query biometric feature $\omega' \in \mathbb{M}$ to compute $R' = C_d(b' \oplus PAR) = C_d(b \oplus b' \oplus C_e(R))$ and $R = R'$ if $dis(b,b') \leq (n-d)$. Here C_d is the decoding function that corrects the errors up to the threshold $(n-d)$.*

2.3 Identity-Based Signature Scheme

An identity-based signature (IBS) scheme $\mathcal{IBS}$ consists of four probabilistic polynomial time algorithms, namely, **Setup**, **Extract**, **Sign** and **Verify**.

1. **Setup.** Given a security parameter 1^k as input, generate the master public key mpk and master secret key msk.
2. **Extract.** Given a public identity ID together with mpk and msk as input, generate the user private key upk.
3. **Sign.** Given mpk, ID, a message m and upk as input, generate the signature σ.
4. **Verify.** Given the signature and message pair $\{\sigma, m\}$ as well as mpk and ID, outputs 1 (accept) or 0 (reject).

We say that an IBS scheme is correct if the **Verify** algorithm always outputs 1 for valid signatures.

2.3.1 Security Model

The existential unforgeability attack game on an IBS scheme between a forger $\mathcal{F}$ and a challenger $\mathcal{C}$ is described as a two-phased game (Kurosawa and Heng, 2004, Bellare et al., 2009) as follows:

1. **Setup.** $\mathcal{C}$ takes as input 1^k and runs the setup algorithm. It gives $\mathcal{F}$ the master public key mpk and keeps the master secret key msk to itself.

2. **Phase 1**

   (a) $\mathcal{F}$ issues some extract queries on $ID_1, ID_2, \ldots$. $\mathcal{C}$ responds by running the extract algorithm to generate the user private key upk_i corresponding to the public identity ID_i. It returns upk_i to $\mathcal{F}$.

   (b) $\mathcal{F}$ issues signing queries on messages $m_1, m_2, \ldots$ of its choice for the chosen message attack.

   (c) The queries in step (a) and step (b) above can be interleaved and asked adaptively. Without loss of generality, we may assume that $\mathcal{F}$ will not query m_i whose ID_i has been issued in the extract queries again.

3. **Phase 2**

   (a) $\mathcal{F}$ plays the role as a signer (forging attempt on the m_i holding the challenged identity ID' such that ID' is not issued an extract query before), trying to convince the verifier.

   (b) $\mathcal{F}$ can still issue some extract queries as in Phase 1.

Definition 2.10. *The advantage of $\mathcal{F}$ denoted $\mathbf{Adv}_{IBS,\mathcal{F}}^{euf-cma}$ is the probability that $\mathcal{F}$ runs time $t_{\mathcal{F}}$ and outputs a valid forgery in the environment set up by $\mathcal{C}$ such that:*

$$\mathbf{Adv}_{IBS,\mathcal{F}}^{euf\text{-}cma} = \Pr[\mathbf{Verify}_{\mathcal{C}} = 1]$$

We say that an IBS scheme is secure if $\mathbf{Adv}_{IBS,\mathcal{F}}^{euf\text{-}cma}$ is negligible for any $\mathcal{F}$ which runs in time $t_{\mathcal{F}}$.

2.4 Identity-Based Identification Scheme

An identity-based identification (IBI) scheme $\mathcal{IBI}$ consists of four probabilistic polynomial time algorithms, namely, **Setup**, **Extract** and **Identification Protocol** (proving and verification).

1. **Setup**. *PKG* takes as input the security parameter 1^k to generate the master public key mpk and the master secret key msk. The master public key will be publicly known while the master secret key will be known to the *PKG* only.

2. **Extract**. *PKG* takes as input mpk, msk and a public identity *ID* to extract a user private key upk.

3. **Identification Protocol ($\mathcal{P}$ and $\mathcal{V}$)**. $\mathcal{P}$ receives as input (mpk, upk, *ID*) and $\mathcal{V}$ receives as input (mpk, ID'), where upk is the user private key corresponding to the public identity *ID*. $\mathcal{P}$ and $\mathcal{V}$ run an interactive protocol which consists of the following steps:

   (a) **commitment**: $\mathcal{P}$ sends a commitment *CMT* to $\mathcal{V}$.

   (b) **challenge**: $\mathcal{V}$ sends a challenge *CH* to $\mathcal{P}$.

   (c) **response**: $\mathcal{P}$ sends a response *RSP* to $\mathcal{V}$.

   Finally, $\mathcal{V}$ outputs a boolean decision 1 (accept) or 0 (reject) based on *RSP*. A legitimate $\mathcal{P}$ should always be accepted.

2.4.1 Security Model

The impersonation attack game on an IBI scheme between an impersonator I and a challenger $\mathcal{M}$ is described as a two-phased game (Kurosawa and Heng, 2004, Bellare et al., 2009) as follows:

1. **Setup.** $\mathcal{M}$ takes as input 1^k and runs the setup algorithm. It gives I the resulting master public key mpk and keeps the master secret key msk to itself.
2. **Phase 1**
   (a) I issues some extract queries on $ID_1, ID_2, \ldots$. $\mathcal{M}$ responds by running the extract algorithm to generate the user private key upk_i corresponding to the public identity ID_i. It returns upk_i to I.
   (b) I issues some transcript queries for passive attack or some identification queries on ID_j for active/concurrent attack.
   (c) The queries in step (a) and step (b) above can be interleaved and asked adaptively. Without loss of generality, we may assume that I will not query the same ID_i that has been issued in the extract queries, in the transcript queries or identification queries again.
3. **Phase 2**
   (a) I plays the role as a cheating prover (impersonation attempt on the prover holding the challenged identity ID' such that ID' is not issued an extract query before), trying to convince the verifier.
   (b) I can still issue some extract queries as well as transcript queries or identification queries as in Phase 1.

Definition 2.11. *The advantage of I denoted* $\mathbf{Adv}_{IBI,I}^{imp-pa/aa/ca}$ *is the probability that I runs in time t_I and outputs a valid response in the environment set up by $\mathcal{M}$ such that:*

$$\mathbf{Adv}_{IBI,I}^{imp\text{-}pa/aa/ca} = \Pr[\mathbf{Verify}_{\mathcal{M}} = 1]$$

We say that an IBI scheme is secure if $\mathbf{Adv}_{IBI,I}^{imp\text{-}pa/aa/ca}$ *is negligible.*

2.5 Fuzzy Identity-Based Signature Scheme

A fuzzy identity-based signature (FIBS) scheme $\mathcal{FIBS}$ consists of four probabilistic polynomial time algorithms, namely, **Setup**, **Extract**, **Sign** and **Verify**.

1. **Setup.** Given a security parameter 1^k as input, generate the master public key mpk and master secret key msk.
2. **Extract.** Given a public identity set ID together with mpk and msk as input, generate the user private key upk.
3. **Sign.** Given mpk, ID, a message m and upk as input, generate the signature σ.
4. **Verify.** Given the signature and message pair $\{\sigma, m\}$ as well as mpk and ID', where $|ID \cap ID'| \geq d$, outputs 1 (accept) or 0 (reject).

We say that a FIBS scheme is correct if the **Verify** algorithm always outputs 1 for valid signatures.

2.5.1 Security Model

The cryptanalysis are performed based on the security model of FIBS, where the existential unforgeability attack game on a FIBS scheme between a forger $\mathcal{F}$ and a challenger $\mathcal{C}$ is described as a two-phased game (Yang, Cao and Dong, 2011) in the selective-ID model as follows:

1. **Init.** $\mathcal{F}$ declares the identity set ID' that it wishes to be challenged upon. Therefore, one identity set ID_i such that $|ID_i \cap ID'| \geq d$ will be under attack in the Phase 2 of the game.
2. **Setup.** $\mathcal{C}$ takes as input 1^k and runs the setup algorithm. It gives $\mathcal{F}$ the resulting master public key mpk and keeps the master secret key msk to itself.
3. **Phase 1**
   (a) $\mathcal{F}$ issues some extract queries on identity sets $ID_1, ID_2, \ldots$, where $|ID_i \cap ID'| < d$. $\mathcal{C}$ responds by running the extract algorithm to generate the user private key upk_i corresponding to the public identity ID_i. It returns upk_i to $\mathcal{F}$.
   (b) $\mathcal{F}$ issues signing queries on messages $m_1, m_2, \ldots$ of its choice on the identity set ID_i such that $|ID_i \cap ID'| < d$ for the chosen message attack.
   (c) The queries in step (a) and step (b) above can be interleaved and asked adaptively. Without loss of generality, we may assume that $\mathcal{F}$ will not query m_i whose ID_i has been issued in the extract queries

again.

4. **Phase 2**

   (a) $\mathcal{F}$ plays the role as a signer (forging attempt on the m_i holding the challenged identity set ID_i such that $|ID_i \cap ID'| \geq d$), trying to convince the verifier.

   (b) $\mathcal{F}$ can still issue some extract queries as in Phase 1.

Definition 2.12. *The advantage of $\mathcal{F}$ denoted $\mathbf{Adv}^{euf-cma}_{FIBS,\mathcal{F}}$ is the probability that $\mathcal{F}$ runs time $t_{\mathcal{F}}$ and outputs a valid forgery in the environment set up by $\mathcal{C}$ such that:*

$$\mathbf{Adv}^{euf\text{-}cma}_{FIBS,\mathcal{F}} = \Pr[\mathbf{Verify}_{\mathcal{C}} = 1]$$

We say that a FIBS scheme is secure if $\mathbf{Adv}^{euf\text{-}cma}_{FIBS,\mathcal{F}}$ is negligible for any $\mathcal{F}$ which runs in time $t_{\mathcal{F}}$.

2.6 Fuzzy Identity-Based Identification Scheme

A fuzzy identity-based identification (FIBI) scheme $\mathcal{FIBI}$ consists of four probabilistic polynomial time algorithms, namely, **Setup**, **Extract**, **Identification Protocol** (proving and verification).

1. **Setup**. *PKG* takes as input the security parameter 1^k to generate the master public key mpk and the master secret key msk. The master public key is publicly known while the master secret key is known to the *PKG* only.

2. **Extract**. *PKG* takes as input mpk, msk and a public identity set ID to extract a user private key upk.

3. **Identification Protocol ($\mathcal{P}$ and $\mathcal{V}$).** $\mathcal{P}$ receives as input (mpk,upk,ID) and $\mathcal{V}$ receives as input (mpk, ID'), where $|ID \cap ID'| \geq d$ and upk is the private key corresponding to the public identity set ID. $\mathcal{P}$ and $\mathcal{V}$ run an interactive protocol which consists of the following steps:

   (a) **commitment**: $\mathcal{P}$ sends a commitment CMT to $\mathcal{V}$.

   (b) **challenge**: $\mathcal{V}$ sends a challenge CH to $\mathcal{P}$.

   (c) **response**: $\mathcal{P}$ sends a response RSP to $\mathcal{V}$.

   Finally, $\mathcal{V}$ outputs a boolean decision 1 (accept) or 0 (reject) based on RSP. A legitimate $\mathcal{P}$ should always be accepted.

2.7 Hamming Distance

Hamming distance is the difference in number of bits between a binary string A and a binary string B:

Definition 2.13.

$$distance = \sum_{i=0}^{i=|A \oplus B|} bit_i$$

where $bit_i \in \{1,0\}$ *is the bit value of the binary string* $\{A \oplus B\}$ *at location i.*

2.8 Biometric Performance Metrics

In biometrics, a prover is authenticated by the verifier if the prover's query biometrics matches to the corresponding enrolled biometrics. Such matching is determined by a matching score or threshold which is identified by using

several benchmarking techniques such as the false rejection rate (FRR), false acceptance rate (FAR) and equal error rate (EER). In this project, only FRR and FAR will be used but not EER as the former indicate the later where EER is the rate when FRR equals to FAR.

Definition 2.14. *FRR refers to the probability that the system fails to detect a match between the input pattern and a matching template in the database. It measures the percent of valid inputs which are incorrectly rejected:*

$$FRR = \frac{number\ of\ rejected\ genuine\ user}{total\ number\ of\ genuine\ access}$$

Definition 2.15. *FAR refers to the probability that the system incorrectly matches the input pattern to a non-matching template in the database. It measures the percent of invalid inputs which are incorrectly accepted:*

$$FAR = \frac{number\ of\ accepted\ impersonator}{total\ number\ of\ impersonator\ access}$$

CHAPTER 3

CRYPTANALYSIS ON FUZZY IDENTITY-BASED SCHEMES

In this chapter, we present the cryptanalysis results of FIBE, FIBS as well as FIBI schemes and then select the most suitable candidate for the focus of this work later on: authentication mechanism. Firstly, we reveal the implementation issues in the variant of FIBE, namely, biometric-identity-based encryption (Bio-IBE) and provide a solution. Secondly, two FIBS schemes are cryptanalysed and the problematic algorithms were identified. Thirdly, due to the scarcity of FIBI scheme, we focus the research on IBI schemes, the primitive which is closely related to FIBI. We cryptanalysed an efficient IBI scheme secure in the standard model to date and have it fixed after that. Based on these cryptanalysis results, we select the optimum candidate for security enhancements in the subsequent work.

3.1 Fuzzy IBE and Bio-IBE Schemes

There are two approaches in integrating biometrics and cryptography: a) FIBE schemes (Sahai and Waters, 2005) and b) fuzzy extractor (FE) (Dodis et al., 2004). FE allows error correction and generates strong cryptographic keys from noisy data such as biometric. The cryptographic key in FE refers to the private key used in some encryption schemes, such as AES, instead of public key encryption scheme such as FIBE. Inspired by FE and FIBE, a new public

key encryption scheme, namely, biometric identity-based encryption (Bio-IBE) was proposed by Sarier (2008) and a few biometric identity-based encryption (Bio-IBE) schemes (Sarier, 2009, 2010*b*, Shi et al., 2010, Sarier, 2011, Yang, Hu, Zhang and Sun, 2011) appeared in the literature after that. In general, Bio-IBE uses FE to generate user public identity $ID = Hash(\omega)$, where ω is the set of extracted biometric features such that $\omega = \{\mu_1, \mu_2, \cdots, \mu_n\}$. To avoid collusion attack, Bio-IBE either applies LP during encryption (Sarier, 2008, 2009, 2010*b*, 2011, Yang, Hu, Zhang and Sun, 2011) or during user secret key extraction (Shi et al., 2010).

Bio-IBE by Sarier (2008) works slightly different in such a way that it applies LP (in addition to FE) during *upk* extraction and decryption. Since FE is applied, the errors of biometric features will be corrected and thus always the same user public identity ID is generated for Bio-IBE. The condition to activate the correction, however, similar to FIBE where the error correction is performed only if $|\omega \cap \omega'| \geq d$ for a threshold value d, where ω is the enrolled biometric data and ω' is the query biometric data.

The difference between the techniques of Lagrange polynomial (LP) and fuzzy extractor (FE) is that the former tolerates errors while the latter corrects errors. Thus, it is a redundancy to use both together in Bio-IBE scheme where no errors exist for LP to tolerate with, as all the errors have been corrected by FE and vice versa. In fact, the basic construction of FE is error correction code which is using polynomial also. For instance, error correction codes such as

BCH codes are applied on the resulting points (of the biometric data ω on LP) into a single value *ID*.

In this section, we revise the algorithm flow of Bio-IBE in order to rule out the unrealistic requirement of encryption. Furthermore, we reveal that the use of Lagrange polynomial and fuzzy extractor together in Bio-IBE is inappropriate as it increases the algorithm complexity without gaining any benefit. As such, we propose a redundancy removal technique for a Bio-IBE (Sarier, 2008) scheme and show that the security is not compromised. This technique can be applied on other Bio-IBE (Sarier, 2009, 2010*b*, Shi et al., 2010, Sarier, 2011, Yang, Hu, Zhang and Sun, 2011) schemes as well.

3.1.1 Sarier's Bio-IBE

We outline Sarier's Bio-IBE (Sarier, 2008) as follows before showing the redundancy of FE or LP in the scheme:

Setup(k): On input of 1^k, PKG generates a group $\mathbb{G}$ of prime order q before picking a random generator $g \in \mathbb{G}$, a random value $x \in \mathbb{Z}_q^*$ to compute $P_{pub} = g^x$ and $e(g,g)$. PKG defines for fuzzy extractor the encoding function C_e and the decoding function C_d along with the feature extraction method F_e which produces biometric feature b. PKG also defines three cryptographic hash functions $H : b \rightarrow \{0,1\}^*$,$H_1 : \mathbb{Z}_q^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ and $H_2 : \mathbb{G}_T \rightarrow \{0,1\}^l$. Let d be the threshold of FE and t be the threshold of LP, the master public key

is $mpk = (q, g, e, d, t, l, \mathbb{G}, \mathbb{G}_T, H, H_1, H_2, P_{pub}, e(g,g), C_e, C_d, F_e)$ and the master secret key is $msk = x$.

Extract(msk, ω, b): An enrolled biometric data ω is obtained from a biometric reader and the feature extractor F_e. Each element $\mu_i \in \omega$ is associated to a unique integer in $\mathbb{Z}_q^*$. Let b be the binarised value of ω, PKG takes msk as input to compute the biometric identity, $ID = H(b)$ and the corresponding helper string $PAR = \mathbf{Gen}(b, ID)$ using FE. Next, PKG calculates $D_{\mu_i}^{ID} = g^{1/(x+H_1(\mu_i, ID'))} = g^{1/(x+h_i^{ID})}$ for each $\mu_i \in \omega$ and assigns $usk = \{D_{\mu_i}^{ID}\}$ to user.

Encrypt(mpk, ω', b', M): Encrypter takes mpk, query biometric data ω', PAR and a plaintext $\mathcal{M}$ as input, calculate $ID' = \mathbf{Rep}(b', PAR) = ID$ where b' is the binarised value of ω'. Encrypter picks a random $(d-1)$-degree polynomial $r(\cdot)$ such that $r(0) = s \in \mathbb{Z}_q^*$ to compute the shares $r(\mu_i) = r_i \in \mathbb{Z}_q^*$ and $L_i = P_{pub} \cdot g^{H_1(\mu_i, ID')} = g^{(x+h_i^{ID})}$ for $\mu_i \in \omega'$. The ciphertext is set to be $C = (\omega', L_i^{r_i}, W)$ where $W = M \oplus V$ such that $V = H_2(e(g,g)^s)$.

Decrypt(mpk, usk, C): Given $C = (\omega', L_i^{r_i}, W)$, an arbitrary set $S \subseteq \omega \cap \omega'$ is selected such that $|S| = d$. For every $\mu_i \in S$, $h_i^{ID} = h_i^{ID'}$ and the plaintext $M = W \oplus V$ can be recovered as follows:

$$
\begin{aligned}
V &= H_2\left(\prod_{\mu_i}\left(e\left(L_i^{r_i}, D_{\mu_i}^{ID}\right)\right)^{\triangle_{\mu_i,S}(0)}\right) \\
&= H_2\left(\prod_{\mu_i}\left(e\left(g^{r_i(x+h_i^{ID'})}, g^{1/(x+h_i^{ID})}\right)\right)^{\triangle_{\mu_i,S}(0)}\right) \\
&= H_2\left(\prod_{\mu_i}\left(e(g,g)^{r_i}\right)^{\triangle_{\mu_i,S}(0)}\right) \\
&= H_2\left(e(g,g)^{s}\right)
\end{aligned}
$$

The flow diagrams for **Extract** and **Encrypt** algorithms of Bio-IBE above are as shown in Figure 3.1.

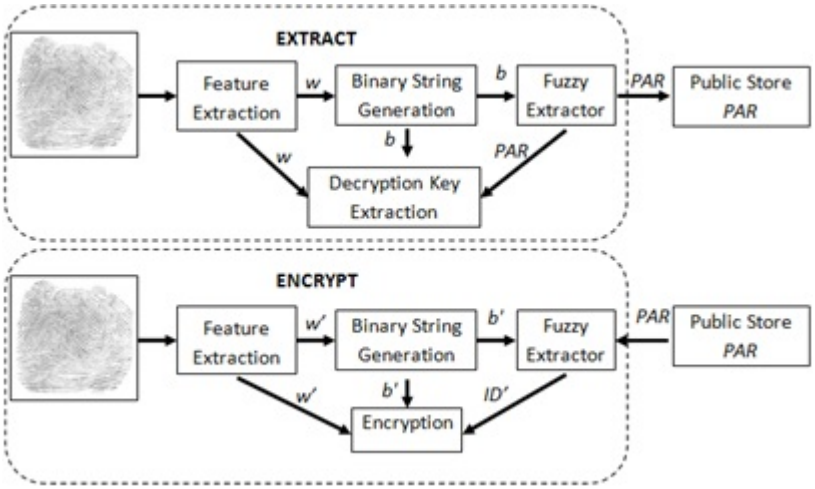


Figure 3.1: Flow Diagrams of Sarier's Bio-IBE

3.1.2 Problems in Algorithm Flow

In the flow diagram of Sarier's Bio-IBE, an impractical assumption is made whereby a fresh biometric reading ω' must be obtained for every encryption. This means that the decrypter always has to ready in providing a fresh biometric reading ω'. The encrypter then get the helper string **PAR** from public storage and run the reproduce algorithm **Rep** to compute ID' for use in encryp-

tion process as depicted in the Figure 3.1. In short, the decrypter has to be prepared with a biometric reader in hand as well as internet connection whenever the encrypter wants to generate a ciphertext to him.

To solve this problem, we suggest to store together in the public storage, the enrolled biometric reading ω used in **Extract** algorithm and the corresponding **PAR** so that anyone can send a ciphertext without being restricted to the availability of the decrypter's biometric. Besides, during the **Decrypt** algorithm, a decrypter must produce a query biometric data to claim the ownership of the corresponding upk. The revised algorithm flow is as shown in Figure 3.2.

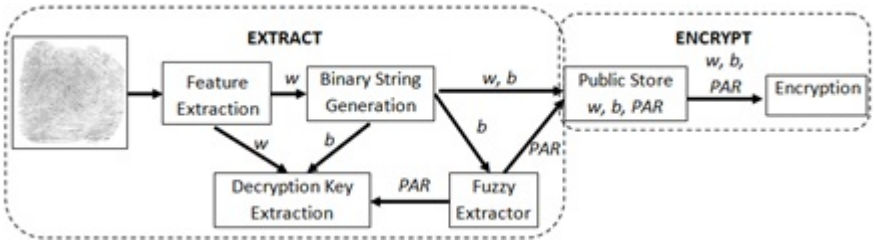


Figure 3.2: Revised Algorithm Flow of Sarier's Bio-IBE

However, redundancy still exists in the algorithms though the flow is corrected. Next, we show how to remove the redundancy and discuss the solution for privacy issue of public storage in the following sections.

3.1.3 Redundancy of Fuzzy Extractor

We show that when the FE is removed from Sarier's Bio-IBE, the resulting scheme resembles Sakai-Kasahara FIBE (SK-FIBE) scheme, which is the SK-IBE (Sakai and Kasahara, 2003, Chen and Cheng, 2005) embedded with

Lagrange polynomial in the **Extract** and **Decrypt** algorithms:

Setup(k): Similar to the algorithm in Section 3.1.1 without the FE parameters.

Extract(msk, ω, b): The involvement of FE is excluded. An enrolled biometric data ω is obtained from the raw biometric information using a reader and the feature extractor F_e. Each element $\mu_i \in \omega$ is associated to a unique integer in $\mathbb{Z}_q^*$. Let b be the binarised value of ω, PKG takes msk as input to compute the biometric identity, $ID = H(b)$ and calculate $D_{\mu_i}^{ID} = g^{1/(x+H_1(\mu_i, ID))} = g^{1/(x+h_i^{ID})}$ for each $\mu_i \in \omega$. PKG returns $usk = \{D_{\mu_i}^{ID}\}$ to user.

Encrypt(mpk, ω, b, M): Let $ID = H(b)$ and exclude the involvement of FE, encrypter gets ω and b from public storage. Encrypter picks a random $(t-1)$-degree polynomial $r(\cdot)$ such that $r(0) = s \in \mathbb{Z}_q^*$ to compute the shares $r(\mu_i) = r_i \in \mathbb{Z}_q^*$ and $L_i = P_{pub} \cdot g^{H_1(\mu_i, ID)} = g^{(x+h_i^{ID})}$ for $\mu_i \in \omega$. The ciphertext is set to be $C = (\omega, L_i^{r_i}, W)$ and $W = M \oplus V$ such that $V = H_2(e(g,g)^s)$.

Decrypt(mpk, usk, C): Given a ciphertext $C = (w, L_i^{r_i}, W)$, an arbitrary set $S \subseteq \omega \cap \omega'$ is selected such that $|S| = t$, where ω' is the query biometric data. For every $\mu_i' \in S$, $H_1(\mu_i', ID) = h_i'^{ID} = h_i^{ID}$ and the plaintext $M = W \oplus V$ can be recovered as in the algorithm of Section 3.1.1:

$$
\begin{aligned}
V &= H_2\left(\prod_{\mu_i}\left(e\left(L_i^{r_i}, D_{\mu_i}^{ID}\right)\right)^{\triangle_{\mu_i,S}(0)}\right) \\
&= H_2\left(\prod_{\mu_i}\left(e\left(g^{r_i(x+h_i^{ID'})}, g^{1/(x+h_i^{ID})}\right)\right)^{\triangle_{\mu_i,S}(0)}\right) \\
&= H_2\left(\prod_{\mu_i}\left(e(g,g)^{r_i}\right)^{\triangle_{\mu_i,S}(0)}\right) \\
&= H_2\left(e(g,g)^{s}\right).
\end{aligned}
$$

If the query biometric data is obtained from the genuine user, together with the valid upk, the decryption is always success with maximum $|\omega| - t$ errors tolerated by the LP.

Recall that the role of FE is to hide a secret random string R using biometric data ω. With the removal of FE, R is now replaced with a public computable hash value $H(b) = ID$. Although this may create privacy issue by publicising biometric identity, it fits the concept of Bio-IBE and FIBE, where ID is a public key and it is expected to be publicly known. Thus, the removal of FE does not affect the security because it only generates a unique ID which is bond to the corresponding b. The amendments show that H alone is sufficient for the job whereby the generation of unique ID can be replaced by computing $ID = H(b)$. Furthermore, the security proof in Sarier (2008) involves only the hash function H but not FE. The errors between b and b' are not an issue as the errors can be tolerated by LP during **Encrypt** and **Decrypt**.

3.1.4 Redundancy of Lagrange Polynomial

We now show that on the other hand, if LP is removed from Bio-IBE instead of FE, the resulting scheme resembles SK-IBE (Sakai and Kasahara, 2003, Chen and Cheng, 2005) whose ID is bond with FE:

Setup(k): Similar to the algorithm in Section 3.1.1 without LP parameters.

Extract(msk, b): An enrolled biometric data ω is obtained from the raw biometric information using a reader and the feature extractor F_e. Each element $\mu_i \in \omega$ is associated to a unique integer in $\mathbb{Z}_q^*$. Let b be the binarized value of ω, PKG takes msk as input to compute the biometric identity $ID = H(b)$ and the corresponding helper string $PAR = \mathbf{Gen}(b, ID)$ using FE. Next, calculate the $usk = D_{ID} = g^{1/(x+H_1(ID))} = g^{1/(x+h_{ID})}$. PKG returns upk to user.

Encrypt(mpk, b, M): Encrypter takes mpk and plaintext $\mathcal{M}$ as input and gets ω and b from public storage. Encrypter then picks a random $s \in \mathbb{Z}_q^*$ to compute the value $L = P_{pub} g^{H_1(ID)} = g^{(x+h_{ID})}$ where $ID = H(b)$. The ciphertext is set as $C = (\omega, L^s, W)$, where $W = M \oplus V$ and $V = H_2(e(g,g)^s)$.

Decrypt(mpk, usk, C): Given $C = (\omega, L^s, W)$, $ID' = \mathbf{Rep}(b', PAR) = ID$ is computed where b' is the query biometric data of the decrypter and $dis(b, b') \leq (n-d)$. The plaintext $M = W \oplus V$ can be obtained similar to the algorithm in

Section 3.1.1:

$$
\begin{aligned}
V &= H_2(e(L^s, D_{ID})) \\
&= H_2(e(g^{s(x+H_1(ID'))}, g^{1/(x+H_1(ID))})) \\
&= H_2(e(g^{s(x+h_{ID'})}), g^{1/(x+H_1(ID))})) \\
&= H_2(e(g,g)^s).
\end{aligned}
$$

If the query biometric is obtained from the genuine user, together with the valid upk, the decryption is success with maximum d errors are corrected by the FE.

The SK-IBE scheme presented in Chen and Cheng (2005) is the stronger version (chosen ciphertext attack secure) of the original version (Sakai and Kasahara, 2003) (chosen plaintext attack secure) using Fujisaki-Okamoto transformation (Fujisaki and Okamoto, 2013). This transformation requires two extra hash functions and notice that if these hash functions in Chen and Cheng (2005) are removed, its **Encrypt** and **Decrypt** algorithms are the same as the one given in this section. Although such removal yield a chosen plaintext attack secure SK-IBE only, we argue that the original Bio-IBE is also proven secure under this security level (Sarier, 2008). The only difference is that the former views user public identity as a string value, while the latter views it as a list of string value. Hence, the removal of LP does not affect Bio-IBE's security as b of every user is corrected and resulted in a single value by FE, which reverts the Bio-IBE back to its building block, SK-IBE. Some may question that the binding of b to ID in FE instead of binding to upk will threaten the security of **Decrypt** algorithm. Recall that publishing ID only creates privacy issue, which is not

related to the cryptographic security of the underneath encryption scheme. The decryption will fail as long as upk is out of the reach of attacker.

3.1.5 Discussion

Since the main ingredient of error encoding code is Lagrange polynomial, to ease our calculation, we assume the complexity of **Gen** and **Rep** are the same as computing a polynomial and the Lagrange coefficient respectively. We use the operations timing from Tan et al. (2010) and represent them using the notations as shown in Table 3.1.

Table 3.1: Notation for operations timing

Notation	Description
T_e	Exponentiation in $\mathbb{G}$
T_p	Pairing in $\mathbb{G}$
T_a	Addition in $\mathbb{Z}_q$
T_i	Inversion in $\mathbb{Z}_q$
T'_e	Exponentiation in $\mathbb{G}_T$
T_h	Hashing to $\mathbb{Z}_q$
T_m	Multiplication in $\mathbb{G}$
F_e	Biometrics feature extraction
T_{Lagr}	Computation of Lagrange coefficient
T'_m	Multiplication in $\mathbb{G}_T$
t	Biometrics' integer threshold
T_{poly}	Compute $t-1$ degree polynomial $r(x)$ in $\mathbb{Z}_q$

Let $d = t = 100$ with $|\omega| = n > 100$, we can calculate the complexity ratio of IBE-FE's algorithms with respect to Bio-IBE[1] as follows:

[1]Since Bio-IBE is used as the base for comparison, its complexity is set to 1

Setup:

$$\frac{1}{R} = \frac{T_e + T_p}{T_e + T_p}$$

$$R = 1$$

Extract:

$$\frac{1}{R} = \frac{n(T_e + T_i + T_a + T_h) + F_e + T_{poly}}{T_e + T_i + T_a + T_h + F_e + T_{poly}}$$

$$R = \frac{T_e + T_i + T_a + T_h + F_e + T_{poly}}{n(T_e + T_i + T_a + T_h) + F_e + T_{poly}}$$

Let $x = (T_e + T_i) \approx F_e \approx T_{poly}$:

$$R \approx \frac{x + 0 + 0 + x + x}{n(x + 0 + 0) + x + x}$$

$$R = 3x/nx = 3/n$$

Encrypt:

$$\frac{1}{R} = \frac{n(2T_e + T_m) + T_{poly} + T_h + T_e' + F_e + T_{Lagr}}{2(T_e + T_h) + T_m + T_a + T_e'}$$

Let $x = F_e \approx T_{poly} \approx T_e \approx T_{Lagr}$:

$$R \approx \frac{2(T_e + 0) + 0 + 0 + 0}{n(2x + 0) + x + 0 + 0 + x + x}$$

$$R \approx \frac{2x}{2nx + 3x} = 2/2n + 3$$

Decrypt:

$$\frac{1}{R} = \frac{t(T_e' + T_p) + T_h + (t-1)T_m' + T_{Lagr}}{T_e + T_h + T_p + F_e + T_{Lagr}}$$

Let $x = F_e \approx T_e \approx T_{Lagr}$ and $T_p \approx 2T_e$:

$$R \approx \frac{x + 0 + 2x + x + x}{t(0 + 2x) + 0 + (t-1)0 + x}$$

$$R \approx 5x/(2t + 1)x = 0.02$$

Repeating the similar calculations for IBE-LP, the results are summarized in Table 3.2.

Table 3.2: Complexity of Bio-IBE, IBE+LP and IBE+FE

		Bio-IBE	IBE+LP	IBE+FE
Size of *upk*		$n\|\mathbb{G}\|$	$n\|\mathbb{G}\|$	$\|\mathbb{G}\|$
Size of C		$n\|\mathbb{G}\|+1$	$n\|\mathbb{G}\|+1$	$\|\mathbb{G}\|+1$
Complexity	Setup	T_e+T_p	T_e+T_p	T_e+T_p
	Extract	$n(T_e+T_i+T_a+T_h)+F_e+T_{poly}$	$n(T_e+T_a+T_i)+(n+1)T_h+F_e$	$T_e+T_i+T_a+T_h+F_e+T_{poly}$
	Encrypt	$n(2T_e+T_m)+T_{poly}+T_h+T'_e+F_e+T_{Lagr}$	$n(2T_e+T_m)+T'_m+(n+1)T_h+T_{poly}+T'_e$	$2(T_e+T_h)+T_m+T_a+T'_e$
	Decrypt	$t(T'_e+T_p)+T_h+(t-1)T'_m+T_{Lagr}$	$t(T'_e+T_p)+T_h+(t-1)T'_m+F_e+T_{Lagr}$	$T_e+T_h+T_p+F_e+T_{Lagr}$
Ratio	Setup	1	1	1
	Extract	1	$n+1/n+2$	$3/n$
	Encrypt	1	$2n+1/2n+3$	$2/2n+3$
	Decrypt	1	1	0.02

Table 3.2 shows that the complexity of the original Bio-IBE is greatly reduced when LP is removed. The puzzle left now is to decide whether to remove LP or FE if the privacy issue as well as the provable security of biometric data are concerned.

Recall that biometric data is used as public key in Bio-IBE and its operation is not affected if a user's biometric data is known through the public storage. In fact, Bio-IBE does not work at all if one's biometric data is not publicised. However, for the completeness of security, we do consider the pro-

tection of biometric data in the public storage. This problem is resolvable using biometric template protection techniques, which can be classified into four main categories, namely, biometric salting, non-invertible transform, key binding and key generation (Jain et al., 2008). Biometric salting and non-invertible transform can be generalised as feature transformation (Teoh et al., 2006, Jin et al., 2010, Ahmad et al., 2011) whereas key binding and key generation fall into the category of bio-cryptography (Xi and Hu, 2009, 2010, Xi et al., 2011). In order to further reduce users' privacy invasion concern, the preferable types of user biometric should be those that can be acquired easily such as fingerprint, palmprint, face and voice. In short, the employment of template protection techniques should be included as a basic requirement of Bio-IBE and FIBE.

Back to Table 3.2, we can see that IBE+FE is faster than IBE+LP in all algorithms. Furthermore, FE itself is already a template protector whereby given the public parameter *PAR*, it is infeasible to reverse engineer *PAR* to the biometric template ω. However, all the secret parameters of FE has to be made public for the purpose of fuzzily regenerating the same public key *ID* and this renders the template protection feature of FE useless. Similar to IBE+FE, IBE+LP does not come with any protection mechanism for their biometric template. Hence, IBE+FE is no better than IBE+LP in handling the user privacy issue.

The missing of template protection in IBE+FE and IBE+LP can be solved by applying template protection techniques. Since the core engine of FE and LP are polynomials, it implies that the cryptography constraints projected on bio-

metrics identity are the same to that of FIBE scheme. The resulted biometric trait ω from template protection scheme must be in an ordered sequence and have a fixed length in which the elements are either in integers or binary form. The index of each $\mu_i \in \omega$ will be used to generate the random polynomial $r(\cdot)$ during encryption while the binarised value b will be fed into H to compute $H(b) = ID$. The implementation issues will be discussed in details in Chapter 5. The properties of each type of Bio-IBE are as depicted in Table 3.3 and obviously, IBE+FE is the best for efficiency while IBE+LP is the best for security.

Table 3.3: Properties of Bio-IBE, IBE+LP and IBE+LP

Scheme	Bio-IBE	IBE+LP	IBE+FE
Avoid Collusion Attack	Bind ω to FE+LP	Bind ω to LP	Bind ω to FE
Redundancy	Yes	No	No
Require fresh ω to encrypt?	Yes	No	No
Constraint of Biometrics Data	Ordered, Fixed-Length, Binary/Integer	Ordered, Fixed-Length, Binary/Integer	Ordered, Fixed-Length, Binary/Integer
Algorithm Complexity	Highest	Medium	Lowest
Privacy Issue of ω	Not a concern	Protect ω with feature transformation	Protect ω with feature transformation
Provable Security	Yes	Yes	No

3.2 Fuzzy Identity-Based Signature Schemes

In 2009, Wang and Kim (2009) proposed a new fuzzy identity-based signature (FIBS) scheme and proved its existentially unforgeability under chosen message attack and fuzzy identity attack in the random oracle model if the

discrete logarithm problem is hard. In the same year, Chen, Zhu, Cao and Geng (2009) proposed a fuzzy identity-based signature with dynamic threshold which is proven secure against unforgeability in the standard model if the multi-sequence of Diffie-Hellman exponents problem is hard. In this section, we falsify the security claim of Wang and Kim's FIBS and Chen et al.'s FIBS by presenting a key only attack and a collusion attack on their schemes respectively. In precise, we show that the upk is not randomised in the former FIBS and so an adversary can forge a upk which has the same distribution as the genuine upk. We then show that the misbehaved users in the latter FIBS can combine their upks in such a way that they can generate a valid signature and this is impossible without combining their upks. In other words, the PKG will be impersonated without being noticed by any user in the system except the PKG itself.

3.2.1 Security Model

The strongest security notion is existential unforgeability against chosen message attack as described in Section 2.5.1 but we show that the weaker notions, namely, total break against key only attack and total break against collusion attack are sufficient for the cryptanalysis. The security games for these two security notions in the selective-ID model are described as follows:

1. **Init.** $\mathcal{F}$ declares the identity set ID' that it wishes to be challenged upon. Therefore, one identity set ID_i such that $|ID_i \cap ID'| \geq d$ will be under attack in the Phase 2 of the game.

2. **Setup.** $\mathcal{C}$ takes as input 1^k and runs the setup algorithm. It gives $\mathcal{F}$ the resulting master public key mpk and keeps the master secret key msk to itself.

3. **Phase 1**

   (a) In collusion attack, $\mathcal{F}$ issues some extract queries on identity sets $ID_1, ID_2, \ldots$, where $|ID_i \cap ID'| < d$. $\mathcal{C}$ responds by running the extract algorithm to generate the user private key upk_i corresponding to the public identity ID_i. It returns upk_i to $\mathcal{F}$.

   (b) In key only attack, $\mathcal{F}$ is only allowed to query the pubic identity ID_i of user i of its choice.

4. **Phase 2**

   (a) $\mathcal{F}$ plays the role as a signer (forging attempt on a random m given by $\mathcal{C}$ using the challenged identity set ID_i such that $|ID_i \cap ID'| \geq d$), trying to convince the verifier.

   (b) $\mathcal{F}$ can still issue some extract queries as in Phase 1.

Definition 3.1. *The advantage of $\mathcal{F}$ denoted $\mathbf{Adv}_{FIBS,\mathcal{F}}^{euf-koa/cola}$ is the probability that $\mathcal{F}$ runs time $t_{\mathcal{F}}$ and outputs a valid forgery in the environment set up by $\mathcal{C}$ such that:*

$$\mathbf{Adv}_{FIBS,\mathcal{F}}^{euf\text{-}koa/cola} = \Pr[\mathbf{Verify}_{\mathcal{C}} = 1]$$

We say that a FIBS scheme is secure if $\mathbf{Adv}_{FIBS,\mathcal{F}}^{euf\text{-}koa/cola}$ is negligible for any $\mathcal{F}$ which runs in time $t_{\mathcal{F}}$.

3.2.2 Cryptanalysis on the Wang and Kim's FIBS Scheme

In this section, we briefly describe the Wang and Kim's FIBS scheme and subsequently present the key only attack on it.

3.2.2.1 The Wang and Kim FIBS Scheme

We describe the scheme by Wang and Kim (2009) as follows:

Setup(1^k). The PKG chooses two distinct groups $\mathbb{G}$ and $\mathbb{G}_T$ of prime order q such that a bilinear pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ can be constructed, and P is a generator of $\mathbb{G}$. The PKG also chooses a hash function $H : \{0,1\}^* \times \mathbb{G}_T \to \mathbb{Z}_q^*$, which is assumed to be the random oracle. Next, the PKG defines the error tolerance parameter d. Finally, the PKG outputs a public parameters $PP = (\mathbb{G}, \mathbb{G}_T, q, e, P, P_{pub} = sP, d, H)$ and secretly keeps master secret key $msk = s$.

Extract(PP, msk, ID). For an identity ID which is described as an attribute set $\omega = (\omega_i)_{i=1}^n$, where $\omega_i \in \mathbb{Z}_q^*$, the PKG randomly picks a $(d-1)$-degree polynomial $p(x) \in \mathbb{Z}_q[x]$ such that $p(0) = s$ and computes the corresponding private key components $D_i = p(\omega_i)P \in \mathbb{G}$ for $i = 1, 2, \ldots, n$, then the PKG sends $D_{ID} = (D_1, D_2, \ldots, D_n)$ to the user with identity ID. The user can validate the correctness of the private key components by choosing an arbitrary d-elements subset S of ω and checking whether:

$$\sum_{\omega_i \in S} D_i \triangle_{\omega_i, S}(0) = P_{pub}.$$

Sign(PP, D_{ID}, m)**.** Given public parameters PP, the private key D_{ID}, which with respect to the identity ID described as the attribute set ω and a message $\mathcal{M}$, the signing procedure is performed as follows:

- The signer randomly chooses $t \in \mathbb{Z}_q^*$, computes $r = e(P,P)^t \in \mathbb{G}_T$ and $h = H(m,r) \in \mathbb{Z}_q^*$.
- Chooses a $(d-1)$-degree random polynomial $f(x)$ such that $f(0) = t$.
- Computes $V_i = f(\omega_i)P + hD_i$ for $i = 1, 2, \ldots, n$.
- The resulting signature is $\sigma = (ID, r, \{V_i\}_{i=1}^n)$.

Verify(PP, ID', m, σ)**.** To verify a signature $\sigma = (ID, r, \{V_i\}_{i=1}^n)$ with respect to the identity ID described as the attribute set ω against an identity ID' described as an attribute set η, where $|\omega \cap \eta| \geq d$, the verifier chooses an arbitrary d-element subset S of $\omega \cap \eta$, computes $h = H(m,r)$ and verifies that

$$\prod_{\omega_i \in S} e(V_i, P)^{\triangle_{\omega_i, S}(0)} = e(P, P_{pub})^h r.$$

3.2.2.2 Key Only Attack

In the FIBS environment, the PKG runs **Setup** algorithm and broadcasts the public parameters and user public keys to its network. An adversary $\mathcal{A}$ can thus easily obtain the public parameter $P_{pub} = sP$, d and the biometric identity (user public key) $\omega = (\omega_i)_{i=1}^n$ of a user Bob and subsequently forge Bob's private key D_{Bob} as follows:

1. $\mathcal{A}$ chooses random $c_j \in \mathbb{Z}_q$ for $1 \leq j \leq d-1$ as the coefficients which are used to construct the $(d-1)$-degree polynomial $p(\cdot)'$ implicitly.
2. $\mathcal{A}$ calculates $D'_i = P_{pub} + c_1\omega_i^1 \cdot P + c_2\omega_i^2 \cdot P + \ldots + c_{d-1}\omega_i^{d-1} \cdot P$ for $i = 1, 2, \ldots, n$.
3. $\mathcal{A}$ lets $D_{Bob} = (D'_1, D'_2, \ldots, D'_n)$.

Since the forged Bob's private key is having the same distribution with the one generated by the PKG, it is obvious that the correctness is always sound:

$$\sum_{\omega_i \in S} \triangle_{\omega_i,S}(0) D'_i = P_{pub}.$$

The signature σ on any message $\mathcal{M}$ that is signed using D_{Bob} is verified successfully due to the same reason also. $\mathcal{A}$ can even create a user private key D_{Alice} for the user Alice who never exists in the system by running the above steps on Alice's biometric identity $\omega' = (\omega'_i)_{i=1}^n$ where $\omega'_i \in \mathbb{Z}_q^*$. Therefore, this shows that Wang and Kim FIBS is not secure against key only attack.

3.2.3 Cryptanalysis on the Chen et al.'s FIBS Scheme

In this section, we briefly describe the Chen et al.'s FIBS scheme and subsequently present the collusion attack on it.

3.2.3.1 The Chen et al. FIBS Scheme

We describe the the Chen et al. FIBS (Chen, Zhu, Cao and Geng, 2009) scheme as follows:

Setup(1^k). Given the security parameter 1^k, PKG constructs a system with groups and a bilinear pairing $\mathcal{B} = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e(\cdot,\cdot))$. All of the three groups have the same order q and $|q| = 1^k$. Two generators $P \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$ as well as two secret values $\gamma, \alpha \in \mathbb{Z}_q^*$ are randomly selected. Besides, $d_i \in \mathbb{Z}_q^*$ are randomly selected to construct the set $D = \{d_i\}_{i=1}^{m-1}$, where $\mathcal{M}$ is the maximal size of an authorized set. This corresponds with a set of dummy users, which is used to complete a set of authorization. Finally, a set $T = \{t_i\}_{i=1}^{n}$ is randomly selected from $\mathbb{Z}_q$. Define the master key as $MK = (P, \gamma, \alpha)$. The signing key is $PK = (m, R, v, \alpha Q, \{\alpha \cdot \gamma^i Q\}_{i=1}^{2m-1}, D)$ and the verifying key is $VK = (m, Q, \{\gamma^i Q\}_{i=1}^{m-2}, D)$, where $R = \alpha \cdot \gamma P$ and $v = e(P,Q)^{\alpha}$.

Extract(MK). Given $MK = (P, \gamma, \alpha)$ and a user U described by an attribute set W, PKG randomly chooses $apk_i = x_i \in \mathbb{Z}_q^*$ for each attribute in W. Output the user keys $\{apk_i = x_i, ask_i = \frac{1}{\gamma + x_i} P\}_{i \in W}$. The private key $\{ask_i\}_{i \in W}$ are privately given to the user, whereas $\{apk_i\}_{i \in W}$ are publicly published.

Sign(PK, S, t, M). Given the signing key $PK = (m, R, v, \alpha Q, \{\alpha \cdot \gamma^i Q\}_{i=1}^{2m-1}, D)$, a set of target attributes S which is identified to $S = (apk = x_1, \ldots, x_s)$ and a threshold t where $t \leq s = |S| \leq m$, signer randomly picks $k \in \mathbb{Z}_q^*$ and compute $Hdr =$

(C_1, C_2, σ). Given message $M = (u_1, u_2, \ldots, u_n) \in \{1,0\}^n$ and $p = t_0 \prod_{i=1}^{n} t_i^{u_i}$, signer calculates $C_1 = -kR$, $C_2 = k \cdot \alpha \cdot \prod_{x_i \in S}(\gamma + x_i) \cdot \prod_{x \in D_{m+t-s-1}}(\gamma + x)Q$, $\sigma = v^{kp}$. Then, signer broadcasts the message $C = (Hdr, S, t)$ to the receivers.

Verify(CK, C, M)**.** When a user receives a message C with an attribute set W, he can verify the signature included in the C if and only if there are at least t same attributes in W and S such that $|W \cap S| \geq t$. The user randomly chooses a subset T from W with t attributes included in S, i.e. $T \in \{W \cap S\}$ and $|T| = t$.

Given the user public keys $\{x_i\}_{i \in T}$, signature elements $\{\sigma_i = e(P, C_2)^{\frac{1}{\gamma + x_i}}\}_{i \in T}$ and any $1 \leq j \leq \beta \leq t$, compute $L_{j,\beta} = \sigma_\beta^{\frac{1}{(\gamma+x_i)\ldots(\gamma+x_j)}} = e(P, C_2)^{\frac{1}{\gamma+x_\beta} \cdot \frac{1}{(\gamma+x_1)\ldots(\gamma+x_j)}}$. From the induction $L_{j,\beta} = (\frac{L_{j-1,j}}{L_{j-1,\beta}})^{\frac{1}{x_\beta - x_j}}$ for $j = 1, \ldots, t-1$ and $\beta = j+1, \ldots, t$, we obtain $L_t = L_{t-1,t} = e(P, C_2)^{\frac{1}{(\gamma+x_1)(\gamma+x_2)\ldots(\gamma+x_t)}}$.

To verify the signature, given $M = (u_1, u_2, \ldots, u_n)$, $S, T, Hdr = (C_1, C_2)$, compute $p = t_0 \prod_{t=1}^{n} t_i^{u_i}$ and $\sigma' = (L \cdot e(C_1, f_{T,S}(\gamma)Q))^{c^{-1}}$, where $f_{T,S}(\gamma)$ is a $(m-2)-$degree polynomial $f_{T,S}(\gamma) = \frac{1}{\gamma} \cdot (\prod_{x \in U}(\gamma + x) - c)$, $U = S \cup D_{m+t-s-1} - T$ and $c = \prod_{x \in U} x \in \mathbb{Z}_q$. Accept the signature if $\sigma'^{f_{T,S}(\gamma)} = \sigma$, reject otherwise.

3.2.3.2 Collusion Attack

We now mount the collusion attack on the Chen et al.'s FIBS scheme. To ease the explanation, let each element $i \in W$ be represented by a number and the threshold t equals to 4. Two misbehaved users Alice and Bob

who hold the information (W_A={1,2,3,4,5}, $apk_1,\ldots,apk_5$, $ask_1,\ldots,ask_5$) and (W_B={6,7,8,9,10}, $apk_6,\ldots,apk_{10}$, $ask_6,\ldots,ask_{10}$), respectively. are trying to forge the signature σ of the user Carol whose information is (W_C={3,4,6,7,10}, $apk_3, apk_4, apk_6, apk_7, apk_{10}$, $ask_3, ask_4, ask_6, ask_7, ask_{10}$). Obviously, Alice or Bob alone cannot forge Carol's signature as $|W_A \cap W_C| = 2 < t$ and $|W_B \cap W_C| = 3 < t$. We now show that Alice and Bob can collude together by combining their user keys to forge Carol's signature as follows:

1. Combine attribute set:

   $W_{AB} = W_A \cup W_B = \{1,2,3,4,5,6,7,8,9,10\}$

2. Combine public key:

   $\{apk_{AB,i}\}_{i\in W_{AB}} = \{apk_{A,i}\}_{i\in W_A} \cup \{apk_{B,i}\}_{i\in W_B} = \{apk_1,\ldots,apk_{10}\}$

3. Combine private key:

   $\{ask_{AB,i}\}_{i\in W_{AB}} = \{ask_{A,i}\}_{i\in W_A} \cup \{ask_{B,i}\}_{i\in W_B} = \{ask_1,\ldots,ask_{10}\}$

Alice and Bob's collusion allows them to generate the signature σ on any message $\mathcal{M}$ on behalf of Carol as they obtained her attributes $W_C \subseteq W_{AB}$, public key $\{apk_{C,i}\}_{i\in W_C} \subseteq \{apk_{AB,i}\}_{i\in W_{AB}}$ and private key $\{ask_{C,i}\}_{i\in W_C} \subseteq \{ask_{AB,i}\}_{i\in W_{AB}}$. Notice that Alice and Bob also can forge the signature of any user U as long as the condition $|W_{AB} \cap W_U| \geq t(=4)$ is met where W_U is the attribute set of U.

3.2.4 Discussion

The **Extract** algorithm in fuzzy identity-based cryptosystems are the same where the user private key is the signature of the PKG on the corresponding user public identity. This mechanism ensures the randomness and unique-

Table 3.4: Comparison on the Similarities of FIBS Schemes

IBS Scheme	Adopt Basic FIBE	Adopt Full FIBE	Key Only Attack	Collusion Attack
Yang, Cao and Dong (2011)	-	Yes	-	-
Wang et al. (2009)	Yes	-	-	-
Wang and Kim (2009)	Partial	-	Yes	-
Chen, Zhu, Cao and Geng (2009)	-	-	-	Yes

ness of every user private key. In Wang and Kim's FIBS, the **Extract** algorithm of Sahai and Waters' basic FIBE construction (Sahai and Waters, 2005) is partially adopted where the master secret key is divided into pieces using a $(d-1)$-degree polynomial with respect to the user public key but without randomisation. As the user public key is publicly known, it is thus easy for an adversary $\mathcal{A}$ to forge a user private key. To avoid this, the **Extract** algorithm of Sahai and Waters' FIBE (Sahai and Waters, 2005) has to be adopted completely but this makes the Wang and Kim's FIBS the same as the Wang et al.'s FIBS (Wang et al., 2009). This is due to both FIBS schemes are using the same **Setup** and **Extract** algorithms of Sahai and Waters' basic FIBE construction.

Chen et al. did not follow the approach of Sahai and Waters FIBE, and constructed their FIBS scheme using the technique of exponent inversion in additive form. The FIBS resists key only attack but it is suffered from the collusion attack since the user private keys are not bound together. From Table 3.4, we can see that the other FIBS schemes resist collusion attack as they are using

the **Extract** algorithm of Sahai and Waters FIBE which applies secret sharing scheme on the user private keys to prevent collusion attack.

On the other hand, in Yang et al.'s scheme, the **Extract** algorithm is the same as that in Sahai and Waters' full FIBE (Sahai and Waters, 2005). Similar to Wang and Kim's FIBS, this **Extract** algorithm also divides the master secret key into pieces but these pieces are further randomised using random values $r_i \in \mathbb{Z}_p$ for each attribute in the biometric identity.

The **Extract** algorithm of Wang et al. (2009) is the same as that of Sahai and Waters' basic FIBE construction. This FIBS resists key only attack due to the master secret keys s and t_i for $i = 1, \ldots, |\Omega|$ are bound together using exponent inversion though the scheme is not efficient as the number of t_i grows linearly with the size of the biometric identity.

In general, we can see that FIBS is also IBS+LP and in fact, the Bio-IBS appeared in literatures are IBS+FE (Burnett et al., 2004, Liu et al., 2007, Fan et al., 2009, Sarier, 2010*a*). These Bio-IBS also have the same redundancy problems as in Bio-IBE. However, we do not intend to analyse this problem again as it would be using almost the similar approach.

3.3 Fuzzy Identity-Based Identification Schemes

Although pairing-based constructions (Sahai and Waters, 2005, Sakai and Kasahara, 2003) are the main stream for FIBE and FIBS, the only fuzzy identity-based identification (FIBI) (Tan et al., 2009) in the literature is in the discrete logarithm (DL) construction. The DL construction has significantly lower complexity compared to that of former and their structures are notably different. Moreover, as discussed in the previous chapter particularly in Table 1.2, FIBI scheme is the best candidate for authentication mechanism. Thus, we are not performing a thoroughgoing analysis on the DL-based FIBI scheme here, but in the coming chapter only.

Anyway, cryptanalysis on identification schemes should be performed for the completeness of this chapter. Due to the scarcity of FIBI scheme in the literature, we deviate our focus to identity-based identification (IBI)[2] schemes.

In EuroPKI 2008, Chin et al. (2008) proposed an efficient and provable secure IBI scheme in the standard model based on Sahai-Waters construction. However, we discovered a subtle flaw in the security proof which renders the proof of security useless. While no weakness has been found in the scheme itself, a scheme that is desired would be one with an accompanying proof of security. In this section, we show an efficient fix to the flaw where by only one extra pairing operation is added to the identification protocol. Moreover, this

[2]Section 3.1 and 3.2 showed that IBC is closely related to FIBC, such that one can easily obtain an FIBE from IBE as well as FIBS from IBS. Thus, the same applied to FIBI and we argue that analysis of an IBI is equivalent to that on an FIBI.

extra pairing operation can be pre-computed with an acceptable size increment in user's private key. Our opinion is that while other IBI schemes have been proposed, a scheme provable secure in the standard model is still of research interest, especially one that is run-time efficient.

3.3.1 Chin et al.'s IBI Scheme

We now review Chin et al.'s scheme (Chin et al., 2008) which is parameterised by two finite cyclic groups $\mathbb{G}$ and $\mathbb{G}_T$ of prime order q. Let g be the generator of $\mathbb{G}$ and set the super-logarithmic challenge length $l : \mathbb{N} \rightarrow \mathbb{N}$ for the identification protocol. Assume a cryptographic hash function $H : \{0,1\}^* \rightarrow \{0,1\}^n$ is used to hash identity strings of any length to ID of size n.

1. **Setup (S)**. Randomly select a secret $a \in \mathbb{Z}_q^*$, g_2, $u' \in \mathbb{G}$ and a vector $u = \langle u_1, u_2, \ldots, u_n \rangle$ of length n, where $u_i \in \mathbb{G}$ for all $i = 1, 2, \ldots, n$. Set $g_1 = g^a$ and publish the public parameters as $\langle \mathbb{G}, \mathbb{G}_T, e, g, g_1, u', \langle u \rangle, H \rangle$. The master secret key is g_2^a.

2. **Extract (E)**. Parse ID as an n-bit identity string with d_i denoting the I-th bit of ID. Let $ID = \{1, \ldots, n\}$ be the set of all I in which $d_i = 1$. Select $r \in \mathbb{Z}_q$. The user secret key can then be constructed as $usk = (S, R) = (g_2^a (u' \prod_{i \in ID} u_i)^r, g^r)$.

3. **Identification protocol**. (**Prover P** and **Verifier V**) will do the following:

   (a) **P** randomly chooses $z \in \mathbb{Z}_q$, computes $X = (u' \prod_{i \in ID} u_i)^z, Y = g_2^z$ and sends X, Y, R to **V**.

(b) **V** randomly picks a challenge $c \in \mathbb{Z}_{2^{l(k)}}$ and returns it to **P**.

(c) **P** computes $Z = S^{z+c}$ and returns Z to **V**.

V accepts if $e(Z,g) = e(Yg_2^c, g_1)e(X(u'\prod_{i\in ID} u_i)^c, R)$, rejects otherwise.

We now show the original security proof from Chin et al. (2008), before highlighting the flaw in the next section for easier readability.

3.3.2 Original Security Proofs

The original proof of security against impersonation under passive attack is done by contradicting the hardness of CDH. In particular, Chin et al. showed that with the help of an impersonator I which is equipped with passive attack ability, there exists an algorithm $\mathcal{M}$ that can be used to solve the CDH. But the CDH is intractable with technologies to date and thus such impersonator does not exist. We now briefly describe the original proofs before pointing out the flaw.

3.3.2.1 Security Against Passive Attack

Recall that in the security model of IBI in Section 2.4.1, the challenger $\mathcal{M}$ setups the scheme's parameters and interacts with the impersonator I to dissolve a solution for the underlying hard problem. I is allowed to query for extract queries and the identification transcript in the passive attack on any ID of its choice. If $\mathcal{M}$ fails to answer any of the queries, the security games has to be aborted.

Theorem 3.1. *If the CDH is (t,ε')-hard, Chin et al.'s IBI scheme is (t,q_I,ε)-secure against impersonation under passive attack in the standard model.*

Proof. $\mathcal{M}$ is given a cyclic group $\mathbb{G}$, a generator $g \in \mathbb{G}$ and elements g^a, g^b. $\mathcal{M}$ simulates the challenger for I as follows in order to compute g^{ab}:

1. **Setup**. $\mathcal{M}$ crafts master public key as $\langle \mathbb{G}, \mathbb{G}_T, e, g, g_1, g_2, u', \langle u \rangle, H \rangle$. $\mathcal{M}$ defines two functions $F(ID)$ and $J(ID)$ to replace the Waters' hash function such that
$$u' \prod_{i \in ID} u_i = g_2^{F(ID)} g^{J(ID)}.$$

2. **Extract Query**. $\mathcal{M}$ computes the user private key as:
$$\left(\tilde{S}_i = g_1^{-J(ID_i)/F(ID_i)} (u' \prod_{i \in ID} u_i)^{r_i}, \tilde{R}_i = g_1^{-1/F(ID_i)} g^{r_i} \right)$$
As $F(ID)$ has been crafted such that its value is equal to zero in some occasions, $\mathcal{M}$ will abort if that happens because it is unable to construct the private key (fraction with denominator zero is undefined).

3. **Transcript Query**. If $F(ID_j) \neq 0 \bmod l$ then $\mathcal{M}$ runs the Extract Query algorithm and produces a valid transcript for I. Else if $F(ID_j) = 0 \bmod l$, $\mathcal{M}$ chooses $r_j, z_j \in \mathbb{Z}_q, c_j \in \mathbb{Z}_{2^{l(k)}}$ and sends I the transcript as:
$$\left(\tilde{X}_j = (u' \prod_{i \in ID} u_i)^{z_j}, \tilde{Y}_j = g^{z_j} g_2^{-c_j}, \tilde{R}_j = g^{r_j}, c_j, \tilde{Z}_j = g_1^{z_j} (u' \prod_{i \in ID} u_i)^{(z_j + c_j) r_j} \right)$$
After some time, I outputs the challenge identity $ID^* \neq ID_i$ that it wishes to be challenged on. $\mathcal{M}$ aborts if $F(ID^*) \neq 0 \bmod q$. I can still issue extract and identification queries but not on ID^*. I then takes the role of the cheating prover to convince $\mathcal{M}$. $\mathcal{M}$ obtains (X, Y, R, c_1, Z_1) and resets I to where it sent its commitment to obtain (X, Y, R, c_2, Z_2). Based on the reset lemma (Bellare and Palacio, 2002), $\mathcal{M}$ can extract S from two conversation transcripts with probability more than $(\varepsilon - q^{-1})^2$. $\mathcal{M}$ then calculates S as $S = (Z_1 Z_2^{-1})^{c_1 - c_2^{-1}}$ and outputs the solution to the CDH

as:

$$\frac{S}{R^{J(ID^*)}} = \frac{g^{ab}(u'\prod_{i\in ID} u_i)^r}{g^{J(ID)r}} = g^{ab}$$

□

3.3.2.2 Security Against Active and Concurrent Attacks

In the active and concurrent attacks, the challenger $\mathcal{M}$ have to answer the identification queries of I because I can act as a cheating verifier. The same proving technique was used for the proof of security against impersonation under active and concurrent attacks based on the OMCDH. We briefly describe the proof as follows.

Theorem 3.2. *If the OMCDH is (t,ε'')-hard, Chin et al.'s IBI scheme is (t,q_I,ε)-secure against impersonation under active and concurrent attacks in the standard model.*

Proof. The proof of the active and concurrent attacks is similar to the one of Theorem 3.1. Here we only point out the differences. To begin the game, $\mathcal{M}$ is given elements (g,g^a) and access to the *CHALL* and *CDH* oracle. $\mathcal{M}$ queries the *CHALL* oracle for W_0.

1. **Setup**.$\mathcal{M}$ sets $g_1 = g^a$ and queries the *CHALL* oracle for the initial challenge W_0, which it sets as g_2. The rest are simulated as the proof before.
2. **Extract Query**. This is similar as the proof before.
3. **Identification Query**. If $F(ID_j) \neq 0 \bmod l$, $\mathcal{M}$ will have no problem simulating an identification protocol instance for I by running the **Extract** algorithm first to obtain the private key for ID_j. Else, if $F(ID_j) = 0 \bmod l$, $\mathcal{M}$ keeps a counter $\mathcal{M}$ and does the following:

(a) $\mathcal{M}$ queries *CHALL* for W_m and lets $Y_j = W_m$. $\mathcal{M}$ also selects $r_j \in \mathbb{Z}_q$ and computes

$$\tilde{X_j} = g^{J(ID_j)} = (u' \prod_{i \in ID_j} u_i), \tilde{R_j} = g_1^{r_j}$$

$\mathcal{M}$ sends $\tilde{X_j}, \tilde{Y_j}, \tilde{R_j}$ to I.

(b) I picks a random challenge $c_j \in \mathbb{Z}_{2^{l(k)}}$ and sends it to $\mathcal{M}$.

(c) $\mathcal{M}$ queries the *CDH* oracle with $[W_m(u' \prod_{i \in ID_j} u_i)^{r_j}(W_0(u' \prod_{i \in ID} u_i))^{c_j}]$ and receives the response $\tilde{Z}_j = [W_m(u' \prod_{i \in ID_j} u_i)^{r_j}(W_0(u' \prod_{i \in ID} u_i))^{c_j}]^a$. $\mathcal{M}$ increments $\mathcal{M}$ by 1.

4. After some time I outputs the challenge identity $ID^* \neq ID_i$ that it wishes to be challenged on. $\mathcal{M}$ aborts if $F(ID^*) \neq 0 \bmod q$. I can still issue extract, except those on ID^*, and identification queries. I then takes the role of the cheating prover to try to convince $\mathcal{M}$. $\mathcal{M}$ obtains (X, Y, R, c_1, Z_1) then resets I to where it just sent its commitment to obtain (X, Y, R, c_2, Z_2). Based on the reset lemma (Bellare and Palacio, 2002), $\mathcal{M}$ can extract S from two conversation transcripts with probability more than $(\varepsilon - q^{-1})^2$. $\mathcal{M}$ then calculates S as $S = (Z_1 Z_2^{-1})^{c_1 - c_2{}^{-1}}$ and outputs the solution to the initial challenge of the OMCDH as:

$$\frac{S}{R^{J(ID^*)}} = \frac{W_0^a (u' \prod_{i \in ID} u_i)^r}{g^{J(ID)r}} = W_0^a$$

□

3.3.2.3 Flaw in Security Proofs

We now point out the portion of the proof where the flaw appears: the response of the simulator $\mathcal{M}$ for identification query queried by the impersonator I in the simulation. In the passive attack proof, whenever I issues a query

on the challenge identity, which is where $\mathcal{M}$ produces a valid transcript for ID_j:

$$\left(\tilde{X}_j = (u' \prod_{i \in ID} u_i)^{z_j}, \tilde{Y}_j = g^{z_j} g_2^{-c_j}, \tilde{R} = g^{r_j}, \tilde{Z}_j = g_1^{z_j} (u' \prod_{i \in ID} u_i)^{r_j (z_j + c_j)}\right)$$

Although this is a valid simulation that passes the check of completeness, the simulated transcript is not identically distributed compared to the honest valid transcript. In precise, I can discern that $e(X, g_2) = e((u' \prod i \in IDu_i)^z, Y)$ when $F(ID_j) \neq 0 \bmod l$ but $e(\tilde{X}, g_2) \neq e((u' \prod_{i \in ID} u_i), \tilde{Y})$ when $F(ID_j) = 0 \bmod l$ and the same occurrence happens in the active and concurrent proof. I who is able to verify this distribution pattern will stop the impersonation and the game will fail. The ability of performing such DH tuple check to distinguish between valid and simulated conversations in both proofs render the scheme not provable secure, even though no attack has yet been found on the scheme itself.

3.3.3 Fixing the IBI Scheme

As current trend in cryptography, we would want a scheme that is provable secure especially an efficient one in the standard model. We provide the fix with the new security proofs under the same hard problems.

3.3.3.1 Amending the IBI Scheme

In this section, we propose the fix for Chin et al.'s IBI scheme as follows:

1. **Setup (S)** and **Extract (E)** are same as original scheme.
2. **Identification protocol**. **Prover P** and **Verifier V** will do the following:

(a) **P** randomly chooses $z \in \mathbb{Z}_q$, computes $X = e((u' \prod_{i \in ID}) u_i, R)^z, Y = g_2^z$ and sends X, Y, R to **V**.

(b) **V** randomly picks a challenge $c \in \mathbb{Z}_{2^{l(k)}}$ and returns it to **P**.

(c) **P** computes $Z = S^{z+c}$ and returns Z to **V**.

V accepts if $e(Z, g) = e(Y g_2^c, g_1) \cdot X \cdot e((u' \prod_{i \in ID} u_i)^c, R)$. To verify completeness:

$$
\begin{aligned}
e(Z, g) &= e(S^{z+c}, g) \\
&= e((g_2^a (u' \prod_{i \in ID} u_i)^r)^{(z+c)}, g) \\
&= e((g_2^a)^{z+c}, g) e((u' \prod_{i \in ID} u_i)^{rz} (u' \prod_{i \in ID} u_i)^{rc}, g) \\
&= e(g_2^z g_2^c, g^a) e((u' \prod_{i \in ID} u_i)^z, g^r) e((u' \prod_{i \in ID} u_i)^c, g^r) \\
&= e(Y g_2^c, g_1) \cdot X \cdot e((u' \prod_{i \in ID} u_i)^c, R).
\end{aligned}
$$

Up to here, the amendment may not be self-evident yet in fixing the flaw mentioned. Recall that the flaw of Chin et al.'s IBI scheme is discovered in the security proof but not the scheme construction itself. In fact, no problem is found in the original construction though the security proof is flawed. We now show how to take advantage of such simple amendment on the construction to overcome the problem in the original security proof. The new detailed security proofs are as follows.

3.3.3.2 Security Against Passive Attack

Theorem 3.3. *If the CDH is (t,ε')-hard, the fixed IBI scheme is (t,q_I,ε)-secure against impersonation under passive attack in the standard model, where*

$$t' = t + O(\rho(2nq_I) + \tau(q_I)), \varepsilon \leq \sqrt{4q_e(n+1)\varepsilon'} + q^{-1},$$

where ρ represents time taken to do a multiplication in $\mathbb{G}$, τ is the time taken to do an exponentiation in $\mathbb{G}$ and q_e represents the number of extract queries made, q_i represents the number of transcript queries made and $q_I = q_e + q_i$.

Proof. Suppose there exists an impersonator I who (t,q_I,ε) breaks the IBI scheme. Then, we show an algorithm $\mathcal{M}$ which (t',ε')-breaks the CDH assumption by runing I as a subroutine. $\mathcal{M}$ is given a cyclic group $\mathbb{G}$, a generator $g \in \mathbb{G}$ and elements g^a, g^b. $\mathcal{M}$ simulates the challenger for I as follows:

1. **Setup**. $\mathcal{M}$ sets $l = 2q_I$ and randomly chooses $k \in \mathbb{Z}_n$. Assume that $l(n+1) < q$ for the given values of q_I and n. Furthermore, $\mathcal{M}$ randomly chooses $x' \in \mathbb{Z}_l$, a vector $\langle X \rangle$ of length n with $x_i \in \mathbb{Z}_l$ for all I, a randomly selected $y' \in \mathbb{Z}_q$ and a vector $\langle y \rangle$ of length n with $y_i \in \mathbb{Z}_q$ for all I. Define the following functions:

$$F(ID) = x' + \sum_{i \in ID} x_i - lk \bmod q \text{ and } J(ID) = y' + \sum_{i \in ID} y_i \bmod q.$$

$\mathcal{M}$ now sets $g_1 = g^a$ and $g_2 = g^b$. $\mathcal{M}$ also sets $u' = g_2^{x'-lk} g^{y'}$ and a vector $\langle u \rangle$ of length n consisting of n elements $u_i = g_2^{x_i} g^{y_i}$. $\mathcal{M}$ passes the system parameters to I as $\langle \mathbb{G}, \mathbb{G}_T, e, g, g_1, g_2, u', \langle u \rangle, H \rangle$. Note that with functions $F(ID)$ and $J(ID)$, we have:

$$u' \prod_{i \in ID} u_i = g_2^{F(ID)} g^{J(ID)}.$$

2. **Extract Query**. When I queries $\mathcal{M}$ for the private key of ID_i, I checks if $F(ID) = 0 \bmod l$ and aborts if it is. This is because given the assumption $l(n+1) < q$ implies $0 \leq lk < q$ and $0 \leq x' + \sum_{i \in ID_i} x_i < q$. Therefore $F(ID) = 0 \bmod q$ implies that $F(ID) = 0 \bmod l$ and the simulator aborts because it is unable to construct the private key. Otherwise if $F(ID) \neq 0 \bmod l$, $\mathcal{M}$ constructs the private key by randomly selecting $r_i \in \mathbb{Z}_q$ and computes the user private key as

$$\left(\tilde{S}_i = g_1^{-J(ID_i)/F(ID_i)} (u' \prod_{i \in ID} u_i)^{r_i}, \tilde{R}_i = g_1^{-1/F(ID_i)} g^{r_i} \right).$$

To I, the private keys generated by $\mathcal{M}$ will be indistinguishable from those generated by a true challenger.

3. **Transcript Query**. When I queries $\mathcal{M}$ for an transcript query ID_j, if $F(ID_j) \neq 0 \bmod l$ then $\mathcal{M}$ just runs the Extract Query algorithm to generate a private key to produce a valid transcript for I. However, if $F(ID_j) = 0 \bmod l$ then $\mathcal{M}$ chooses $r_j, z_j \in \mathbb{Z}_q, c_j \in \mathbb{Z}_{2^{l(k)}}$ and sends I the transcript as

$$\left(\tilde{X}_j = e\left(u' \prod_{i \in ID} u_i, \tilde{R} \right)^{z_j}, \tilde{Y}_j = g^{z_j} g_2^{-c_j}, \tilde{R}_j = g^{r_j}, c_j, \tilde{Z}_j = g_1^{z_j} (u' \prod_{i \in ID} u_i)^{(z_j + c_j) r_j} \right).$$

I can check that this is a valid transcript since:

$$\begin{aligned}
e\left(\tilde{Z}_j, g\right) &= e\left(\tilde{Z}_j^{z_j + c_j}, g\right) \\
&= e\left(g_1^{z_j} \left(u' \prod_{i \in ID} u_i \right)^{(z_j + c_j) r_j}, g \right) \\
&= e\left(g_1^{z_j}, g\right) e\left(\left(u' \prod_{i \in ID} u_i \right)^{(z_j + c_j) r_j}, g \right) \\
&= e\left(g^{z_j}, g_1\right) e\left(\left(u' \prod_{i \in ID} u_i \right)^{z_j} \left(u' \prod_{i \in ID} u_i \right)^{c_j}, g^{r_j} \right) \\
&= e\left(g^{z_j} g_2^{-c_j}, g_1\right) e\left(\left(u' \prod_{i \in ID} u_i \right)^{z_j}, \tilde{R} \right) e\left(\left(u' \prod_{i \in ID} u_i \right)^{c_j}, \tilde{R} \right) \\
&= e(\tilde{Y} g_2^{c_j}, g_1) \tilde{X}_j e\left(\left(u' \prod_{i \in ID} u_i \right)^{c_j}, \tilde{R} \right).
\end{aligned}$$

After some time, I outputs the challenge identity $ID^* \neq ID_i$ that it wishes to be challenged on. $\mathcal{M}$ aborts if $F(ID^*) \neq 0 \bmod q$. I can still issue extract and identification queries, except those on ID^*. I then takes the role of the cheating prover to convince $\mathcal{M}$. $\mathcal{M}$ obtains (X, Y, R, c_1, Z_1) and resets I to where it just sent its commitment to obtain (X, Y, R, c_2, Z_2). Based on the reset lemma (Bellare and Palacio, 2002), $\mathcal{M}$ can extract S from two conversation transcripts with probability more than $(\varepsilon - q^{-1})^2$. $\mathcal{M}$ then calculates S as $S = (Z_1 Z_2^{-1})^{c_1 - c_2^{-1}}$ and outputs the solution to the CDH as:

$$\frac{S}{R^{J(ID^*)}} = \frac{g^{ab}(u' \prod_{i \in ID} u_i)^r}{g^{J(ID)r}} = g^{ab}.$$

The probability of $\mathcal{M}$ winning the game and solving the CDH is now calculated. Firstly, the probability that $\mathcal{M}$ can extract two valid transcripts from I is given by $\Pr[M \text{ computes } g^{ab} | \neg abort] \geq (\varepsilon - \frac{1}{q})^2$. Upon extraction of S, $\mathcal{M}$ is able to compute g^{ab}. We break down the probability of $\mathcal{M}$ winning the CDH to:

$$\begin{aligned}
\Pr[M \text{ computes } g^{ab}] &= \Pr[M \text{ computes } g^{ab} \wedge \neg \text{abort}] \\
&= \Pr[M \text{ computes } g^{ab} | \neg \text{abort}] \Pr[\neg \text{abort}] \\
&\geq \left(\varepsilon - q^{-1}\right)^2 \Pr[\neg \text{abort}].
\end{aligned}$$

It remains to calculate $\Pr[\neg \text{abort}]$. Define the following events: 1) Event A_i where $\mathcal{M}$ answers all queries $F(ID_i) \neq 0 \bmod l$, and 2) Event A^* where I outputs the challenge identity ID^* where $F(ID) = 0 \bmod q$. Calculate the probability of A^* as:

$$\begin{aligned}
\Pr[A^*] &= \Pr[F(ID^*) = 0 \bmod q \vee F(ID^*) = 0 \bmod l] \\
&= \Pr[F(ID^*) = 0 \bmod l] \Pr[F(ID^*) = 0 \bmod q | F(ID^*) = 0 \bmod l] \\
&= \frac{1}{l}\left(\frac{1}{n+1}\right).
\end{aligned}$$

Notice that:

$$\begin{aligned}\Pr[\bigcap_{i=1}^{q_e} A_i|A^*] &= 1-\Pr[\bigcup_{i=1}^{q_e} A_i|A^*] \\ &= 1-\sum_{i=1}^{q_e}\Pr[\neg A_i|A^*] \\ &= 1-q_e/l.\end{aligned}$$

Therefore, the probability of $\mathcal{M}$ not aborting is:

$$\begin{aligned}\Pr[M\neg\text{abort}] &= \Pr[\bigcap_{i=1}^{q_e} A_i \wedge A^*] \\ &= \Pr[A^*]\Pr[\bigcap_{i=1}^{q_e} A_i|A^*] \\ &= \frac{1}{l(n+1)}(1-q_e/l) \\ &= \frac{1}{4q_e(n+1)}\end{aligned}$$

since $l = 2q_e$ in the simulation. Finally the probability of $\mathcal{M}$ breaking CDH is:

$$\begin{aligned}\Pr[M \text{ computes } g^{ab}] &\geq \left(\varepsilon - q^{-1}\right)^2 \frac{1}{4q_e(n+1)} \\ \varepsilon' &\geq \left(\varepsilon - q^{-1}\right)^2 \frac{1}{4q_e(n+1)} \\ \varepsilon &\leq \sqrt{4q_e(n+1)\varepsilon'} + q^{-1}\end{aligned}$$

as desired. □

Although the new scheme needs to compute one extra pairing where $X = e(u'\prod_{i\in ID} u_i, R)^z$, I cannot perform the DH check as in the original proof because $\tilde{X}_j$ is no longer a point, but an element in $\mathbb{G}_T$. Besides, the randomness of z_j will uniformly distributes $\tilde{X}_j$, making it indistinguishable from the actual value in I's view. The same reasoning is applicable to the proof of active and concurrent attacks and hence fix the flaw completely.

3.3.3.3 Security Against Active and Concurrent Attacks

Theorem 3.4. *If the OMCDH is (t,ε'')-hard, the fixed IBI scheme is (t,q_I,ε)-secure against impersonation under active and concurrent attacks in the standard model, where*

$$t'' = t + O(\rho(2nq_I) + \tau(q_I)), \varepsilon \leq \sqrt{4q_e(n+1)\varepsilon''} + q^{-1},$$

where ρ represents time taken to do a multiplication in G, τ is the time taken to do an exponentiation in G and q_e represents the number of extract queries made, q_i represents the number of identification queries made and $q_I = q_e + q_i$.

Proof. The proof of the active and concurrent attacks is similar to the one of Theorem 3.3. Here we only point out the differences. To begin the game, $\mathcal{M}$ is given elements (g, g^a) and access to the *CHALL* and *CDH* oracles. $\mathcal{M}$ queries the *CHALL* oracle for W_0.

1. **Setup**.$\mathcal{M}$ sets $g_1 = g^a$ and queries the *CHALL* oracle for the initial challenge W_0, which it sets as g_2. The rest are simulated as the proof before.
2. **Extract Query**. This is similar as the proof before.
3. **Identification Query**. As before if $F(ID_j) \neq 0 \bmod l$, $\mathcal{M}$ will have no problem simulating an identification protocol instance for I by running the **Extract** algorithm first to obtain the private key for ID_j. However, if $F(ID_j) = 0 \bmod l$, $\mathcal{M}$ keeps a counter $\mathcal{M}$ and does the following:
   (a) $\mathcal{M}$ queries *CHALL* for W_m and lets $Y_j = W_m$. $\mathcal{M}$ also selects $r_j \in \mathbb{Z}_q$ and computes $\tilde{X_j} = e(g^{J(ID_j)}, \tilde{R_j})^{z_j} = e((u' \prod_{i \in ID_j} u_i), \tilde{R_j})^{z_j}, \tilde{R_j} = g_1^{r_j}$.$\mathcal{M}$ sends $\tilde{X_j}, \tilde{Y_j}, \tilde{R_j}$ to I.
   (b) I randomly picks a challenge $c_j \in \mathbb{Z}_{2^{l(k)}}$ and returns it to $\mathcal{M}$.
   (c) $\mathcal{M}$ queries the *CDH* oracle with $[W_m W_0^{c_j}]$ and receives the response $[W_m W_0^{c_j}]^a$. $\mathcal{M}$ returns the response $\tilde{Z_j} = (W_m^a W_0^{ac_j})(u' \prod_{i \in ID_j} u_i)^{r_j(z_j + c_j)}$

and increments $\mathcal{M}$ by 1. I can check this is a valid conversation:

$$\begin{aligned}
e(\tilde{Z}_j, g) &= e(W_m^a W_0^{ac_j} (u' \prod_{i \in ID_j} u_i)^{r_j(z_j+c_j)}, g) \\
&= e(W_m^a W_0^{ac_j}, g) e((u' \prod_{i \in ID_j} u_i)^{r_j(z_j+c_j)}, g) \\
&= e(W_m W_0^{c_j}, g^a) e((u' \prod_{i \in ID} u_i)^{z_j+c_j}, g^{r_j}) \\
&= e(W_m W_0^{c_j}, g^a) e((u' \prod_{i \in ID} u_i)^{z_j}, g^{r_j}) e((u' \prod_{i \in ID} u_i)^{c_j}, g^{r_j}) \\
&= e(\tilde{Y}_j W_0^{c_j}, g_1) \cdot \tilde{X}_j \cdot e((u' \prod_{i \in ID} u_i)^{c_j}, \tilde{R}_j)
\end{aligned}$$

After some time I outputs the challenge identity $ID^* \neq ID_i$ that it wishes to be challenged on. $\mathcal{M}$ aborts if $F(ID^*) \neq 0 \bmod q$. I can still issue extract, except those on ID^*, and identification queries. I then takes the role of the cheating prover to convince $\mathcal{M}$. $\mathcal{M}$ obtains (X, Y, R, c_1, Z_1) and resets I to where it just sent its commitment to obtain (X, Y, R, c_2, Z_2). Based on the reset lemma (Bellare and Palacio, 2002), $\mathcal{M}$ can extract S from two conversation transcripts with probability more than $(\varepsilon - q^{-1})^2$. $\mathcal{M}$ then calculates S as $S = (Z_1 Z_2^{-1})^{c_1 - c_2{}^{-1}}$ and outputs the solution to the initial challenge of the OMCDH as:

$$\frac{S}{R^{J(ID^*)}} = \frac{W_0^a (u' \prod_{i \in ID} u_i)^r}{g^{J(ID)r}} = W_0^a$$

$\mathcal{M}$ then calculates the other $\mathcal{M}$ challenge points' solution as:

$$\frac{Z_j}{W_0^{ac_j} (u' \prod_{i \in ID} u_i)^{r_j(z_j+c_j)}} = \frac{W_m^a W_0^{ac_j} (u' \prod_{i \in ID} u_i)^{r_j(z_j+c_j)}}{W_0^{ac_j} (u' \prod_{i \in ID} u_i)^{r_j(z_j+c_j)}} = W_m^a$$

Calculation for the probability of $\mathcal{M}$ winning the game and solving the OMCDH is similar to the proof before, only the CDH is substituted with the OMCDH.

□

3.3.4 Efficiency Analysis

The major effect brought by the amendment to the protocol is an additional pairing in commitment phase and changes on the verifier's verification formula. Even though the new verification $X \cdot e((u' \prod_{i \in ID} u_i)^c, R)$ looks more complicated compared to $e(X(u' \prod_{i \in ID} u_i)^c, R)$ of the original's, the former is actually more efficient as multiplication in $\mathbb{G}_T$ is about 7 times faster than point addition in $\mathbb{G}$ (Tan et al., 2010).

On the other hand, we can compute the extra pairing operation incurred by the fix for only once instead of every time the identification protocol is activated by preparing an intermediate value $X' = e((u' \prod_{i \in ID} u_i), R)$ during the **Extract** phase, so that computing $X' = e((u' \prod_{i \in ID} u_i), R)^z$ during each interaction can be simplified into $X = X'^z$. With such pre-computation, the amendment can be viewed as replacing the point multiplication $X = (u' \prod_{i \in ID} u_i)^z$ in $\mathbb{G}$ from the original commitment phase by exponentiation $X = X'^z$ in $\mathbb{G}_T$. This will speed up the **Identification Protocol** as the latter is approximately 10 times faster than the former (Tan et al., 2010). If the underlying elliptic curve (in prime field) is using parameters of 80-bit security, we only need to increment the size of private key for $|\mathbb{G}_T| = 1024$ bits to enjoy the this efficiency.

The only available pre-computation for the original scheme is to pre-compute the hash value *ID* during Extract phase with an increment of $2 \times |\mathbb{G}|$ in the private key size. Given the same size growth in the private key, the amended

scheme obviously outperformed the original's. The complexity comparison for the original and amended identification protocols is summarized in Table 3.5.

Table 3.5: Complexity Comparison for Identification Protocols

Operation	Identification Protocol			
	Original		Amended	
	Without Pre-computation	With Pre-computation	Without Pre-computation	With Pre-Computation
Addition in $\mathbb{Z}_q$	1	1	1	1
Point Addition	$2n+4$	$n+3$	$2n+3$	$n+2$
Point Multiplication	5	5	4	4
Multiplication in $\mathbb{G}_T$	1	1	2	2
Exponentiation in in $\mathbb{G}_T$	0	0	1	1
Pairing	3	3	4	3

3.4 Conclusion

We have performed analysis on the primitives of FIBC, namely, FIBE and FIBS schemes. Due to the lack of FIBI schemes in the literature, we analysed IBI schemes, which is the ancestor of FIBI. We pointed out some implementation issues, as well as the cryptographic flaws in these primitives. The results showed that even when a cryptographic scheme is free from error and its security proof is sound, it may not be practical in the real world. Besides, certain cryptographic schemes are not able to be proven secure in nature despite the fact that the scheme construction is error-free and efficient.

We showed that IBC is closely related to FIBC where they share the similar **Setup** and **Extract** algorithms besides the fact that FIBC equals to IBC+LP or IBC+FE. In these two constructions, FIBC=IBC+FE is not preferable because such construction is placing an biometrics add-on on top of an IBC scheme where the security of IBC and FE are separated. In precise, these two entities are only linked by a hash function but not the core security mechanism of biometrics, which is the matching threshold. On the other hand, the FIBC=IBC+LP is an integration of biometrics into an IBC's public key in which the biometric matching threshold is embedded into it. Thus, the former cryptographic construction cannot provide provable security but the latter can, while enjoying the benefits of biometrics.

CHAPTER 4

SECURITY ENHANCEMENTS FOR SCHNORR FIBI SCHEME

In this chapter, we enhance the security of the only FIBI (Tan et al., 2009) scheme in the literature by providing tighter security proofs for its underlying IBI scheme, namely, Schnorr IBI scheme. Using the rewinding technique from Reset Lemma (Bellare and Palacio, 2002), Schnorr IBI can be proven secure against impersonation under passive attack and active and concurrent attacks if the DL problem and OMDL problem are hard in the ROM though its security reduction is not tight. We present two techniques to provide a tight security reduction: 1) using weaker hard problem with three extra elements in public key 2) using weaker hard problem with easy ID. Both reduce the required security parameters from k^2 to only k.

4.1 Technique 1: Weaker Hard Problem

The concept of identity-based cryptography was introduced by Shamir (1985) but there was no rigorous definition as well as security proof for identity-based identification (IBI) scheme until the independent works of Kurosawa and Heng (2004) and Bellare et al. (2009). Kurosawa and Heng proposed transforming certain class of standard digital signature schemes to an IBI scheme, while Bellare et al. concerned about transforming standard identification schemes to IBI. Some other IBI schemes (Kurosawa and Heng, 2005, 2006, Chin et al.,

2008, Yang et al., 2008, Thorncharoensri et al., 2009, Rckert, 2010) were published after that.

Although the Schnorr IBI can be obtained easily by applying the transformation frameworks on the Schnorr standard signature scheme, the resulted security proof is not tight. The generality of the transformation frameworks cannot provide scheme-dependent optimisation as opposed to direct proof which will provide tighter security reductions. A security reduction is said to be tight if the probability of breaking an IBI scheme is close (i.e. ≈ 1) to the probability of solving the underneath mathematical hard problem. If the security reduction is tight, it indicates that the IBI scheme is almost as secure as the underneath mathematical hard problem. A non-tight security reduction needs a larger key size k in order to achieve the same level of security. For instance, let the probability of breaking an IBI scheme be $\varepsilon_{IBI} \approx \sqrt{\varepsilon}$, where ε is the probability of solving a hard problem such as discrete logarithm (DLOG) problem. If $\varepsilon = 2^{-80}$ when $k = 160$, $\varepsilon_{IBI} \approx \sqrt{2^{-80}} \approx 2^{-40}$ which is not acceptable. In order to achieve a desired security level i.e. $\varepsilon_{IBI} \approx 2^{-80}$, value of k needs to be increased so that ε decreases until $\varepsilon_{IBI} \approx \sqrt{2^{-160}} \approx 2^{-80}$.

4.1.1 Related Works

In year 2007, Goh et al. (2007) presented two novel techniques to prove the security of standard signature schemes. The authors showed that with only an additional public key and private key, any DL-based standard signature scheme

can be proven as secure as the decisional Diffie-Hellman (DDH) problem in the random oracle model (ROM). They also exploited the collision-resistant property of cryptographic hash function to prove the security of any RSA-based standard signature scheme with tight security reduction. Combining the Goh et al.'s proving technique and the Kurosawa-Heng transformation framework (Kurosawa and Heng, 2004), one can trivially obtain from Goh et al. standard signature scheme an IBI which is secure against impersonation under passive attack in the ROM but the security against active and concurrent attacks is not known for the transformed scheme.

In the same year of Goh et al.'s work, Arita and Kawashima (2007) proposed a variant of Schnorr standard identification scheme which is secure against impersonation under passive attack based on the DLOG assumption and knowledge-of-exponent (KEA1) without random oracle. They also proved that the scheme is secure against impersonation under active and concurrent attacks based on the one-more DL (OMDL) assumption and KEA1 without random oracle. These proofs can achieve tight security reduction by eliminating the need of rewinding the adversary as in Reset Lemma (Bellare and Palacio, 2002) where the simulator can open the commitment from two items which are the simulator transcript and the non-black-box extractor of KEA1. According to the Bellare et al. transformation framework (Bellare et al., 2009), the Schnorr standard identification (Schnorr, 1990) and its variants is not captured by any category of the convertible standard identification scheme. Thus, it indicates that even though we can transform the Arita and Kawashima's standard identification scheme to

an IBI, we still need to provide a direct proof for it.

The question of whether a transformed IBI from either standard signature or identification scheme is secure against active and concurrent attacks has been answered in the Yang et al. IBI framework (Yang et al., 2008). The framework shows that if there exists trapdoor strong-one-more relation and witness dualism proof of knowledge in an IBI scheme, then the IBI scheme is secure against active and concurrent attacks. This was further illustrated by constructing an IBI scheme using the Goh et al. signature (Goh et al., 2007) scheme as the building block. However, the resulted security reduction is also not tight as rewinding of adversary is needed in the security proof to extract a witness.

In this section, inspired by the technique of Goh et al. (2007), we propose a variant of Schnorr IBI which requires only three additional elements in the system parameters for achieving tight security reduction. Besides, the proposed scheme can be proven secure against impersonation under passive, active and concurrent attacks based on the decisional Diffie-Hellman (DDH) assumption in ROM. This technique may be applied on other IBI schemes as well.

4.1.2 Schnorr IBI

We first give the construction of the Schnorr IBI scheme which is a transformation from standard Schnorr signature (Schnorr, 1990) scheme by using the Kurosawa and Heng transformation (Kurosawa and Heng, 2004):

Setup. On input 1^k, generate two large primes p and q such that $q|(p-1)$. Choose a random $s \in \mathbb{Z}_q$ to compute $v = g^{-s}$ where $g \in \mathbb{G}$. Let the master public key be $mpk = (p, q, g, v, H)$ and the master secret key be $msk = s$ where $H : \{0,1\}^* \times \mathbb{G} \times \mathbb{G} \to \mathbb{Z}_{2^{l(k)}}$ for super-logarithmic challenge length $l : \mathbb{N} \to \mathbb{N} < \log(q)$.

Extract. Given ID, randomly select $t \in \mathbb{Z}_q$ to compute $X = g^t$ and $Y = t + s\alpha$ where $\alpha = H(ID, X, v)$. Set the user private key as $upk = (\alpha, Y)$.

Identification Protocol.

1. $\mathcal{P}$ first computes $X = g^Y v^\alpha$. $\mathcal{P}$ next chooses a random value $r \in \mathbb{Z}_q$, computes $R = g^r$ and sends (X, R) to $\mathcal{V}$.
2. $\mathcal{V}$ chooses a random value $c \in \mathbb{Z}_{2^{l(k)}}$ and sends c to $\mathcal{P}$.
3. $\mathcal{P}$ computes $y = r + cY$ and sends y to $\mathcal{V}$.
4. $\mathcal{V}$ accepts if and only if $g^y = R(X/v^\alpha)^c$ where $\alpha = H(ID, X, v)$.

Correctness:

$$\begin{aligned} R(X/v^\alpha)^c &= g^r(g^Y g^{-s\alpha}/g^{-s\alpha})^c \\ &= g^r(g^Y)^c \\ &= g^{r+cY} \\ &= g^y \end{aligned}$$

If the equality holds, output **1 (accept)**, else output **0 (reject)**.

4.1.3 A Variant of Schnorr IBI

The construction is similar to the works (Goh et al., 2007) which aim to eliminate the use of forking lemma in proving the security of signature schemes. The difference of this variant and the original scheme in Section 4.1.2 are: 1) two additional elements of $\mathbb{G}$ in mpk; 2) one additional element of $\mathbb{G}$ in msk. We now show the variant of Schnorr IBI which takes three additional elements in the system parameters as follows:

Setup. On input 1^k, generate two large primes p and q such that $q|(p-1)$. Choose $x \in \mathbb{Z}_q$ to compute, $,y_1 = g^{-x}$ and $y_2 = h^{-x}$, where $g,h \in \mathbb{G}$. Let the master public key be $mpk = (p,q,g,h,y_1,y_2,H)$ and the master secret key be $msk = x,z$ where $H : \{0,1\}^* \times \mathbb{G} \times \mathbb{G} \times \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{Z}_{2^{l(k)}}$ for super-logarithmic challenge length $l : \mathbb{N} \rightarrow \mathbb{N} < \log(q)$.

Extract. Let ID be the public identity. Select a value $t \in \mathbb{Z}_q$, compute $A = g^t, B = h^t$ and $s = t + x\alpha$, where $\alpha = H(ID,A,B,y_1,y_2)$. Return the user secret key as $upk = (\alpha,s)$.

Identification Protocol.

1. Firstly, $\mathcal{P}$ computes $A = g^s y_1^{\alpha}$ and $B = h^s y_2^{\alpha}$. Next, $\mathcal{P}$ chooses $r \in \mathbb{Z}_q$, then $\mathcal{P}$ computes $X = g^r$ and sends (A,B,X) to $\mathcal{V}$.

2. $\mathcal{V}$ chooses $c \in \mathbb{Z}_{2^{l(k)}}$ and sends c to $\mathcal{P}$.

3. $\mathcal{P}$ computes $y = r + cs$ and sends y to $\mathcal{V}$.

4. $\mathcal{V}$ accepts if and only if $g^y = X \cdot (A/y_1^{\alpha})^c$, where $\alpha = H(ID, A, B, y_1, y_2)$.

Correctness:

$$\begin{aligned} X(A/y_1^{\alpha})^c &= g^r(g^s g^{-x\alpha}/g^{-x\alpha})^c \\ &= g^r(g^s)^c \\ &= g^{r+cs} \\ &= g^y. \end{aligned}$$

If the equality holds, output **1 (accept)**, else output **0 (reject)**.

Notice that the value B is not involved in the verification of prover's response and this is due to the reason that the value B is bonded with upk by a secure hash function H. Recall that the normal way of proving the security of an IBI is by exploiting the proof of knowledge (POK) of a discrete logarithm in the **Identification Protocol** using Reset Lemma (Bellare and Palacio, 2002) (which resulted in non-tight reduction). The trick of our proof is that we are getting the prover I to prove that y (and subsequently s) contains the information of the discrete logarithm of y_1 and y_2, instead of proving the knowledge of the discrete logarithm and this does not require the help of B during the interaction. Besides, since the hash function H in **Extract** binds B with the scheme parameters, there is no way an adversary makes the verifier outputs **1** with an altered B value except with negligible probability (e.g. collision in H). We can save some complexity by not adding the value B and the protocol is still as secure as the

original Schnorr IBI scheme.

4.1.4 Security Analysis

We show that the Schnorr IBI variant remains secure with the additional elements and at the same time achieves tight reduction.

4.1.4.1 Security against Impersonation under Passive Attack

Recall that in the security model of IBI in Section 2.4.1, the challenger $\mathcal{M}$ setups the scheme's parameters and interacts with the impersonator I to solve underlying hard problem. I is allowed to query for upk and also the protocol transcript on an ID of its choice in the passive attack. If $\mathcal{M}$ fails to answer any of the queries, the IBI scheme is not proven secure.

Theorem 4.1. *The Schnorr IBI variant is $(t, q_e, \varepsilon_{IBI})$-secure against impersonation under passive attack in the random oracle model if the decisional Diffie-Hellman assumption (DDH) holds such that:*

$$t \leq t' - 2.4(q_e + 1)t_{exp}$$
$$\varepsilon_{IBI} \geq \varepsilon_{DDH} + 2(q_e + 1)q^{-1}$$

where q_e is the total extract queries that are queried by an impersonator I and assuming a two-exponent multi-exponentiation takes time $1.2t_{exp}$.

Proof. Given the tuple (g, h, y_1, y_2) as input, we construct an algorithm $\mathcal{M}$ running in time t' that can determine whether the given tuple is a DH tuple or not with the help of the impersonator I.

Phase 1

Setup. $\mathcal{M}$ sets $mpk = (p, q, g, h, y_1, y_2, H)$, where H is used as an random oracle which takes five inputs $H : \{0,1\}^* \times \mathbb{G} \times \mathbb{G} \times \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{Z}_{2^{l(k)}}$.

Extract Query. For an extract query on ID queried by I, $\mathcal{M}$ selects $s \in \mathbb{Z}_q$ and $\alpha \in \mathbb{Z}_{2^{l(k)}}$ to compute $A = g^s y_1^{\alpha}$ and $B = h^s y_2^{\alpha}$. $\mathcal{M}$ sets $\alpha = H(ID, A, B, y_1, y_2)$ and returns $upk_{ID} = (s, \alpha)$ to I. Notice that two additional two-exponent multi-exponentiations are required and this leads to the additional complexity of $2.4t_{exp}$ per upk_{ID} extraction (Goh et al., 2007, Arita and Kawashima, 2007).

Transcript Query. For a transcript query on ID queried by I, $\mathcal{M}$ first checks if ID has been queried an extract query before. If yes, $\mathcal{M}$ uses the existing upk_{ID} to return a valid transcript for I. If no, $\mathcal{M}$ randomly selects $\alpha, c \in \mathbb{Z}_{2^{l(k)}}, y \in \mathbb{Z}_q^*$ and $A, B \in \mathbb{Z}_p^*$ to generate $X = g^y (A/y_1^{\alpha})^{-c}$. $\mathcal{M}$ sets $\alpha = (ID, A, B, y_1, y_2)$ and returns (A, B, X, c, y) to I as the transcript query.

Phase 2

I pretends to be a valid prover of an identity ID^* which has not been queried an extract query before and runs the identification protocol with $\mathcal{M}$. At the end of the identification protocol, $\mathcal{M}$ obtains a transcript (A, B, X, c, y) on ID^*. If the transcript is not valid, $\mathcal{M}$ aborts and it fails in the security game. Else if the transcript is valid, $\mathcal{M}$ can determine whether the given tuple is a DH tuple and wins in the security game with the probability as follows:

$$\begin{aligned}
&\Pr[M \text{ wins}] \\
&= \Pr[M \text{ accepts prover}] - \Pr[M \text{ aborts if DH tuple}] - \Pr[M \text{ not aborts if random tuple}] \\
&\leq \varepsilon_{IBI} - \Pr[M \text{ aborts if DH tuple}] - \Pr[M \text{ not aborts if random tuple}].
\end{aligned}$$

We first examine the probability that $\mathcal{M}$ aborts if the tuple is a DH tuple. If it is a DH tuple, $\mathcal{M}$ simulates the IBI perfectly except with the negligible probability

$2^{-l(k)}$ of collision in H each time answering I's extract queries for q_e+1 times. Secondly, if the tuple is a random tuple, $\mathcal{M}$ does not abort with probability q^{-1} each time answering I's extract queries for q_e times and does not abort with probability q^{-1} when I made a call to the random oracle H in Phase 2. The probability of not abort depends on the value s such that there is at most one possible value of α for which there exist a s satisfying $A = g^s y_1^\alpha$ and $B = h^s y_2^\alpha$.

Combining these probabilities together, we get:

$$\Pr[M \text{ wins}] \leq \varepsilon_{IBI} - (q_e+1)2^{-l(k)} - (q_e+1)q^{-1}$$

$$\varepsilon_{DDH} \leq \varepsilon_{IBI} - 2(q_e+1)q^{-1}.$$

□

4.1.4.2 Security against Impersonation under Active and Concurrent Attacks

In the active and concurrent attacks, $\mathcal{M}$ have to answer the identification queries instead of transcript queries from I, which is now a cheating verifier.

Theorem 4.2. *The Schnorr IBI variant is (t, ε_{IBI})-secure against impersonation under active and concurrent attack in the random oracle model if the decisional Diffie-Hellman assumption (DDH) holds such that:*

$$t \leq t' - 2.4(q_e+1)t_{exp},$$

$$\varepsilon_{IBI} \geq \varepsilon_{DDH} + 2(q_e+1)q^{-1},$$

where q_e is the total extract queries that are queried by an impersonator I and assuming a two-exponent multi-exponentiation takes time $1.2t_{exp}$.

Proof. Given the tuple (g, h, y_1, y_2) as input, we construct an algorithm $\mathcal{M}$ running in time t' that can determine whether the given tuple is a DH tuple or not

with the help of the impersonator I.

Phase 1

Setup and **Extract Query** are the same as Section 4.1.4.1.

Identification Query. For an identification query on ID by I, $\mathcal{M}$ first checks if ID has been queried an extract query before. If yes, starting with the clone $m = 1$, $\mathcal{M}$ to return a valid transcript for I using upk_{ID}. Else, $\mathcal{M}$ plays the role of prover as in the identification protocol starting with the clone $m = 1$:

1. $\mathcal{M}$ randomly selects $\alpha, s \in \mathbb{Z}_q^*$ to compute $A = g^s y_1^\alpha, B = h^s y_2^\alpha$ and sets $\alpha = H(ID, A, B, y_1, y_2)$. $\mathcal{M}$ chooses $r \in \mathbb{Z}_q$, computes $X = g^r$ and sends (A, B, X) to I.
2. I sends $c \in \mathbb{Z}_q$ to $\mathcal{M}$.
3. $\mathcal{M}$ computes $y = r + cs$ and sends y to I.
4. $\mathcal{M}$ increases $\mathcal{M}$ by 1.

Phase 2

I pretends to be a valid prover of an identity ID^* which has not been queried an extract query before and runs the identification protocol with $\mathcal{M}$. At the end of the identification protocol, $\mathcal{M}$ obtains a transcript (A, B, X, c, y) on ID^*. If the transcript is not valid, $\mathcal{M}$ aborts and it fails in the security game. Else if the transcript is valid, $\mathcal{M}$ can determine whether the given tuple is a DH tuple and wins in the security game with the same probability as the proof in Section 4.1.4.1. If there remain any unanswered query of other clone $\mathcal{M}$, $\mathcal{M}$ reacts as in the identification query algorithm.

Up to date, the proof of security against active and concurrent attacks for IBI must involve one-more hard problems and Reset Lemma but we manage to avoid that. Reset Lemma is not applied here due to the fact that $\mathcal{M}$ is not finding the discrete logarithm of any system parameters (h, y_1, y_2, A, B, X). Instead, $\mathcal{M}$ is only deciding whether the discrete logarithm of y_1 is the same with y_2. Similar to the work in (Goh et al., 2007), this shows that a tighter security reduction can be achieved by basing the security of a scheme on a stronger assumption. □

4.1.4.3 Reset Attacks

The resettable attacks (Bellare et al., 2001) grant an adversary the power of resetting the identification protocol to any state it wants. It is obvious that the proposed Schnorr IBI variant which is applying the zero-knowledge proof of knowledge protocol cannot resist such attack. When the adversary pretends to be a verifier is running an identification protocol with a prover, after receiving the prover's response y_1 on the challenge c_1, it can reset the prover (which is normally a smart card) back to the state where the prover has just sent the commitment A, B, X (by terminating the power supply to smart card). The adversary will now send a second challenge c_2 to the prover and get the second response y_2 from prover. At the end, the adversary can compute the prover's secret such that $s = (y_2 - y_1)/(c_2 - c_1)$.

However, we are not considering resettable attacks in this work and we argue that such attack is an implementation issue which can be prevented. For instance, we can program the prover to delete the commitment value r from the

memory before sending the response y to the verifier. In this case, whenever the prover is reset, the identification protocol cannot be completed due to the fact that the prover no longer knows r and will abort during the computation of response (i.e., null value error).

4.2 Technique 2: Easy *ID*

Recall that some transformations frameworks for IBI schemes existed (Kurosawa and Heng, 2004, Bellare et al., 2009, Yang et al., 2008) and they can transform a standard digital signature scheme or identification scheme to an IBI scheme. These transformation frameworks also presented some IBI schemes which are not (fully) captured such as Okamoto IBI[1], BNN IBI and Schnorr IBI schemes. In such cases, direct proofs in the random oracle model (ROM) (Bellare and Rogaway, 1993) were provided.

We may always utilize the current transformation frameworks to evaluate the security of an IBI but there are some limitations. Firstly, the frameworks are not applicable if the underlying signature scheme or standard identification scheme is not covered by the frameworks. Secondly, some tweakings are normally needed in the process of crafting the security proofs and thus fall back to the direct proofs (Chin et al., 2008, Fujioka et al., 2012*a*, Tan et al., 2013). As an instance, the classic Schnorr IBI which is one of the few efficient IBI schemes is not captured by the existing transformation frameworks (Bellare et al., 2009,

[1]Okamoto IBI was not captured by Bellare et al.'s framework but it was later captured by Yang et al.'s framework. The remaining uncaptured schemes are BNN IBI (Bellare et al., 2009) and Schnorr IBI (Kurosawa and Heng, 2004).

Kurosawa and Heng, 2004, Yang et al., 2008) as it does not fulfil the first condition. Therefore, it is not possible to evaluate the security of such schemes via transformation framework and direct proofs are always needed either for standard identification schemes (Bellare and Palacio, 2002, Arita and Kawashima, 2007, Anada and Arita, 2011) or IBI schemes (Kurosawa and Heng, 2005, 2006, Chin et al., 2008, Thorncharoensri et al., 2009, Rckert, 2010, Fujioka et al., 2012*a*) to be proven secure, especially with a tight one.

In this section, we propose a proving technique which can be used to prove the security of the Schnorr IBI scheme with tight reduction based on the intractability of the OMDL problem without the need to amend the Schnorr IBI. Besides, we show that IBI schemes provably secure under the proposed technique is remain secure under the full *ID* technique (e.g., *ID*s of every information level will be queried) without security loss. Table 4.1 summarises the security tightness of existing security proofs for Schnorr IBI and its variants:

Table 4.1: Security Tightness of Schnorr IBI and Its Variants

Scheme	Passive	Active & Concurrent
Bellare et al. (2009)	$\sqrt{q_{pool} \cdot \varepsilon_{DL}} + \sqrt{\varepsilon_{sig}}$	$\sqrt{q_{pool} \cdot \varepsilon_{OMDL}} + \sqrt{\varepsilon_{sig}}$
Kurosawa and Heng (2004)	$\sqrt{e(1+q_e)\varepsilon_{DL}}$	$\sqrt[k]{e(1+q_e)\varepsilon_{DL}}$
Technique 1	ε_{DDH}	ε_{DDH}
Technique 2	ε_{OMDL}	ε_{OMDL}

where q_{pool} is the pool size of *ID*, e is the natural exponent, q_e is the total extract queries made and k is the bit length of the order in the underlying finite field.

In particular, we can overcome these two security losses in the security proofs of the Schnorr IBI scheme. Recall that security losses in security proofs lead to non-tightness in terms of the probability of simulator aborting the security game:

1. Simulator ($\mathcal{M}$) is unable to answer all extract queries of adversary (I).
2. Simulator ($\mathcal{M}$) needs to reset adversary (I) during impersonation.

The first loss is due to the reason that the challenge value $X = g^x$ which $\mathcal{M}$ needs to solve is crafted into the hash value α of an ID such that $\alpha = H(ID, X, v)$ where $v = g^{-s}$ is the master public key. If I issues extract query on the ID for user private key $Y = x + s\alpha$, $\mathcal{M}$ does not know the discrete logarithm x and has to abort the security game. In Bellare et al. (2009), the extract queries were answered in such a way that the ID queried must be coming from an ID pool and so the probability to abort is 1 over the pool size. On the contrary, Kurosawa and Heng used coin tossing technique (Kurosawa and Heng, 2004) in which the adversary's choice of ID is represented by a probability σ. In our proof, we can allow $\mathcal{M}$ to answer extract queries of any ID by using the $I\mathcal{NV}$ oracle of OMDL to output the discrete logarithm of a challenge value (from $\mathcal{CHALL}$ oracle) which resides in the corresponding hash values. This resulted the same situation as in the proofs of Technique 1, where hash query on every ID can be answered successfully and $\mathcal{M}$ does not need to abort the security game except with the negligible probability of hash collision.

The second loss is due to the nature of the zero knowledge protocol in Schnorr IBI where one must reset the protocol for extracting the secret DL value. The works in Bellare et al. (2009), Kurosawa and Heng (2004) but not Technique 1 suffer from this because the underlying hard problems used in their proofs are DL problem and OMDL problem while Technique 1 uses DDH which does not need to output the secret exponent. Since we are using OMDL, we cannot enjoy the flexibility of DDH and must reset the protocol during security proof also.

In order to avoid the loss, we examine the need to reset the adversary in Phase 2. We found that reset is redundant for the *ID* queried for transcript in the passive attack, and also for identification query in the active and concurrent attacks. We term such *ID* as the "easy *ID*" for they leak more information to an adversary than the unquried *ID*. Attacking easy *ID* is the most powerful attack of an adversary and so, only the easy *ID* cases need to be considered in the security proofs. If an adversary fails to impersonate an easy *ID*, it implies that the adversary will fail also, to impersonate other *ID* that leak lesser information.

4.2.1 Related Works

Technique 1 is the tightest reduction of Schnorr-based IBI schemes to date where the IBI scheme is based on the hardness of the decisional Diffie-Hellman (DDH) problem instead of DL or OMDL[2] problem as in Kurosawa and

[2]The security proofs of concurrent attacks in the works of Kurosawa and Heng (2004), Bellare et al. (2009) were based on OMDL but their simulators can only answer hash queries selectively and need to use Reset Lemma. The same goes for standard Schnorr identification scheme (Bellare and Palacio, 2002).

Heng (2004), Bellare et al. (2009). The easier DDH problem allows the simulator to win the security game by answering all hash queries from the adversary besides not resetting the impersonator and thus the tight reduction. This is because the simulator only needs to decide whether the given tuple is a DH tuple instead of finding the discrete logarithm as required by the harder DL problem. The only drawback is that the variant requires additional elements in the public key as compared to the original Schnorr IBI in achieving the tight reduction.

Recently, Fujioka et al. (2012*b*) showed that if an IBI scheme is proven secure against impersonation under passive attack, it can be upgraded to be secure against impersonation under active and concurrent attacks. They showed that by using either dual identities, master identity or double parameters, the OR-proof can be applied on IBI schemes to enhance the security as claimed. Besides, the resulted IBI is secure under harder intractable problems which makes the proof more convincing. However, this technique needs to double up the key size and the complexity of protocol in addition to non-tight reduction.

To summarise briefly, the security enhancements of an IBI scheme can be done either by applying stronger assumptions as done in Technique 1 and Arita and Kawashima (2007) or enlarging the schemes' construction domain (Fujioka et al., 2012*b*,*a*). These provide flexibilities to the simulator in answering the impersonator's queries. Notice that only the former can provide tight reduction, i.e., when using easier intractable problems, while the latter cannot.

4.2.2 Security Analysis

Although the hash value of an ID can be computed by anyone in the Schnorr IBI, such freedom is available only after the transcript of ID is known to the public. Notice that the value X is a private key element even though it is known after a single run of the identification protocol. So, the user has no say in choosing X during the issuing of extract and hash queries. In fact, if a user can choose his own $X = g^x$, he can extract the PKG's master secret key by calculating $s = (Y - x)/\alpha$. Thus, the adversary in the security game also cannot choose its own X when making a query to the hash and extract oracles.

This restriction is important to our technique as X can now be used as the challenge value to be solved by the simulator $\mathcal{M}$ in Phase 2 for an easy ID. If X can be freely chosen by the adversary, even though finally $\mathcal{M}$ solves the discrete logarithm problem of X, it is not a challenge produced by the $\mathcal{CHALL}$ oracle and so $\mathcal{M}$ is not considered to have solved the OMDL problem. We now explain in detail what is easy ID and how to fully utilize the two oracles from the OMDL problem in helping $\mathcal{M}$ to achieve tight reduction.

4.2.2.1 Easy ID

The easy ID literally means an ID which is easier to be impersonated compared to other IDs. These easy IDs can be identified based on several types of information provided by the simulator to an adversary for impersonation.

This is inspired by the work of Numayama et al. (2008) which categorised variants of the random oracle model based on the strength of additional oracles. Different from their work, we categorise *only* the information provided by the random oracle itself according to the types of impersonation attack.

Table 4.2: Information Provided by Simulator to Adversary

Sensitivity Level	Information	Obtained From	Attack Type
Level 0	Null	Null	All
Level 1	Hash(*ID*)	Hash Oracle	All
Level 2	Transcript(*ID*)	Transcript Oracle	Passive
Level 3	Identification(*ID*)	Identification Oracle	Active & Concurrent
Level 4	Extract(*ID*)	Extract Oracle	All

These information help the simulator to identify the *ID* that an adversary *wants* to impersonate and subsequently helps the simulator in binding the underlying hard problem to the targeted *ID* to result in a tight reduction. One may argue that this assumption is not appropriate as an adversary should be viewed as a black box and its choice of *ID* for impersonation should not be controlled by the simulator. Note that this argument is not relevant as we do not need to determine which *ID* will be chosen by the adversary; we only need to know when the adversary reaches the strongest state. In the real world, if there exists an *ID* which is relatively easier to be impersonated compared to others, the *ID* will likely be the adversary's target, i.e., the adversary is more powerful when doing so. In the security proofs, the simulator will encounter such a scenario of providing more information for certain *ID* through hash, transcript and identification queries. So, these *ID*s which have been queried are relatively easier to be impersonated compared to others whose information are limited.

Table 4.2 shows according to sensitivity level, the information of an *ID* which can be provided by the simulator. The case where the user private key is known will not be considered because an adversary does not need to impersonate a user whose user private key is known. Thus, referring to Table 4.2 the information amount of an *ID* can be arranged from Level 0 to Level 3.

During impersonation, an adversary stands the highest chance to successfully impersonate an *ID* when it holds the largest information amount of the *ID* such as information of Level 2 and 3. In other words, the targeted *ID* is relatively easier to be impersonated compared to other if such information amount are obtained by the adversary. This happens when the targeted *ID* is previously queried to the simulator in hash queries and either transcripts or identification queries.

In order to define the relations, let $\mathbf{A}$ be an attack and I be an adversary. Let $\mathbf{A}(I(\text{Level } i)) \Rightarrow \mathbf{A}(I(\text{Level } j))$ and $\mathbf{A}(I(\text{Level } i)) \not\Rightarrow \mathbf{A}(I(\text{Level } j))$ be as follows:

- $\mathbf{A}(I(\text{Level } i)) \Rightarrow \mathbf{A}(I(\text{Level } j))$: if Schnorr IBI resists an attack $\mathbf{A}$ with information amount Level i provided to I, then it also resists $\mathbf{A}$ with the information amount Level j provided to I
- $\mathbf{A}(I(\text{Level } i)) \not\Rightarrow \mathbf{A}(I(\text{Level } j))$: Schnorr IBI resists an attack $\mathbf{A}$ with information amount Level i provided to I, but it does not resist $\mathbf{A}$ with the information amount Level j provided to I

Now it is obvious that the following relations hold based on the information amount for any attacks (passive, active and concurrent attacks):

$$\mathbf{A}(I(\text{Level } 0)) \Leftarrow \mathbf{A}(I(\text{Level } 1)) \Leftarrow \mathbf{A}(I(\text{Level } 2)) \Leftarrow \mathbf{A}(I(\text{Level } 3))$$

and

$$\mathbf{A}(I(\text{Level } 0)) \not\Leftarrow \mathbf{A}(I(\text{Level } 1)) \not\Leftarrow \mathbf{A}(I(\text{Level } 2)) \not\Leftarrow \mathbf{A}(I(\text{Level } 3))$$

If an ID is of Level 2 or 3, the hash value α of ID already exists, as well as the element X. Thus, during impersonation in Phase 2, X will be reused in the commitment but R can be a new random value. However, again, note that we only need to prove the security of Schnorr IBI based on the strongest attack, which would imply the reuse of the commitment value R because the adversary's success probability is lower when R is not reused as shown in Table 4.3. Since $l(k) < \log(q)$, we can see that:

$$\begin{aligned} &\Pr[I \text{ can impersonate } ID' | \text{New } R] \\ &< \Pr[I \text{ can impersonate } ID' | \text{Reused } R] \\ &\leq \varepsilon_{IBI} \end{aligned}$$

Table 4.3: Adversary's Success Probability in Schnorr IBI

R	$c \xleftarrow{\$} \mathbb{Z}_{2^{l(k)}}$	$\Pr[I \text{ can impersonate}]$	Remarks
Reused	New	$1/q$	I correctly computes r, y or Y
	Reused	$1/2^{l(k)}$	I receives the same c
New	New	$1/q$	I correctly computes y or Y
	Reused	$1/q$	I correctly computes y or Y

Therefore, we only need to focus on the impersonation of *ID* of information Level 2 and 3 with respect to the reused of (X,R) in passive attack and active & concurrent attacks, where we show that if there exists for Schnorr IBI an adversary which can impersonate an *ID* with such information amount, then there exist an algorithm which can break the OMDL problem.

4.2.2.2 From Easy *ID* to Full *ID*

The question remain is how weak the scheme provably secure under Technique 2 is, compared to those provably secure under the full *ID* technique. We answer this question positively that the Technique 2, namely, easy *ID* technique implies the full *ID* technique. Surprisingly, there is no security loss in extending the security implication of the easy *ID* technique to the full ID technique. We will show only the security against impersonation under active and concurrent attacks as this security notion implies the security notion under passive attack. The proving methodology is similar to that used by Kurosawa and Heng (2004) in showing the relation of security notions between standard digital signature scheme and IBI scheme.

Theorem 4.3. *Let $\mathcal{IBI}$ be an identity-based identification scheme which is secure against impersonation under active and concurrent attacks using easy ID technique. Then, $\mathcal{IBI}$ is also secure against impersonation under active and concurrent attacks using full ID technique.*

Proof. Assume there exists an impersonator I_{full} which can (t,ε)-break the $\mathcal{IBI}$ scheme using full *ID* technique, then there exists an impersonator I_{easy} which can (t',ε')-break the $\mathcal{IBI}$ using easy *ID* technique. We show how to build the

impersonator I_{easy} with the help of I_{full} such that:

$$\varepsilon' = \varepsilon, t' \leq t$$

Setup. I_{easy} obtains the mpk and passes it to I_{full}.

Phase 1

Hash Queries. If I_{full} issues a hash query on ID, I_{easy} issues a hash query on ID to its underlying Hash Oracle. I_{easy} then forwards the answer α to I_{full}.

Extract Queries. If I_{full} issues an extract query on ID, I_{easy} issues an extract query on ID to its underlying Extract Oracle. I_{easy} then forwards the answer upk to I_{full}.

Identification Queries. We assume without loss of generality that I_{full} will not issue an identification query on an identity that it has already issued an extract query on. When I_{easy} forwards the identification query on ID from I_{full} to the underlying Identification Oracle, the identification session between the oracle with I_{easy} will be started. Staring with the clone $m = 1$, I_{easy} simulates a cheating prover for I_{full} on the identity ID_j as follows:

1. I_{easy} forwards the commitment $CMT = (X_j, R_{j,m})$ received from oracle to I_{full}.
2. I_{full} selects $c_{j,m} \xleftarrow{\$} \{1,0\}^{l(k)}$ and sends $c_{j,m}$ to I_{easy}. I_{easy} forwards $c_{j,m}$ to the oracle.
3. The oracle returns $y_{j,m}$ and I_{easy} forwards $y_{j,m}$ to I_{full}.
4. I_{easy} increments $\mathcal{M}$ by 1 whenever an identification session is initiated by the Identification Oracle.

Phase 2

Impersonation. I_{full} can still issue some extract and identification queries as in Phase 1. I_{full} now plays the role of the cheating prover trying to convince I_{easy} that it knows the user private key of the identity ID^* which I_{full} has not yet issued an extract query on.

As I_{full} is from the full ID technique, it will attack *any* ID which includes those non-easy ID. In this case, I_{full} will not necessary reuse the α^* and the commitment values (X^*, R^*) from hash query and identification queries respectively. Note that it is possible for I_{full} to compute its own R^* though doing so will decrease its advantage in breaking the $I\mathcal{B}I$, i.e., performing a less powerful attack. In short, we have to consider the impersonation carried out by I_{full} using both reused or freshly computed (X^*, R^*).

However, despite the freshness of (X^*, R^*), if the response value y^* produced by I_{full} is valid, I_{easy} can directly use y^* to impersonate $I\mathcal{B}I$ successfully. Thus, it is clear that both impersonators I_{full}, I_{easy} take the similar[3] time in impersonation attempt and their advantages in breaking $I\mathcal{B}I$ are the same. □

At the first glance, the extension from easy ID to full ID should cause a security loss of k^2 as simulator $\mathcal{M}$ have to reset the impersonator when the non-easy ID is challenged in Phase 2. However, the proof shows that the role of $\mathcal{M}$ is played by the oracles that I_{easy} can get access to and the security loss of $\mathcal{M}$ breaking the underlying hard problem is hidden from the security proof. Thus, we can claim that an IBI scheme which is provably secure using the easy ID technique is also provably secure using the full ID technique and vice versa.

[3] $t' \leq t$ as a very little amount of times are lost during message forwarding.

4.2.2.3 Security against Impersonation under Passive Attack

We are now ready to present the usage of easy *ID* in achieving tight reduction for passive attack proof of Schnorr IBI scheme.

Theorem 4.4. *The Schnorr IBI scheme is secure against impersonation under passive attack (imp-pa) if the one-more discrete logarithm (OMDL) problem is hard.*

Proof. Assume there exists an impersonator I who can $(t_{IBI}, \varepsilon_{IBI})$ break the Schnorr IBI scheme, then there exists an algorithm $\mathcal{M}$ which $(t_{OMDL}, \varepsilon_{OMDL})$ solves the one-more discrete logarithm (OMDL) problem. $\mathcal{M}$ will be given a cyclic group $\mathbb{G}$ and the parameters $q, p, g \in \mathbb{G}$. We show how to build the algorithm $\mathcal{M}$ with the help of I such that:

$$\varepsilon_{IBI} \leq \varepsilon_{OMDL}$$

$$t_{IBI} = t_{OMDL} + 3t_{add} + 2t_{mul} + t_{inv}$$

where t_{add}, t_{mul} and t_{inv} is the time needed to compute in $\mathbb{G}$ an addition, multiplication and inverse respectively.

Setup. Same as the original algorithm described in Section 4.1.2.

Phase 1

Hash Queries. If the record (ID_j, X_j) exists, $\mathcal{M}$ replies using this existing record. Else, $\mathcal{M}$ asks for $X_j = g^{x_j}$ from $\mathcal{CHALL}$ and add X_j to the list before returning $\alpha_j = H(ID_j, X_j, v)$ as the hash value of ID_j.

Extract Queries. When I queries $\mathcal{M}$ for an identity ID_j, if ID_j is not queried before, $\mathcal{M}$ issues a hash query for ID_j followed by issuing X_j to $I\mathcal{NV}_{\mathcal{DL}}$ asking for x_j. Using α_j and x_j, $\mathcal{M}$ computes the user private key (α_j, Y_j) for I using the original algorithm.

Transcript Queries. We assume without loss of generality that I will not issue a transcript query on an identity that it has already issued an extract query on. When I queries for a transcript on the identity ID_j, $\mathcal{M}$ issues a hash query for ID_j besides asking for a random value $R_j = g^{r_j}$ from $\mathcal{CHALL}$. Next, $\mathcal{M}$ selects $c_j \xleftarrow{\$} \{1,0\}^{l(k)}$ and queries $I\mathcal{NV}_{\mathcal{DL}}$ with $R_j(X_j/v^{\alpha_j})^{c_j}$ to get the value y_j. $\mathcal{M}$ returns $((X_j, R_j), c_j, y_j)$ as the transcript to I. $\mathcal{M}$ records the transcripts in its list.

Phase 2

Impersonation. I can still issue some extract and transcript queries as in Phase 1. I now plays the role of the cheating prover trying to convince $\mathcal{M}$ that it knows the user private key of the identity ID^* which I has not yet issued an extract query on.

Application of easy ID. Recall that for passive attack, we only need to consider the case of ID^* with information Level 2 where the ID^* is previously queried in hash query and transcript queries. In this case, I will reuse the α^* and randomly pick the commitment values (X^*, R^*) from previous hash queries and transcript queries respectively. Note that if I computes its own (X^*, R^*), its advantage in breaking the Schnorr IBI will be lesser, i.e., performing a less powerful attack. Thus, we only need to consider the most powerful attack carried out by I, which is the attack on easy ID^* with reused (X^*, R^*).

Subsequently, $\mathcal{M}$ is able to obtain a valid transcript $((X^*,R^*),c^*,y^*)$ by interacting with I as prover where $\mathcal{M}$ did not previously pair the values (R^*,c^*) in any single transcript query. Therefore, $\mathcal{M}$ can now extract the value $Y^* = (y^* - y_0)/(c^* - c_0) \bmod q$ where (c_0,y_0) are from the previous transcripts $((X^*,R^*),c_0,y_0)$ of ID^*. $\mathcal{M}$ solves the discrete logarithm problem of $X^* = g^{x^*}$ such that $x^* = Y^* - s\alpha^*$. Discrete logarithm of R^* can be solved by simply calculating either $r^* = y - c^*Y^*$ or $r^* = y_0 - c_0Y^*$.

For each identity $ID_j \neq ID^*$, $\mathcal{M}$ solved the two discrete logarithm problems $R_j = g^{r_j}$ and $X_j = g^{x_j}$ by making two queries to the $I\mathcal{NV}_{\mathcal{DL}}$ oracle. For the identity ID^*, $\mathcal{M}$ did not query the $I\mathcal{NV}_{\mathcal{DL}}$ oracle to compute the discrete logarithms x^*. Hence, $\mathcal{M}$ saves one $I\mathcal{NV}_{\mathcal{DL}}$ query and wins in the one-more discrete logarithm problem.

Since the original algorithms were used as in a real attack against Schnorr IBI, the cheating prover's and cheating verifier's environment are simulated perfectly. The advantage of $\mathcal{M}$ in solving the one-more discrete logarithm problem is:

$$
\begin{aligned}
\varepsilon_{OMDL} &= \Pr[M\text{computes } x_i] \\
&= \varepsilon_{IBI} - \sum_{n=1}^{n=q_h} \Pr[\text{Collision occurs at } H(ID_n)] \\
&= \begin{cases} \varepsilon_{IBI} - \left(\frac{q_h}{2^{l(k)}}\right)\frac{q_h-2}{2} + \frac{q_h/2}{2^{l(k)}} & \text{if } q_h \text{ is even} \\ \varepsilon_{IBI} - \left(\frac{q_h}{2^{l(k)}}\right)\frac{q_h-1}{2} & \text{if } q_h \text{ is odd} \end{cases} \\
&= \varepsilon_{IBI} - q_h(q_h-1)/2^{l(k)+1} \\
&\geq \varepsilon_{IBI}
\end{aligned}
$$

where ε_{IBI} is the probability of $\mathcal{M}$ having a successful identification protocol with I in Phase 2, q_h is the total number of hash queries throughout the game and $\Pr[\text{Collision occurs at } H(ID_n)]$ is the probability of collision occurs when an-

swering a hash query. Note that the probability is proportional to q_h as there will be more hash values to collide with when time goes by, such that the collision probability is having the series of: $0/2^{l(k)}, 1/2^{l(k)}, 2/2^{l(k)}, \ldots, (q_h-1)/2^{l(k)}$.

Since the probability of a hash collision to happen throughout the game is negligible[4], this shows that ε_{IBI} is bounded by ε_{OMDL} as required. After I outputs the transcript in Phase 2, $\mathcal{M}$ performed three additions, two multiplications and one inverse operations to solve the discrete logarithm of X_i. Thus, $\mathcal{M}$ has extra operation time of $3t_{add} + 2t_{mul} + t_{inv}$ compared to that of I. □

4.2.2.4 Security against Impersonation under Active and Concurrent Attacks

We now show the usage of easy *ID* in achieving tight reduction for active and concurrent attacks proof of Schnorr IBI scheme.

Theorem 4.5. *The Schnorr IBI scheme is secure against impersonation under active and concurrent attacks if the one-more discrete logarithm (OMDL) problem is hard.*

Proof. Assume there exists an impersonator I who can break the Schnorr IBI scheme, then there exists an algorithm $\mathcal{M}$ which solves the one-more discrete logarithm (OMDL) problem with the advantage ε_{OMDL}. $\mathcal{M}$ will be given a cyclic group $\mathbb{G}$ and the parameters $q, p, g \in \mathbb{G}$. We show how to build the algorithm $\mathcal{M}$ with the help of I such that:

$$\varepsilon_{IBI} \leq \varepsilon_{OMDL}$$

$$t_{IBI} = t_{OMDL} + 3t_{add} + 2t_{mul} + t_{inv}$$

[4]If SHA-256 is the hash function, $l(k) = 256$. Assuming we allow $q_h = 2^{30}$ hash queries, the probability for a collision to happen is $(2^{30})^2/2^{256} = 1/2^{196}$ which is still negligible.

where t_{add}, t_{mul} and t_{inv} is the time needed to compute in $\mathbb{G}$ an addition, multiplication and inverse respectively.

Setup. Same as described in Section 4.1.2.

Phase 1

Hash Queries and **Extract Queries** are the same as described in Section 4.2.2.3.

Identification Queries. We assume without loss of generality that I will not issue an identification query on an identity that it has already issued an extract query on. Starting with $m = 1$, $\mathcal{M}$ simulates a cheating prover for I for a round of interaction on the identity ID_j as follows:

1. $\mathcal{M}$ issues a hash query for ID_j besides asking for a random value $R_{j,m} = g^{r_{j,m}}$ from $\mathcal{CHALL}$. The produced commitment CMT is $(X_j, R_{j,m})$.
2. I selects $c_{j,m} \xleftarrow{\$} \{1,0\}^{l(k)}$ and sends $c_{j,m}$ to $\mathcal{M}$.
3. $\mathcal{M}$ queries $I\mathcal{NV}_{\mathcal{DL}}$ with $R_{j,m}(X_j/v^{\alpha_j})^{c_{j,m}}$ and gets the value $y_{j,m}$. $\mathcal{M}$ returns $y_{j,m}$ to I. Since $y_{j,m} = I\mathcal{NV}_{\mathcal{DL}}(R_{j,m}) + c_{j,m}Y_j$, this is exactly the response that the clone $\mathcal{M}$ would return to I.
4. $\mathcal{M}$ increments $\mathcal{M}$ by 1.

Phase 2

Impersonation. I can still issue some extract and identification queries as in Phase 1. I now plays the role of the cheating prover trying to convince $\mathcal{M}$ that it knows the user private key of the identity ID^* which I has not yet issued an extract query on.

Application of easy ID. Recall that for active & concurrent attacks, we only need to consider the case of ID^* with information Level 3 where the ID^* is previously queried in hash query and identification queries. In this case, I will reuse the α^* and randomly pick a commitment values (X^*, R^*) from hash query and identification queries respectively. Note that if I computes its own (X^*, R^*), its advantage in breaking the Schnorr IBI will be lesser, i.e., performing a less powerful attack. Thus, we only need to consider the most powerful attack carried out by I, which is the attack on easy ID^* with reused (X^*, R^*).

Subsequently, $\mathcal{M}$ is able to obtain a valid transcript $((X^*, R^*), c^*, y^*)$ by interacting with I as prover where the pair (R^*, c^*) did not appear before in any single identification query. Therefore, $\mathcal{M}$ can now extract the value $Y^* = (y^* - y_0)/(c^* - c_0) \bmod q$ where (c_0, y_0) are from the previous transcripts $((X^*, R^*), c_0, y_0)$ of ID^*. $\mathcal{M}$ solves the discrete logarithm problem of $X^* = g^{x^*}$ such that $x^* = Y^* - s\alpha^*$. Discrete logarithm of R^* can be solved by calculating either $r^* = y - c^* Y^*$ or $r^* = y_0 - c_0 Y^*$.

$\mathcal{M}$ continues to solve the discrete logarithm problems of the rest of the random values $R_{i,m} = g^{r_{i,m}}$ by calculating $R_{i,m} = y_{i,m} - c_{i,m} Y_i$. For other identities $ID_j \neq ID_i$ which has been issued an identification query on but not yet the extract query, $\mathcal{M}$ simply queries g^{x_j} to $I\mathcal{NV}_{\mathcal{DL}}$ for x_j. $\mathcal{M}$ then computes Y_j and find the discrete logarithms $r_{j,m} = y_{j,m} - c_{j,m} Y_j$.

For each identity $ID_j \neq ID_i$, $\mathcal{M}$ computes the discrete logarithm problems of $\mathcal{M}$ values $R_{j,m}$ and a value g^{x_j} by making $m+1$ queries to the $I\mathcal{NV}_{\mathcal{DL}}$ oracle. For the identity ID^*, $\mathcal{M}$ computes $m+1$ discrete logarithms by making only $\mathcal{M}$ queries to the $I\mathcal{NV}_{\mathcal{DL}}$ oracle. Hence, $\mathcal{M}$ saves one $I\mathcal{NV}_{\mathcal{DL}}$ query and wins in the one-more discrete logarithm problem. The probability analysis

of $\mathcal{M}$ wins the game is the same as in Section 4.2.2.3, thus this completes the proof. □

4.2.3 Discussion

We briefly describe the way $\mathcal{M}$ answers hash or extract queries in existing security proofs (Kurosawa and Heng, 2004, Bellare et al., 2009) and Technique 1 of Schnorr IBI in order to highlight the distinguishing features of Technique 2. The main trick in the previous proofs is the simulator $\mathcal{M}$ can answer the hash queries by reversing the output of hash function H. $\mathcal{M}$ fixes a random value α_j as the output of an identity ID_j before knowing the value of hash input X_j. If the record (ID_j, X_j, g^{Y_j}, α_j) exists, $\mathcal{M}$ replies using this existing record. Else, $\mathcal{M}$ queries for a random value g^{Y_j} from the $\mathcal{CHALL}$ oracle and selects $\alpha_j \xleftarrow{\$} \mathbb{Z}_{2^{l(k)}}$. Then $\mathcal{M}$ sets $X_j = g^{Y_j} v^{\alpha_j}$ and fixes $\alpha_j = H(ID_j, X_j, v)$ as the hash value of ID_j. The similar programming mechanism is done in answering the transcript queries under passive attacks.

On the other hand, in the proof of Technique 2, $\mathcal{M}$ only used $\mathcal{CHALL}$ as the seed for H and it did not reverse the procedure of hashing. Although $\mathcal{M}$ still needs to keep track of the hash input X_j for each identity ID_j, $\mathcal{M}$ does not need to keep track of the hash output. Since the value of α_j will be changed if the value of X_j is changed, the way $\mathcal{M}$ answers the hash queries in Technique 2 has no difference compared to the real algorithm, i.e., $\mathcal{M}$ does not program the hash function as in previous proofs. Moreover, we manage to avoid the Reset

Lemma in Phase 2 by examining only the *ID* of information Level 2 for passive attack and Level 3 for active & concurrent attacks. We summarize for Schnorr IBI scheme and its variants, the difference of existing techniques and ours in Table 4.4.

Table 4.4: Proving Techniques for Schnorr IBI Scheme and Its Variants

Query	Existing Techniques & Technique 1	Technique 2
Hash	Program the random oracle to fix a chosen hash value to an *ID*.	Choose hash input/output from $\mathcal{CHALL}$ oracle.
Extract	Program the random oracle to fix a chosen secret value to an *ID*.	Obtain the secret value form $\mathcal{INV}$ oracle.
Transcript	Program the random oracle to fix a commitment value to an *ID* for chosen challenge and response values.	Use $\mathcal{CHALL}$ and $\mathcal{INV}$ oracles to generate transcript.
Identification	Use $\mathcal{CHALL}$ and $\mathcal{INV}$ oracles to get commitment values and answer the challenge.	Use $\mathcal{CHALL}$ and $\mathcal{INV}$ oracles to get commitment values and answer the challenge.

If this technique is applied on RSA-based and CDH-based IBI, we can achieve tighter security reduction as $\mathcal{M}$ can exclude the artificial coin tossing in the security proof and answer all extract queries with the help of $\mathcal{CHALL}$ and $\mathcal{RSA}$ or $\mathcal{CDH}$ respectively. For instance, in the security proof of GQ-IBI (Bellare et al., 2009), $\mathcal{M}$ can answer the hash query on ID_j by asking a challenge value W_j from $\mathcal{CHALL}$ and set $H(ID_j) = W_j$. $\mathcal{M}$ keeps a list of (ID_j, W_j) pair in order to make sure the hash value for ID_j is always the same. Finally, the transcript returned by the I in Phase 2 can be used by $\mathcal{M}$ to avoid the Reset Lemma and solve the OM-RSA problems by examining only the cases of easy

ID for passive attack and active & concurrent attacks.

4.3 Schnorr-based FIBI Scheme with Tight Security Reduction

In this section, we show that Technique 2 can be applied on Tan et al.'s fuzzy identity-based identification scheme (Tan et al., 2009) and achieve tight reduction without any amendments on the scheme construction. Notice also the new proofs is not in the selective-ID model where the **Init** phase is not presented here. The reason behind this is due to the fact that Technique 2 subsumes the full *ID* model, as proven in Section 4.3. Since security notion of impersonation against active and concurrent attacks implies the security over passive attack, we show only the security proof of the former. Before going into the details, we briefly describe the FIBI scheme (Tan et al., 2009) as follows.

Setup. Except defining the threshold d, other are the same as the algorithm in Section 4.1.2.

Extract. Let *ID* be the set of n identities for some fixed n and d represents the distance metric of two identity sets. Randomly select a $(d-1)$-degree polynomial $q(\cdot)$ such that $q(0) = t \in \mathbb{Z}_q$. Compute the set $\{Y_i\} = \{q(i) + s\alpha_i\}_{i \in ID}$ and the set $\{\alpha_i\} = \{H(i, X, v)\}_{i \in ID}$ where $X = g^t$. Return the user private key as $upk = (\{\alpha_i\}, \{Y_i\})$.

Identification Protocol.

1. $\mathcal{P}$ first computes $\{X_i\} = \{g^{Y_i} v^{\alpha_i}\}_{i \in ID}$. $\mathcal{P}$ next chooses $\{r_i\}_{i \in ID} \in \mathbb{Z}_q$, computes $\{R_i\} = \{g^{r_i}\}_{i \in ID}$ and sends $(\{X_i\}, \{R_i\})$ to $\mathcal{V}$.
2. $\mathcal{V}$ chooses $c \in \mathbb{Z}_{2^{l(k)}}$ and sends c to $\mathcal{P}$.
3. $\mathcal{P}$ computes $\{y_i\} = \{r_i + cY_i\}_{i \in ID}$ and sends $\{y_i\}$ to $\mathcal{V}$.
4. $\mathcal{V}$ randomly chooses $S = \{ID \cap ID'\}$ where $|S| = d$ and for every $i \in S$, $\mathcal{V}$ checks if $g^{y_i} = R_i (X_i / v^{\alpha_i})^c$ where $\{\alpha_i\} = \{H(i, X, v)\}_{i \in S}$ and $X = \prod_S X_i^{\triangle_{i,S}(0)}$.

Correctness:

$$\begin{aligned} R_i (X_i / v^{\alpha_i})^c &= g^{r_i} (g^{Y_i} g^{-s\alpha_i} / g^{-s\alpha_i})^c \\ &= g^{r_i} (g^{Y_i})^c \\ &= g^{r_i + cY_i} \\ &= g^{y_i} \end{aligned}$$

If the equality holds, output **1 (accept)**, else output **0 (reject)**.

4.3.1 Security Model

We give the security notions for FIBI before proving its security. The impersonation attack game on an FIBI scheme between an impersonator I and a challenger $\mathcal{M}$ is described as a two-phased game in the selective-ID model (Tan et al., 2009) as follows:

1. **Init.** I declares the identity set ID' that it wishes to be challenged upon. Therefore, one identity set ID_i such that $|ID_i \cap ID'| \geq d$ will be under attack in the Phase 2 of the game.

2. **Setup.** $\mathcal{M}$ takes as input 1^k and runs the setup algorithm. It gives I the resulting master public key mpk and keeps the master secret key msk to itself.

3. **Phase 1**

   (a) I issues some extract queries on $ID_1, ID_2, \ldots$. The $\mathcal{M}$ responds by running the extract algorithm to generate the private key upk_i corresponding to the public identity set ID_i. It returns upk_i to I.

   (b) I issues some transcript queries for passive attack or some identification queries for active and concurrent attacks on ID_i such that $|ID_i \cap ID'| \geq d$.

   (c) The queries in step (a) and step (b) above can be interleaved and asked adaptively. Without loss of generality, we may assume that I will not query the same ID_i that has been issued in the extract queries, through the transcript queries or identification queries again.

4. **Phase 2**

   (a) I plays the role as a cheating prover (impersonation attempt on the prover holding a challenged identity set ID_i such that $|ID_i \cap ID'| \geq d$), trying to convince the verifier that it knows the upk of ID_i.

   (b) I can still issue some extract queries as well as transcript queries or identification queries as in Phase 1.

   (c) I wins the game if it is successful in convincing the verifier.

Definition 4.1. *The advantage of I denoted* $\mathbf{Adv}_{FIBI,I}^{imp\text{-}pa/aa/ca}$ *is the probability that I runs in time t_I and outputs a valid response in the environment set up by $\mathcal{M}$ such that:*

$$\mathbf{Adv}_{FIBI,I}^{imp\text{-}pa/aa/ca} = \Pr[\mathbf{Verify}_{\mathcal{M}} = 1]$$

We say that an FIBI scheme is secure if $\mathbf{Adv}_{FIBI,I}^{imp\text{-}pa/aa/ca}$ *is negligible.*

4.3.2 Security Analysis

Using Technique 2, we present the new security proof for Tan et al.'s FIBI as follows.

Theorem 4.6. *Tan et al.'s FIBI scheme is secure against impersonation under active and concurrent attacks if the one-more discrete logarithm (OMDL) problem is hard.*

Proof. Assume there exists an impersonator I who can break the Schnorr FIBI scheme, then there exists an algorithm $\mathcal{M}$ which solves the one-more discrete logarithm (OMDL) problem with the advantage ε_{OMDL}. $\mathcal{M}$ will be given a cyclic group $\mathbb{G}$ and the parameters $q, p, g \in \mathbb{G}$. We show how to build the algorithm $\mathcal{M}$ with the help of I such that:

$$\varepsilon_{FIBI} \leq \varepsilon_{OMDL}$$
$$t_{FIBI} = d(t_{OMDL} + 3t_{add} + 2t_{mul} + t_{inv})$$

where t_{add}, t_{mul} and t_{inv} is the time needed to compute in $\mathbb{G}$ an addition, multiplication and inverse respectively while d is the threshold value.

Setup. $\mathcal{M}$ generates $mpk = (p, q, g, v, H)$ and $msk = s$ as well as choosing an appropriate threshold d. M_{IBI} sends mpk to I.

Hash Query. If the record (i_j, X_j) exists, $\mathcal{M}$ replies using this existing record. Else, $\mathcal{M}$ asks for $X_j = g^{x_j}$ from $\mathcal{CHALL}$ and add X_j to the list before returning $\alpha_{i,j} = H(i_j, X_j, v)$ as the hash value of $i_j \in ID_j$.

Extract Query. When I queries $\mathcal{M}$ for an identity ID_j, if ID_j is not queried before, $\mathcal{M}$ issues a hash query for ID_j followed by issuing X_j to $I\mathcal{NV}_{\mathcal{DL}}$ asking for x_j. Using $\alpha_{i,j}$ and x_j, $\mathcal{M}$ computes the user private key $(\alpha_{i,j}, Y_{i,j})$ for I using the original algorithm.

Identification Query. We assume without loss of generality that I will not issue an identification query on an identity that it has already issued an extract query on. Starting with $m = 1$, $\mathcal{M}$ simulates a cheating prover for I for a round of interaction on the identity ID_j as follows:

1. $\mathcal{M}$ issues a hash query on ID_j to add (i_j, X_j) into hashing list besides asking for $n = |ID_j|$ random value $R_{i,j,m} = g^{r_{i,j,m}}$ from $\mathcal{CHALL}$. The produced commitment CMT is $(X_{i,j}, R_{i,j,m})$.
2. I selects $c_{j,m} \xleftarrow{\$} \{1,0\}^{l(k)}$ and sends $c_{j,m}$ to $\mathcal{M}$.
3. $\mathcal{M}$ queries $I\mathcal{NV}_{\mathcal{DL}}$ with $R_{i,j,m}(X_{i,j}/v^{\alpha_j})^{c_{j,m}}$ and gets the value $y_{i,j,m}$. $\mathcal{M}$ returns $y_{i,j,m}$ to I. Since $y_{i,j,m} = I\mathcal{NV}_{\mathcal{DL}}(R_{i,j,m}) + c_{j,m}Y_{i,j}$, this is exactly the response that the clone $\mathcal{M}$ would return to I.
4. $\mathcal{M}$ increments $\mathcal{M}$ by 1.

Phase 2

Impersonation. I can still issue some extract and identification queries as in Phase 1. I now plays the role of the cheating prover trying to convince $\mathcal{M}$ that it knows the user private key of the identity ID^* such that $|ID_j \cap ID^*| \geq d$ which I has not yet issued an extract query on.

Application of easy ID. Recall that for active & concurrent attacks, only the case of ID^* which is previously queried in hash query and identification queries is considered. In this case, I set $ID^* = ID_j$ and reuse the α_i^* and randomly pick a commitment values $(X^*, \{R_i^*\})$ from hash query and identification queries, respectively. Note that if I computes its own $(X^*, \{R_i^*\})$, its advantage in breaking the Schnorr IBI will be lesser, i.e., performing a less powerful attack. Thus, we only need to consider the most powerful attack carried out by I, which is the attack on easy $ID^* = ID_j$ with reused $(X^*, \{R_i^*\})$.

$\mathcal{M}$ is able to obtain a valid transcript $((X^*, \{R_i^*\}), c^*, \{y_i^*\})$ by interacting with I as prover where the pair $(\{R_i^*\}, c^*)$ does not appear before in identification query. So, $\mathcal{M}$ can extract $Y_i^* = (y_i^* - y_{i,0})/(c^* - c_0) \bmod q$ where $(c_0, y_{i,0})$ are from the previous transcripts $((X^*, \{R_i^*\}), c_0, y_{i,0})$ of ID^*. $\mathcal{M}$ solves the discrete logarithm problem of $X^* = g^{x^*}$ such that $x^* = \sum_{i=0}^{d-1} (\frac{y_i^* - y_{i,0}}{c^* - c_0} - s\alpha_i^*) \triangle_{i,S}(0) \bmod q$ where $S \in \{ID^* \cap ID_j\}$ and $|S| = d$. Discrete logarithm of R_i^* can be solved by calculating either $r_i^* = y_i^* - c^* Y_i^*$ or $r_i^* = y_{i,0} - c_0 Y_i^*$.

$\mathcal{M}$ continues to solve the discrete logarithms of the rest of the random values $R_{i,m}^* = g^{r_{i,m}^*}$ by calculating $R_{i,m}^* = y_{i,m}^* - c_{i,m}^* Y_i^*$. For other identities $|ID_j \cap ID^*| \leq d$ which has been issued an identification query on but not yet the extract query, $\mathcal{M}$ simply queries g^{x_j} to $I\mathcal{NV}_{\mathcal{DL}}$ for x_j. $\mathcal{M}$ then computes $Y_{i,j}$ and find the discrete logarithms $r_{i,j,m} = y_{i,j,m} - c_{j,m} Y_{i,j}$.

For each identity $|ID_j \cap ID^*| \leq d$, $\mathcal{M}$ computes the discrete logarithm problems of nm values $R_{i,j,m}$ and a value g^{x_j} by making $n(m+1)$ queries to the $I\mathcal{NV}_{\mathcal{DL}}$ oracle where $n = |ID|$. For the identity ID^*, $\mathcal{M}$ computes $n(m+1)$ discrete logarithms by making only $n \cdot m$ queries to the $I\mathcal{NV}_{\mathcal{DL}}$ oracle. Hence, $\mathcal{M}$ saves one $I\mathcal{NV}_{\mathcal{DL}}$ query and wins in the one-more discrete logarithm prob-

lem. After I outputs the transcript in Phase 2, $\mathcal{M}$ performed three additions, two multiplications and one inverse operations to solve the discrete logarithm of X_i. Thus, $\mathcal{M}$ has extra operation time of $d(3t_{add} + 2t_{mul} + t_{inv})$ compared to that of I. The probability analysis of $\mathcal{M}$ wins the game is the same as in Section 4.2.2.4, thus this completes the proof. □

4.4 Conclusion

In this chapter, we presented two techniques which can provide tighter security reduction for Schnorr IBI. The positive effects of these results are that Schnorr IBI is now proven to be secure against impersonation under passive, active and concurrent attacks without security loss. Besides, we showed that this benefit is inherited by the Schnorr FIBI and thus resulted a practical solution for real-world problem. These two techniques can be similarly applied on other IBI schemes in the literature.

CHAPTER 5

THE IMPLEMENTATION OF FIBI SCHEME

In this chapter, we provide a proof of concept by implementing the only FIBI (Tan et al., 2009) scheme in the literature. We report the first realisation of FIBI scheme by means of fingerprint biometrics using minutia representation where our technique integrates the security features of biometric and cryptography effectively. As fingerprint databases are confidential, we use the databases DB1 and DB2 from Fingerprint Verification Competition 2002 (FVC2002) whose images are erroneous instead of a neat fingerprint database used in practice. The simulation shows that the identification protocol at the highest security level (3072-bit) can be completed within 2.1 seconds where the best recorded matching scores are FAR=0%, FRR=28%, $d = 27$, vote=0.75 and FAR=0%, FRR=20%, $d = 42$, vote=0.75 for DB1 and DB2, respectively. Our integration technique may also be similarly applied on other fuzzy identity-based cryptosystems.

5.1 Introduction

Although we learned from Chapter 3 that FIBI is merely IBI+FE or IBI+LP, the realization of an FIBI on the other hand is not as trivial since it requires:

1. biometric trait to be represented in a fixed size discrete array;
2. each trait element is in the discrete form; and
3. biometric and cryptography must share the same threshold (or matching score).

The first and second requirements are due to the use of polynomial in binding the public biometric identity to user private key where the polynomial degree d is used as the threshold to verify the genuineness of biometrics identity. The last requirement is caused by the way the matching score works because AND operation is the only feasible method which allows one to calculate the matching score in the form of integer. We can view a real number as string but it is unlikely to have a biometrics feature extraction algorithm to output two real numbers which are exactly the same but in different executions. These requirements originated from the fact that FIBI tolerates the errors using polynomial interpolation. Unfortunately, most biometric modalities are represented in a set of continuous array such as real numbers.

In this chapter, we present the realisation of IBI by using fingerprint biometrics. However, the number of minutiae extracted from a fingerprint image is indefinite depends on the image quality and the minutiae are characterised neither in integer nor binary form. Therefore, we modify the template protection technique from Jin et al. (2010) that transforms the fingerprint minutiae into a fixed length bit string using minutiae pair representation. Subsequently, we calculate a matching score using majority voting and bit-wise AND opera-

tion. We show that our technique integrates the security features of both biometric and cryptography effectively such that biometrics provides physical security for cryptography while cryptography provides provable security for biometrics. More precisely, the extracted binary strings from fingerprint images fit well into FIBI in generating the random $(d-1)$-degree polynomial for user private key as well as reconstructing the correct cryptography values at the end of identity verification process.

5.2 Overview on Tan et al.'s FIBI

Before going into the implementation details of FIBI, we first review Tan et al.'s FIBI scheme from the implementation point of view and define a few important symbols used in the scheme:

Table 5.1: Symbols in FIBI Scheme

Symbol	Descriptions
$ID \in \mathbb{Z}^n$	enroll biometric trait of length n
$ID' \in \mathbb{Z}^n$	query biometric trait of length n
$S(b_e, b_q)$	matching score of ID and ID' where b_e and b_q are enrolled bit-string and query bit-string
d	polynomial with the input x
$q(x)$	Extract(ID)
$H(i, X, v)$	hashing algorithm with the input i, X and v
tk_{ID}	helper data of a biometrics
$\triangle_{i,U}(x)$	Lagrange coefficient with the input x

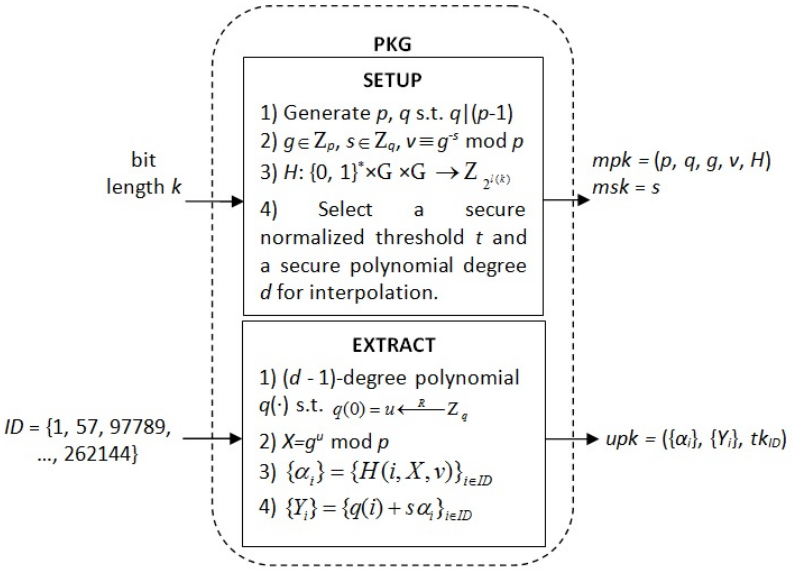


Figure 5.1: Setup and Extract Algorithms Performed by PKG

The FIBI scheme requires a Private Key Generator (PKG) which runs the Setup algorithm as follows (Figure 5.1):

1. On the input of a security parameter k, choose a large random prime $p > 2^k$ such that the discrete logarithm problem in the finite field $\mathbb{Z}_p$ is hard.

2. Choose a large prime $q \geq 2^{160}$ such that $q|(p-1)$.

3. Choose a random generator $g \in \mathbb{Z}_p$ and a random value s in $\mathbb{Z}_q$ to compute $v = g^{-s} \bmod q$.

4. Select a secure security parameter d.

5. Choose a collision resistant hash function H (for instance SHA-1, SHA-256, SHA-512 etc.) which will take as input a string and two elements in the group generated by the generator g. The master public key, $mpk = (p, q, g, v, H)$ will be made public while the master secret key, $msk = s$ will be kept secret to PKG only.

When a user enrols with the public biometric identity ID to generate the user private key upk, PKG will run the Extract algorithm as follows (Figure 5.1):

1. Choose a random value $u \in \mathbb{Z}_q$ and random coefficients $a_i \in \mathbb{Z}_q$ for $1 \leq i \leq d-1$ to construct a $(d-1)$-degree polynomial $q(x) = u + a_1 x^1 + \ldots + a_{d-1} x^{d-1} \bmod q$.
2. Compute $X = g^u$ and calculate the hash value $\alpha_i = H(i, X, v) \bmod q$ for every $i \in ID$.
3. Compute $Y_i = q(i) + s\alpha_i \bmod q$ for every $i \in ID$.
4. PKG returns $upk = (\{\alpha_i\}, \{Y_i\}, tk_{ID})$ to the user.

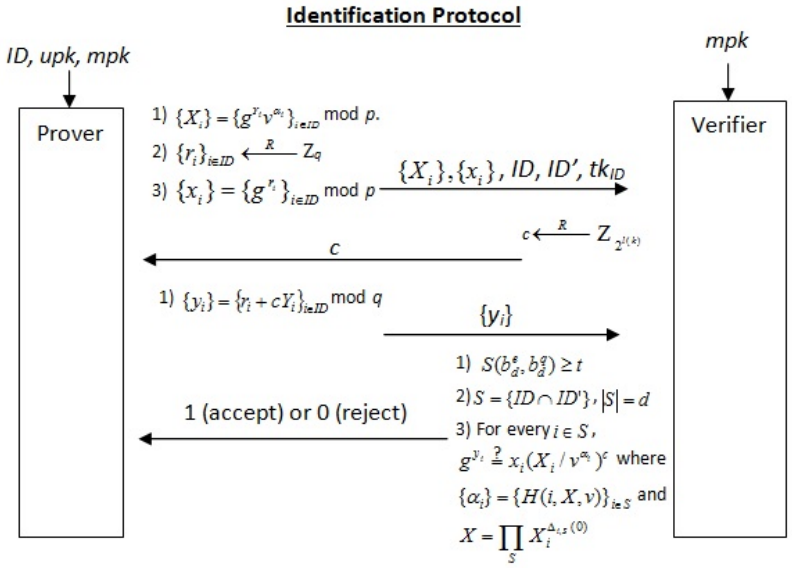


Figure 5.2: Identification Protocol of Prover and Verifier

During the identification process as shown in Figure 5.2, the user (prover) first sends a commitment to the verifier to initiate the protocol. In return, the verifier sends the challenge to user and based on the challenge, the user generates a response for verifier. At the end, based on the user's response, the verifier will output reject or accept:

1. For every $i \in ID$, user chooses random values $r_i \in \mathbb{Z}_q$, computes $x_i = g^{r_i} \bmod p$ and $X_i = g^{Y_i} v^{\alpha_i} = g^{q(i)} \bmod p$. User then sends the commitment values $(\{X_i\}, \{x_i\}, ID, ID', tk_{ID})$ to verifier.

2. In return, verifier chooses a random $c \in \mathbb{Z}_{2^{l(k)}}$ as the challenge and sends c to user.

3. As a response to the challenge, user calculates $\{y_i\} = \{r_i + cY_i\}_{i \in ID} \bmod q$ and sends $\{y_i\}$ to verifier.

4. Once verifier confirms that $S(b_e, b_q) \geq d$, a set $U \subseteq ID$ with d elements is then randomly selected and outputs 1 (accept) if $g^{y_i} = x_i (X_i / v^{\alpha_i})^c$ for every $i \in U$, where $\{\alpha_i\} = \{H(i, X, v)\}_{i \in S}$ and $X = g^u = \prod_S X_i^{\triangle_{i,U}(0)}$. Otherwise, outputs 0 (reject).

The polynomial $q(\cdot)$ in Extract algorithm is a Lagrange polynomial that binds every $i \in ID$ to the secret value u. It prevents FIBI from the collusion attack where more than one legitimate user cannot collude together to generate a more privileged upk which none of them alone could. In particular, the polynomial ensures at least d out of $n = |ID|$ elements in the upk corresponding to the biometric trait ID are valid such that the value X can be recovered by computing $X = g^u$ where $u = \sum_{\eta=0}^{d-1} q(i_\eta) \triangle_{i_\eta,U}(0), U \in ID$ and $\triangle_{i,U}(x) = \prod_{j \in U, j \neq i} \frac{x-j}{i-j}$.

In the Identification Protocol, the user's secret information $\{Y_i\}$ acts as the password that proves to the verifier that the person (or to be exact, smart card) that initiates the protocol is indeed who he/it claims to be. Anyway, there is a significant diverse where the values $\{Y_i\}$ are not revealed throughout the

identification protocol or otherwise some eavesdroppers or the verifier itself can impersonate the user. The user proves that he knows the values $\{Y_i\}$ by computing the values $\{y_i\}$ as the response to verifier's challenge. This type of protocol is called the honest verifier zero-knowledge (HVZK) protocol (Goldreich, 2009). We also note that this HVZK protocol is different from the symmetric key cryptosystems' challenge-and-response protocol as in Xi et al. (2011) which requires the user and verifier to reach consensus on a symmetric key prior to the execution of protocol.

5.2.1 A Toy Example

We now present a toy example for FIBI. Consider the scenario where a credit card company would like to adopt FIBI as their customers' identity verification mechanism. The security administrator Bob will instruct the Private Key Generator (PKG) to run the Setup algorithm of FIBI for defining the security parameters mpk and msk as well as the secure threshold d.

To register a user Alice to the system, PKG runs the Extract algorithm which takes fingerprint images of Alice as the input. At the end of biometric feature extraction, a bit string, b is generated and the indexes i of bit 1 of ID are recorded. Then the $(d-1)$-degree random polynomial $q(i)$ is constructed and bound to the master secret key $msk = s$ along with the corresponding α_i (See step 3 of Extract algorithm in Section 5.2).

For the purpose of key revocation, the PKG may concatenate the credit card expiry date to Alice's ID during the key extraction process, such as $\alpha_i = H(i||expire\_date, X, v)$. PKG will return to Alice her public key ID and $upk = (\{\alpha_i\}, \{Y_i\}, tk_{ID})$ which are stored in her cryptography-enabled credit card. Since biometric trait is used as the public key and no further documentation is required, we can see that the credit card initialization process can be completed within minutes under a trained operator.

After receiving her credit card, Alice plans to make some purchasing and she is verifying her identity on a credit card verification device, V which comprised of a fingerprint scanner and a credit card reader. Alice will give V her fingerprint reading ID' while scanning her credit card which contains her public key ID and upk. V will calculate Alice's biometric matching score $S(b_e, b_q)$. V checks firstly, $S(b_e, b_q) \geq d$, if this condition is not met, V rejects Alice or otherwise continues to verify the validity of Alice's upk through the Identification Protocol and outputs reject or accept. The details of V outputs accept are as depicted in Table 5.2.

Note that throughout the identity verification process, V does not need to communicate with database or certificate authority in order to verify Alice's identity. Moreover, due to the zero knowledge property, at the end of protocol, V learns nothing on the upk of Alice except the fact that she is the owner for the credit card which is valid in the system. These advantages cannot be achieved alone either by using IBI or biometric authentication system.

Table 5.2: Toy Example of FIBI

Algorithm	Parameter	Value
Setup	q	557
	q bit length	10
	p	1102861
	$k = p$ bit length	21
	g	273948
	s	506
	v	660497
	H	SHA-1
	d	3
Extract	ID	{8,15,23,28,33}
	ID bit string	0000000010000001000000100001000010
	u	116
	X	669450
	$q(\cdot)$	116+520x+3x^2
	α_i	{48,288,21,469,320}
	Y_i	{349,30,338,350,324}
Identification Protocol	X_i	{953382,177830,1032349,354429,824705}
	r_i	{8,14,435,106,63}
	x_i	{633433,828074,735186,404711,994240}
	C	372
	y_i	{55,34,289,525,279}
	ID'	{2,8,14,23,28}
	ID' bit string	0010000010000010000000100001000000
	$S(b_e, b_q)$	3
	U	{8,23,28}
	$\triangle_{i,U}(0)$	{17,49,492}

5.3 Biometric Identity Extraction Method

In this section, we demonstrate the biometric identity ID extraction method which at the same time protecting the user privacy[1]. Recall that this is the main challenge in implementing FIBI: how to extract a fix-length biometric trait in discrete from which supports the AND-matching score.

[1]Note that FIBI works just fine even without considering the privacy issue of biometrics. We take into account the privacy issue of biometrics only for the completeness of security.

5.3.1 Overview

Some well-known binary-string-based fingerprint template protection methods (Kevenaar et al., 2005, Tuyls et al., 2005, Farooq et al., 2007, Chen, Veldhuis, Kevenaar and Akkermans, 2009, Lee and Kim, 2010, Ahmad et al., 2011, Lim et al., 2012) were proposed in literature. The works by Kevenaar et al. (2005), Tuyls et al. (2005), Chen, Veldhuis, Kevenaar and Akkermans (2009), Lim et al. (2012) focused on the biometrics discretisation schemes in template protection methods. Inspired by Chen, Veldhuis, Kevenaar and Akkermans (2009) which proposed a multi-bit discretisation scheme using BRGC code, Lim et al. (2012) improved the single-bit discretisation schemes (Kevenaar et al., 2005, Tuyls et al., 2005) into a dynamic bit allocation discretisation scheme. The improved discretisation scheme also can produce for each user a longer binary string with higher amount of entropy. Though these discretisation schemes can produce binary strings of fixed length, they do not fulfil the last requirement of FIBI as both bit 0 and bit 1 in the binary strings are significant. For an instance, Alice enrols her biometric ID and obtains the binary string $Bits_{ID} = 011100$; queries her biometric ID' again during identification protocol and obtains another binary string $Bit_{ID'} = 010100$. Let the index of bit 1 be the public identity for FIBI, Alice's public identity are $ID = \{2,3,4\}$ and $ID' = \{2,4\}$ for the enrolled and queried biometrics respectively. Assume that the biometric threshold to authenticate Alice is $|Bit_{ID} \cap Bit_{ID'}| \geq 4$. We can see that $|Bit_{ID} \cap Bit_{ID'}| = 5 \geq 4$ and Alice is authenticated as a genuine user. However, when the verifier executes Step 4 of identification protocol,

$|ID \cap ID'| = 2 < 4$ and there is not enough values to interpolate the polynomial $q(\cdot)$ to get back the $msk = q(0) = s$. Therefore, the discretisation schemes mentioned above are not suitable for the use of FIBI.

On the other hand, Farooq et al. (2007) proposed a method to transform minutiae triplet invariant features into 2^{24} bits binary string. Although the proposed method does not require a pre-selected reference minutia point and thus free from fingerprint alignment issue, one would have to test all possible combinations of minutiae triplet when matching a query fingerprint image. Lee and Kim (2010) commented on the high complexity of matching the minutiae triplets and proposed to store all possibly generates binary strings in database instead of freshly generate all during authentication. In order to reduce the length of binary string, Lee and Kim utilised a 3D array to capture the information of all minutiae points and manage to result shorter binary string which has the maximum length of height (in pixels) $\times$ width (in pixels) $\times$ 6 (in radians) bits. If a 300×300 pixels images is used, the maximum length of the binary string is approximately 2^{20} bits, which is still 4 bits shorter than 2^{24} bits.

Jin et al. (2010) revisited the minutia triplet method and proposed a fingerprint template protection method using minutiae pair representation instead of minutiae triplet. In particular, given a set of minutiae points, $m_i = \{x_i, y_i, \theta_i\}$, where x_i, y_i and $\theta_i \in [0, 360]$ represent the coordinate and the orientation angle of the i-th minutiae, a set of the minutiae pairs is then derived from m_i and invariant features are extracted from the derived minutiae pairs. The invariant

features are further processed through minutiae pair quantization and histogram binning, hashing, binarisation and permutation to produce a bit string. By incorporating the majority voting training process, a binary string which is having a fixed length of 2^{14} bits is generated. A pictorial illustration of the fingerprint minutia to bit string transformation is showed in Figure 5.3.

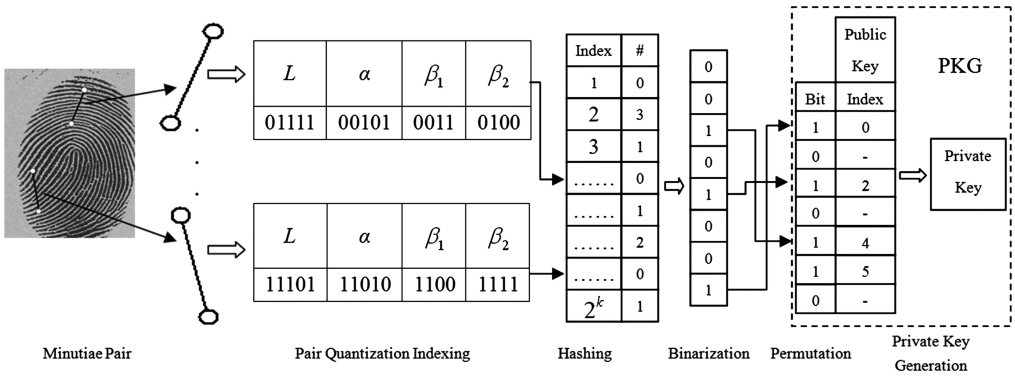


Figure 5.3: Transforming Minutiae Representation Into Bit String

Ahmad et al. (2011) later on presented a pair polar coordinate-based template protection method which has the lowest complexity and storage size but unfortunately, the generated template is in real numbers and cannot be adopted in FIBI scheme. Notice that only bit 1 in the binary string represents the existence of minutiae points and this overcomes the problem of applying discretisation schemes in FIBI scheme.

The minutaie pair template protection method (Jin et al., 2010) appeared to be the most suitable candidate and we now show how to adopt to generate a public *ID* for FIBI scheme. We also explain the possible way of achieving lower False Rejection Rate by using the concept of Bio-IBE, i.e., sacrificing the coverage of provable security on biometrics.

5.3.2 Feature Extraction from Minutiae Pairs

A single minutiae point normally suffers from the elastic deformation from fingerprint to fingerprint but the change of a minutiae pair formed by two minutiae points is not evident under rigid transformation. Besides that, minutiae pairing provide a certain degree of immunity against noise due to the use of redundant combinations of two minutiae points. The four invariant features we used are as follows:

1. The distance L between the two minutiae, where L is measured in pixel.
2. The angle α between the orientation of the two minutiae (angular difference between O_1 and O_2), the range of the angle α is $(0, 2\pi]$, and O_1 and O_2 represent the orientation of minutiae m_1 and m_2, respectively.
3. The angles β_1 and β_2 between the orientation of each minutia and the segment connecting them - the range of β_1 and β_2 is $(0, \pi]$.

β_1, β_2, and α are three distinct invariant measurements, where β_1 and β_1 are the angles between a straight line connects to two minutiae, and another straight line along the minutiae orientation; while α is the angular difference between two minutiae orientations in the range of 0 to 360. Since orientation records the direction of local fingerprint ridge, β_1, β_2, and α belong to different domains that are not correlative with each other. Figure 5.4 shows the invariant features extracted from a minutiae pair m_1 and m_2.

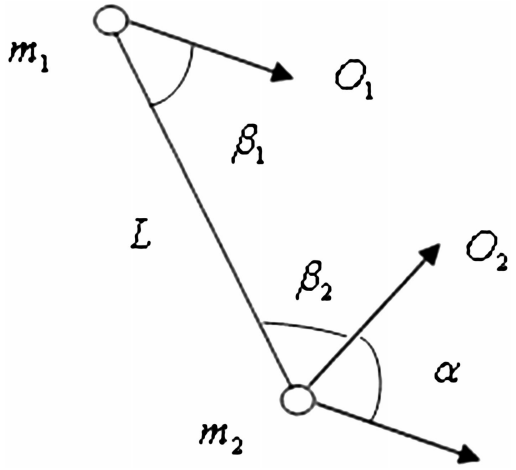


Figure 5.4: Invariant Features Extraction From Minutiae Pair

5.3.3 Minutiae Pair Quantisation

The invariant features are quantised to alleviate the distortion that occurred during the image capturing process. Let the maximum distance, L, between two minutiae points as l pixels, L is quantised into q segments with each segment containing l/q pixels for each quantization step and a total of $\log_2(l/q)$ bits are required to represent all q segments.

Let that the maximum angle between the orientations of two minutiae be 2π and the quantisation step be p, $\lfloor 2\pi/p \rfloor$ bits are required to represent the angle α between the orientations of the two minutiae. The same procedure applies to the remaining features β_1 and β_2.

After determining the number of bits needed to represent each feature, we are ready to quantise the feature into binary string. The feature value is quantised on the basis of segment index that it falls in where the segments are

each labelled by a binary decimal code. If the length L is represented by l bits, angle α by a bits, β_1 by b_1 bits and β_2 by b_2 bits, then every minutiae pair can be represented by a bit string with length $l_{mp} = l + a + b_1 + b_2$ bits. The binary string is then converted to its corresponding integer, for instance, 01001 11100 1110 0110 to 81126.

The same procedure is repeated to all minutiae pairs found in a fingerprint image where $s = {}^{n}C_2 = \frac{n(n-1)}{2}$ possible minutiae pairs combinations can be generated from a fingerprint image and n is the number of minutiae in an image.

5.3.4 Histogram Binning and Binary String Generation

Because there are $2^{l_{mp}}$ possible combinations of bits for each minutiae pair, a histogram m_i is formed to count the number of minutiae pairs that fall into each of the disjoint bins in the histogram as follows:

$$s = \sum_{i=1}^{2^{l_{mp}}} m_i,$$

where s is the total number of minutiae pairs for all $2^{l_{mp}}$ of bins.

Next, we binarise the histogram m_i by retaining the count of value 1 while setting the rest of the count values to 0. This is to ensure that the fingerprint image can be represented by a set of unique minutiae pairs, that is, occur

only once in the fingerprint image. The binarisation rule is given as follow:

$$\forall i \in [0, 2^{l_{mp}}), b_i = \begin{cases} 0, & \text{if } m_i \neq 1, \\ 1, & \text{otherwise,} \end{cases}$$

5.3.5 Public Biometric Identity Generation

We use majority voting training process to obtain a binary template for a user where the vote value is set to 0.75. In the training process, seven out of eight images are selected as the training samples and the remaining one is used for testing. Besides, to avoid statistical biasses, cross-validation by examining ${}^{8}C_{7} = 8$ combinations is performed to determine the average of false reject rate (FRR) when false acceptance rate (FAR) equals to zero (FAR=0%). The value in every position in the user template is based on the majority count of the training data. Figure 5.5 shows the majority voting scheme for generating the user template. The majority voting scheme also can be described in mathematical form as follows:

$$b_d = \{b_{d_i} | i = 1, 2, 3, \ldots, m\},$$

where $b_{d_i} = majority(b_{1,i}, b_{2,i}, \ldots, b_{k,i})$; b_{d_i} is the trained binary template of most occurrence of bit 1s from k training binary templates.

However, it is undesirable to transmit plain biometric template because of privacy concern. Therefore, a transformed version of the binary vectors is used as the user template. The said transformation is the permutation that is based on a user-specific token (tk), which is uniquely assigned to each individ-

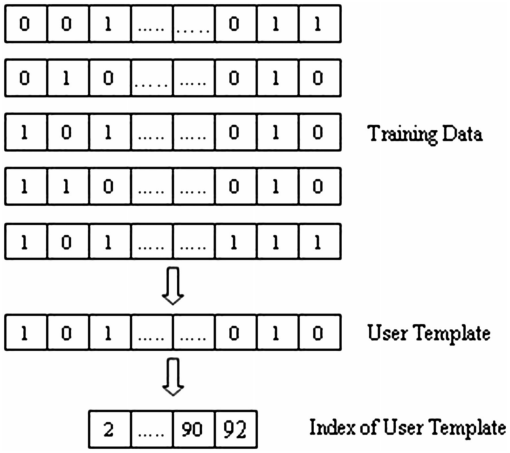


Figure 5.5: Generating User Template Through Majority Voting

ual. The user-specific token guarantees that the fingerprint presented for verification is permuted in the same manner as the one enrolled for the same users and in a different manner for the different users.

We then store the indexes of bit 1 of the trained binary template in an array as the public biometric identity *ID* for FIBI.

5.3.6 Experimental Results

The fingerprint verification competition databases FVC2002 (DB1, DB2) are used to evaluate the proposed method as the source of a neat fingerprint database is scarce. Both DB1 and DB2 contain images of 100 fingers where each finger has eight sample images. Seven out of eight images are selected as training samples, and the remaining image is used for testing, which results eight possible combinations for training samples; that is, ${}^{8}C_{7} = 8$ and the average of FRR is recorded when FAR equals to zero.

Three performance measurements are used to evaluate the proposed technique, namely, FRR, FAR, and equal error rate (EER). EER indicates the rate at which both accept and reject rates are equal. EER provides a quick way to compare the accuracy between different biometrics systems. In general, the lower the EER is, the more accurate the system is considered to be. When the threshold d increases, it causes FAR to decrease while FRR to increase.

5.3.6.1 Matching Score: AND-operation

Assume that b_e represents an enrolled bit string and b_q represents a query bit string; the matching score $S(b_e, b_q)$ can be calculated as follows:

$$S(b_e, b_q) = \sum_{i=1}^{n} (b_{e_i} \bullet b_{q_i})$$

where $\bullet$ represents a bit-wise AND operator. $\sum_{i=1}^{n}(b_e \bullet b_q)$ counts the positions in the bit string that have a bit 1 in both enrolled and query templates before summing them up.

Tables 5.3 and 5.4 depict the FRRs against the threshold, d of each combinations for FVC2002 DB1 and DB2. Although cryptographic protocol like FIBI requires zero risk of intrusion, it might be less user-convenient subsequently. Hence, d should be carefully selected such that FARs is pushed to the lowest possible while FRR can be within a certain degree of inconvenient tolerance[2]. We re-run the experiments by fixing the threshold to the average

[2]It is noted that a higher FRR implies worse user opportuneness and otherwise. In other words, the threshold d is directly proportional to FRR, which is inversely proportional to user opportuneness.

value $d = 27$ and $d = 42$ for DB1 and DB2 respectively to identify the optimum FARs and FRRs for FIBI as shown in Tables 5.5 and 5.6.

The results showed that FIBI with DB1 will have FRR=51.50% and FAR=3.99% at average while DB2 will have FRR=64.04% and FAR=3.86%. The average FRR is higher than that in cross-validation experiment because of the large ratio[3] of minimum d and maximum d which is approximately 1:10. Although the error rates are quite high, they will be much more lower in practice[4] as the experiments were done using the competition database FVC2002 whose fingerprint images are in bad conditions such as partly scanned, blurred, darkened and misaligned as shown in Figure 5.3.6.1.

Furthermore, as FIBI requires the biometric trait to be in the form of binary string, the discretisation or binarisation process of the biometric extraction algorithm will lost some information from the images. For instance, the decimal numbers will be neglected and resulted in higher FRR and FAR. Lastly, the FIBI threshold (matching score) which must be in the integer form also contributed to the high error rate. We will discuss the impact of normalising the integer matching score in the next section, i.e., using threshold in the form of real numbers.

[3]The relations of FRR,FAR and matching scores are closely related to biometric extraction algorithms which are out of the scope of this thesis.

[4]In the real world, the operator can always request a user to reproduce his fingerprint images during enrolment to generate a good quality fingerprint image.

Table 5.3: Cross-validation Performance of FVC2002 DB1 when FAR=0%

Training images (th)	Testing images (th)	Average FRR (%)	Average d	Min d	Max d
1,2,3,4,5,6,7	8	28	24.62	4	54
1,2,3,4,5,6,8	7	41	26.63	4	69
1,2,3,4,5,7,8	6	44	26.78	5	68
1,2,3,4,6,7,8	5	64	29.21	6	63
1,2,3,5,6,7,8	4	53	26.84	8	57
1,2,4,5,6,7,8	3	31	25.28	6	52
1,3,4,5,6,7,8	2	25	25.60	4	52
2,3,4,5,6,7,8	1	36	25.28	6	58
Average	-	40.25	26.28	≈ 5	≈ 59

Table 5.4: Cross-validation Performance of FVC2002 DB2 when FAR=0%

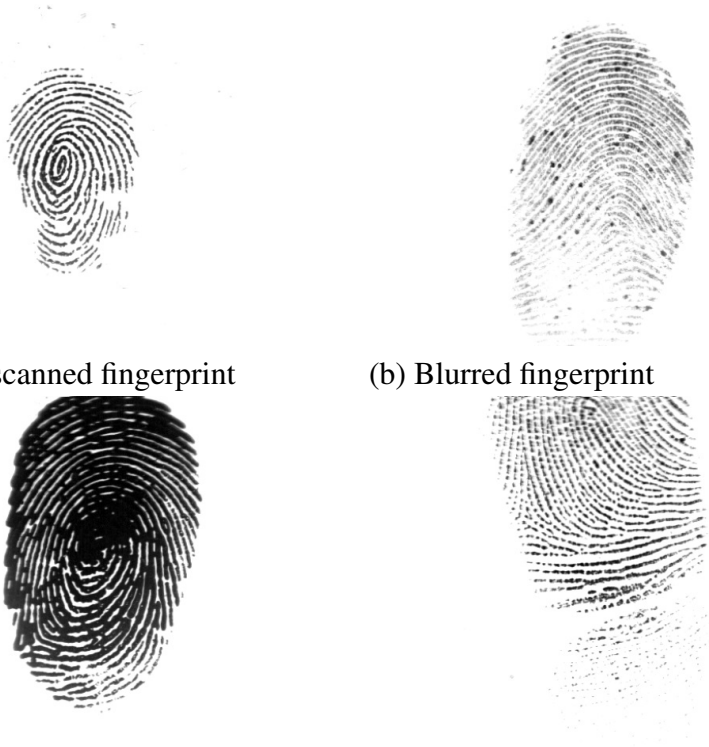
Training images (th)	Testing images (th)	Average FRR (%)	Average d	Min d	Max d
1,2,3,4,5,6,7	8	37	41.01	10	101
1,2,3,4,5,6,8	7	31	40.09	12	87
1,2,3,4,5,7,8	6	44	43.95	9	94
1,2,3,4,6,7,8	5	56	43.77	10	114
1,2,3,5,6,7,8	4	58	43.80	10	116
1,2,4,5,6,7,8	3	31	40.62	8	95
1,3,4,5,6,7,8	2	20	41.43	8	98
2,3,4,5,6,7,8	1	25	40.52	9	100
Average	-	37.75	41.90	≈ 10	≈ 101

Table 5.5: FRR and FAR for FVC2002 DB1 Using Averaged $d = 27$

Training images (th)	Testing images (th)	d	Average FRR (%)	Average FAR (%)
1,2,3,4,5,6,7	8	27	47.69	3.39
1,2,3,4,5,6,8	7	27	45.65	3.37
1,2,3,4,5,7,8	6	27	53.29	4.61
1,2,3,4,6,7,8	5	27	72.00	6.02
1,2,3,5,6,7,8	4	27	57.85	3.41
1,2,4,5,6,7,8	3	27	46.35	3.64
1,3,4,5,6,7,8	2	27	40.02	4.04
2,3,4,5,6,7,8	1	27	48.74	3.45
Average	-	27	51.45	3.99

Table 5.6: FRR and FAR for FVC2002 DB2 Using Averaged $d = 42$

Training images (th)	Testing images (th)	d	Average FRR (%)	Average FAR (%)
1,2,3,4,5,6,7	8	42	63.18	2.91
1,2,3,4,5,6,8	7	42	60.71	3.07
1,2,3,4,5,7,8	6	42	65.15	5.16
1,2,3,4,6,7,8	5	42	71.45	3.35
1,2,3,5,6,7,8	4	42	75.36	4.66
1,2,4,5,6,7,8	3	42	59.38	3.76
1,3,4,5,6,7,8	2	42	56.14	4.32
2,3,4,5,6,7,8	1	42	60.94	3.65
Average	-	42	64.04	3.86



(a) Partly scanned fingerprint (b) Blurred fingerprint

(c) Darkened fingerprint (d) Misaligned fingerprint

Figure 5.6: Bad Fingerprint Images

5.3.6.2 Matching Score: Normalised AND-operation

In order to increase the user opportuneness, we now normalize the matching score as follows to get lower FRRs when FARs=0%:

$$S(b_e, b_q) = \frac{\sum_{i=1}^{n}(b_{e_i} \bullet b_{q_i})}{\sqrt{\sum_{i=1}^{n} b_{e_i} \sum_{i=1}^{n} b_{q_i}}}$$

where the denominator denotes the total of bit 1 in both enrol and query *ID*.

The same experiments are run again using the normalized matching score and Tables 5.8, 5.10 display the results. Obviously, the FARs and FRRs of normalised matching score drop for approximately half. Besides, the ratio of minimum and maximum d are only around 1:3, as opposed to 1:10 previously.

Table 5.7: Cross-validation Performance of FVC2002 DB1 when FAR=0%

Training images (th)	Testing images (th)	Average FRR (%)	Average *d*	Min *d*	Max *d*
1,2,3,4,5,6,7	8	15	0.16	0.10	0.25
1,2,3,4,5,6,8	7	21	0.16	0.08	0.22
1,2,3,4,5,7,8	6	34	0.16	0.08	0.24
1,2,3,4,6,7,8	5	38	0.16	0.09	0.23
1,2,3,5,6,7,8	4	45	0.16	0.11	0.24
1,2,4,5,6,7,8	3	21	0.16	0.07	0.22
1,3,4,5,6,7,8	2	22	0.16	0.08	0.22
2,3,4,5,6,7,8	1	22	0.16	0.08	0.21
Average	-	27.25	0.16	0.09	0.23

Table 5.8: Cross-validation Performance of FVC2002 DB2 when FAR=0%

Training images (th)	Testing images (th)	Average FRR (%)	Average *d*	Min *d*	Max *d*
1,2,3,4,5,6,7	8	21	0.17	0.11	0.22
1,2,3,4,5,6,8	7	17	0.16	0.10	0.23
1,2,3,4,5,7,8	6	27	0.16	0.10	0.21
1,2,3,4,6,7,8	5	30	0.16	0.11	0.22
1,2,3,5,6,7,8	4	36	0.16	0.10	0.23
1,2,4,5,6,7,8	3	16	0.16	0.09	0.22
1,3,4,5,6,7,8	2	13	0.16	0.09	0.23
2,3,4,5,6,7,8	1	16	0.16	0.11	0.22
Average	-	22	0.16	0.10	0.22

Table 5.9: FRR and FAR for FVC2002 DB1 Using the Averaged $d = 0.16$

Training images (th)	Testing images (th)	d	Average FRR (%)	Average FAR (%)
1,2,3,4,5,6,7	8	0.16	16.45	1.49
1,2,3,4,5,6,8	7	0.16	19.62	1.03
1,2,3,4,5,7,8	6	0.16	38.52	1.03
1,2,3,4,6,7,8	5	0.16	51.52	1.98
1,2,3,5,6,7,8	4	0.16	37.78	0.95
1,2,4,5,6,7,8	3	0.16	24.80	1.05
1,3,4,5,6,7,8	2	0.16	14.61	1.36
2,3,4,5,6,7,8	1	0.16	24.07	1.05
Average	-	0.16	28.42	1.24

Table 5.10: FRR and FAR for FVC2002 DB2 Using the Averaged $d = 0.16$

Training images (th)	Testing images (th)	d	Average FRR (%)	Average FAR (%)
1,2,3,4,5,6,7	8	0.16	27.09	1.36
1,2,3,4,5,6,8	7	0.16	25.72	1.29
1,2,3,4,5,7,8	6	0.16	27.45	1.12
1,2,3,4,6,7,8	5	0.16	29.45	1.02
1,2,3,5,6,7,8	4	0.16	38.29	1.28
1,2,4,5,6,7,8	3	0.16	24.24	1.14
1,3,4,5,6,7,8	2	0.16	25.75	1.12
2,3,4,5,6,7,8	1	0.16	24.99	0.91
Average	-	0.16	27.87	1.16

5.3.6.3 Discussion on Unnormalised and Normalised Matching Score

At the first glance, the user opportuneness was significantly improved when the normalized matching score is used as the threshold d for FIBI. Unfortunately, d is now not in the form of neither integer nor binary, which does not fit the fundamental requirements of FIBI. Although we can work around with this issue by assigning a random value for the polynomial threshold d, doing so will result in the redundancy problems as discussed in Chapter 3.

Another alternative to preserve the user opportuneness brought by normalized matching score is to downgrade the FIBI scheme to the original IBI scheme shown in Chapter 4. Recall that an IBI scheme views the public key as a publicly verified identity, the discretised binary string (ID) works well with IBI. As long as a prover can produce a ID' which passes the normalized matching score during identification protocol, the verifier can confidently reuse the same binary string ID which is used by PKG in generating the user private key.

Reader may have realized that the alternative is purely implementation issue and the security of ID is not covered by IBI because the matching score is now independent from the IBI scheme. Besides, this shares the similar concept of biometric key extraction algorithms such as Fuzzy Extractor, Fuzzy Vault, Fuzzy Commitment, Fuzzy Sketch etc. in generating a public key for the IBI scheme. On the contrary, FIBI binds ID and the matching score with its user private key generation process and thus one can mathematically prove the se-

curity of the biometrics *ID*. The significant differences of unnormalised and normalised matching scores are shown in Table 5.11.

Table 5.11: Unnormalised and Normalised Matching Scores

			Biometric d		Cryptography d	
Matching Score	Provable Security	Cryp-tosystems	FVC2002 DB1	FVC2002 DB2	FVC2002 DB1	FVC2002 DB2
Unnoma-lised	Yes	FIBC	27	42	27	42
Nor-malised	No	IBC	0.16	0.16	N/A	N/A

As we prefer provable security, the remaining sections will be based on the unnormalised matching score and FIBI scheme.

5.4 FIBI Simulation and Computation Time

Using the public biometric identity extraction method presented in the previous section, we manage to produce a 2^{14} bit string given a user fingerprint image as well as define the threshold d for matching score using AND operation. We show in the simulation that the FIBI is efficient and the extracted public biometric identity serves the FIBI scheme perfectly.

5.4.1 Optimisations

After the first step of protocol, verifier can decide to continue or abort the Identification Protocol base on the condition $S(b_e, b_q) \geq d$. If the condition is met, verifier can now randomly select d elements from *ID* to form the set

S such that $|S| = d$ and send both the set S and the challenge c to the prover. So, the prover and the verifier can reduce the computations in step 3 and step 4 for a factor of $n-d$. Note that this optimization does not affect the security because the verifier only needs to know d out of n elements of X_i to reconstruct $X = \prod_U X_i^{i,U}(0)$ and prover only needs to prove the partial knowledge of upk corresponding to the set S, which is the partial elements of public biometric identity.

Besides, some pre-calculations can be performed in the identification protocol. Firstly, the PKG can include in upk, the value $X_i = g^{q(i)}$ for $i \in ID$ so that the user does not need to compute $\{X_i\}$ during step 1. Secondly, the verifier can compute the division value $v^{\alpha_{i\eta}}$ of step 4 by raising v to the power of $-\alpha_i$ for $1 \leq i \leq d$ to avoid computing d times multiplicative inverse such that $X_i/v^{\alpha_i} = X_i(v)^{-\alpha_i}$. Thirdly, the verifier can compute the Lagrange coefficient on the point 0, which is the value $_{i,U}(0)$ immediately after determining the set U in step 2 instead of doing so after receiving the response in step 3.

5.4.2 Results

With the use of J2SE 6 and NetBeans as the IDE, the FIBI is implemented on Intel Core i7-4702MQ 2.2GHz, 8-GB RAM with Windows 8 64-bit. ID and ID' for Extract and Identification Protocol are generated the same way as in Section 5.3.5. The identified averaged threshold value $d_{DB1} = 27$ and $d_{DB2} = 42$ for FVC2002 DB1 and DB2 respectively are used as the biometrics

threshold and the degree of Lagrange polynomial. Adopting the security specification of DSA algorithm in FIPS-184-6, we set the bit lengths for the primes (q, p) as $(160, 1024), (256, 2048), (256, 3072)$ with SHA-1 as the hashing algorithm.

The FIBI is executed for 1000 rounds where the algorithms Setup, Extract, and Identification Protocol are executed sequentially. The average timing is calculated in seconds as shown in the Table 5.12.

Table 5.12: Average Timing of 1000 Rounds of FIBI

	Time (s)					
	FVC2002 DB1			FVC2002 DB2		
Algorithm	1024	2048	3072	1024	2048	3072
Setup	0.002	52.337	154.830	0.002	55.065	186.784
Extract	0.017	0.031	0.044	0.024	0.030	0.048
Identification protocol	0.161	0.801	1.935	0.233	0.992	2.077

5.5 Security Issues

Recall that adversary of IBI can perform three types of attacks, namely, passive attack, active attack and concurrent attacks. Tan et al. (2009) proved that the FIBI is secure against impersonation under passive attack as well as active and concurrent attacks but they did not mention the potential security issues from biometrics perspective. We notice that besides taking into account the attacks of IBI, FIBI needs to further consider the false acceptance attack of

biometrics which falls under the active attack and concurrent attacks categories. We hereby define the two types of false acceptance attack:

1. Outsider attack: the impersonator is not a registered user in the system, but he manages to present two biometric identities ID and ID' to the verifier such that $S(b_e, b_q) \geq d$, where ID and ID' are the biometric identities of an existing user in the system.
2. Insider attack: the adversary is a registered user in the system, and he manages to present two biometric identities ID and ID' to the verifier such that $S(b_e, b_q) \geq d$, where ID is the enrolled biometric identity of the adversary, whereas ID' is the biometric identity of an existing user in the system who is not the impersonator himself.

The outsider attack is harmless to FIBI because the adversary is not a registered user. Thus, he does not possess a valid upk to run a successful identification protocol with the verifier. The insider attack on the other hand allows a user A who has a valid user private key to impersonate as another user B. To overcome this problem, we must set the threshold d of the biometric identity extraction method to the maximum where FAR is equals to 0% for every user as presented in Section 5.3.6. However, doing so will result in extremely high FRR and sacrifice the user convenience as shown in Tables 5.13 and 5.14.

Table 5.13: Worst FRR for FVC2002 DB1 Using Largest $d = 69$

Training images (th)	Testing images (th)	d	Average FRR (%)	Min FRR (%)	Max FRR (%)
1,2,3,4,5,6,7	8	69	96.58	95.62	100
1,2,3,4,5,6,8	7	69	97.99	97.02	100
1,2,3,4,5,7,8	6	69	98.62	97.64	100
1,2,3,4,6,7,8	5	69	99.33	98.34	100
1,2,3,5,6,7,8	4	69	98.66	97.68	100
1,2,4,5,6,7,8	3	69	97.99	97.02	100
1,3,4,5,6,7,8	2	69	98.70	97.72	100
2,3,4,5,6,7,8	1	69	97.99	97.02	100
Average	-	69	96.58	97.23	100

Table 5.14: Worst FRR for FVC2002 DB2 Using Largest $d = 116$

Training images (th)	Testing images (th)	d	Average FRR (%)	Min FRR (%)	Max FRR (%)
1,2,3,4,5,6,7	8	116	98.61	97.65	100
1,2,3,4,5,6,8	7	116	98.61	97.65	100
1,2,3,4,5,7,8	6	116	99.59	98.61	100
1,2,3,4,6,7,8	5	116	99.85	98.86	100
1,2,3,5,6,7,8	4	116	99.74	98.75	100
1,2,4,5,6,7,8	3	116	98.82	97.85	100
1,3,4,5,6,7,8	2	116	98.77	97.81	100
2,3,4,5,6,7,8	1	116	98.82	97.85	100
Average	-	116	98.38	97.23	100

5.6 Conclusion

We showed that it is feasible to implement FIBI scheme as a bio-crypto authentication mechanism. A public biometric identity in the scheme is realised by transforming the fingerprint minutiae into a fixed-length binary string. Experiment results indicate that the identification protocol of the strongest security level (3072 bits) can be completed within 2.1 seconds. By and large, other FIBCs (Sahai and Waters, 2005, Baek et al., 2007, Yang, Cao and Dong, 2011) can adopt this implementation in a similar way because they share the same private key extraction mechanism.

CHAPTER 6

CONCLUSION

We have studied the methods of providing authentication service by using the primitives of Bio-Crypto, namely, fuzzy identity-based cryptography (FIBC). The work began by introducing the development of Bio-Crypto and followed by the cryptanalysis on encrytion, signature and identification schemes in FIBC.

We have shown that a variant of fuzzy identity-based encryption (FIBE) scheme, namely, biometric identity-based encryption (Bio-IBE) schemes are having redundancy problem and is not feasible to be deployed. These problems are fixed by removing either the fuzzy extractor (FE) or the Lagrange polynomial (LP) from the schemes and new algorithm flow was proposed. Next, two fuzzy identity-based signature (FIBS) schemes was cryptanalysed and the result shows that although the construction of FIBS is merely IBS+LP or IBS+FE, extra care is needed when performing integration on the key extraction algorithm. We also pointed out that some cryptosystems by nature[1] cannot be proven secure even though the construction itself does not contain any flaw.

As there exist only one FIBI which is provably secure, extensive analysis was done on its underlying IBI scheme, namely, Schnorr IBI scheme. We discovered two new proving techniques which can tighten the security reduction

[1]The word "nature" is limited to the existing proving techniques in the literature.

of Schnorr IBI. In precise, we showed that by using security parameter of k bits, Schnorr IBI can achieve the same security level of security parameter in k^2 bits. The first technique requires minimal increment on the public key size while the second technique does not require any amendments on Schnorr IBI. We notice that these proving techniques can be generally applied on other IBI schemes.

Finally, as a proof of concept, the Schnorr FIBI is implemented based on the security parameters of only k bits, by using fingerprints as the biometric identity. Since FIBI requires the biometric identity to be in integer array of fixed length, we adopted a fingerprint template protection method which can convert the fingerprint images into binary string. The binary string is then used as the input to generate the user public key while the biometric matching score is then treated as the degree of polynomial in the user private key. Running the simulation in J2SE on Intel Core i7-4702MQ, the three-move canonical identification protocol of Schnorr FIBI can be completed within 2.1 seconds in 3072-bit security using the fingerprint competition database FVC2002 DB1 and FVC2002 DB2 respectively. As FIBI shares the similar nature in **Setup** and **Extract** algorithms with FIBE and FIBS, the same implementation technique can be applied on the latter two primitives as well.

6.1 Future Works

We would like to list out some future works which may shape the results in this thesis into the better forms:

1. rigorously define a transformation framework for IBS⇒FIBS and IBI⇒FIBI;
2. explore the possibility of proving Schnorr IBI or FIBI in the standard model;
3. generalise the Easy ID technique to encryption schemes;
4. further optimise the security and implementation of Schnorr FIBI scheme particularly towards the field of biometrics.

6.1.1 Transformation Frameworks

It is obvious that FIBS and FIBI imply IBS and IBI, respectively, where the latters are the singleton of the formers. From Section 4.3, we see that FIBI it is feasible to first construct FIBI from the underlying IBI and subsequently provide a direct proof for the resulting FIBI and the same shall goes for FIBS and IBS. However, we are not sure if there exist a general framework which can provide security implications for them or not.

6.1.2 Schnorr (F)IBI in the Standard Model

The unique hashing mechanism in Schnorr IBI and IBS is the point which inspired the idea of easy ID. Though the "secret" hashing input X is used to generate the *usk*, it can be known publicly after *usk* is generated and in fact, it can be used as a public key element since then. We believe there exists a security model which hasn't been discovered yet (Bellare et al., 2009, Kurosawa and Heng, 2004, Yang et al., 2008), that can capture such mechanism. From the initial result on hand, we think that such security model may be not as strong as the standard model, but is certainly weaker than the random oracle model.

6.1.3 Easy ID

We see the potential on easy ID technique to be adopted by the identity-based encryption schemes as well. Easy ID can be further formulated as a security model, similar to the Selective-ID model which stands as an intermediate model between the standard model and random oracle model.

6.1.4 Bio-Crypto

There are still rooms of improvement for the implementation of Schnorr FIBI scheme particularly towards the field of biometrics. A potential starting point is to explore the factors which contribute to the difference between minimum matching and maximum matching scores. We are interested in knowing

the feasibility of reducing the information loss during discretisation of biometric images. In other words, the biometric pattern of intra class should be as close as possible while the biometric pattern of inter class should be as far as possible.

The realisation of Bio-Crypto techniques such as FIBI can be used to provide some database-less yet certificate-less multi-factor authentication solutions. These solutions can ease the cost burden on verifier's end and simplify the certificate and database management issues. As authentication is a fundamental security requirement, FIBI can be employed by any electronic application that needs authentication, where by the verifier can be of any user, device or server. This can effectively solve the user genuinity problem in smart card lost incidents.

LIST OF REFERENCES

Agrawal, S., Boyen, X., Vaikuntanathan, V., Voulgaris, P. and Wee, H. (2012), 'Functional encryption for threshold functions (or fuzzy ibe) from lattices', **7293**, 280–297.

Ahmad, T., Hu, J. and Wang, S. (2011), 'Pair-polar coordinate-based cancelable fingerprint templates', *Pattern Recognition* **44**(1011), 2555 – 2564. Semi-Supervised Learning for Visual Content Analysis and Understanding.

Anada, H. and Arita, S. (2011), Identification schemes from key encapsulation mechanisms, *in* A. Nitaj and D. Pointcheval, eds, 'Progress in Cryptology AFRICACRYPT 2011', Vol. 6737 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 59–76.

Arita, S. and Kawashima, N. (2007), 'An identification scheme with tight reduction', *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **E90-A**(9), 1949–1955.

Baek, J., Susilo, W. and Zhou, J. (2007), New constructions of fuzzy identity-based encryption, *in* 'Proceedings of the 2nd ACM symposium on Information, computer and communications security', ASIACCS '07, ACM, New York, NY, USA, pp. 368–370.

Bellare, M., Boldyreva, A. and Palacio, A. (2004), An uninstantiable random-oracle-model scheme for a hybrid-encryption problem, *in* C. Cachin and J. Camenisch, eds, 'Advances in Cryptology - EUROCRYPT 2004', Vol. 3027 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 171–188.

Bellare, M., Fischlin, M., Goldwasser, S. and Micali, S. (2001), Identification protocols secure against reset attacks, *in* B. Pfitzmann, ed., 'Advances in Cryptology EUROCRYPT 2001', Vol. 2045 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 495–511.

Bellare, M., Namprempre, C. and Neven, G. (2009), 'Security proofs for identity-based identification and signature schemes', *Journal of Cryptology* **22**(1), 1–61.

Bellare, M. and Palacio, A. (2002), Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks, *in* 'CRYPTO', pp. 162–177.

Bellare, M. and Rogaway, P. (1993), Random oracles are practical: a paradigm for designing efficient protocols, *in* 'Proceedings of the 1st ACM Conference on Computer and Communications Security', CCS '93, ACM, pp. 62–73.

Bethencourt, J., Sahai, A. and Waters, B. (2007), Ciphertext-policy attribute-based encryption, *in* 'IEEE Symposium on Security and Privacy, 2007. SP '07.', pp. 321–334.

Burnett, A., Duffy, A., Dowling, T. and Maynooth, N. (2004), 'A biometric identity based signature scheme'.

Canetti, R., Goldreich, O. and Halevi, S. (2004), 'The random oracle methodology, revisited', *J. ACM* **51**(4), 557–594.

Chen, C., Veldhuis, R. N. J., Kevenaar, T. A. M. and Akkermans, A. H. M. (2009), 'Biometric quantization through detection rate optimized bit allocation', *EURASIP J. Adv. Signal Process* **2009**, 29:1–29:16.

Chen, L. and Cheng, Z. (2005), Security proof of sakai-kasaharas identity-based encryption scheme, *in* N. Smart, ed., 'Cryptography and Coding', Vol. 3796 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 442–459.

Chen, W., Zhu, L., Cao, X. and Geng, Y. (2009), A novel fuzzy identity-based signature with dynamic threshold, *in* 'Third International Conference on Network and System Security, 2009. NSS '09.', pp. 192–198.

Chin, J.-J., Heng, S.-H. and Goi, B.-M. (2008), An efficient and provable secure identity-based identification scheme in the standard model, *in* 'EuroPKI', pp. 60–73.

Diffie, W. and Hellman, M. (1976), 'New directions in cryptography', *IEEE Transactions on Information Theory,* **22**(6), 644–654.

Dodis, Y., Reyzin, L. and Smith, A. (2004), Fuzzy extractors: How to generate strong keys from biometrics and other noisy data, *in* C. Cachin and J. Camenisch, eds, 'Advances in Cryptology - EUROCRYPT 2004', Vol. 3027 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 523–540.

Fan, L., Zheng, J. and Yang, J. (2009), A biometric identity based signature scheme in the standard model, *in* 'IEEE International Conference on Network Infrastructure and Digital Content, 2009. IC-NIDC 2009.', pp. 552–556.

Farooq, F., Bolle, R., Jea, T.-Y. and Ratha, N. (2007), Anonymous and revocable fingerprint recognition, *in* 'IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07.', pp. 1–7.

Fiat, A. and Shamir, A. (1987), How to prove yourself: Practical solutions to identification and signature problems, *in* A. Odlyzko, ed., 'Advances in Cryptology CRYPTO 86', Vol. 263 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 186–194.

Fujioka, A., Saito, T. and Xagawa, K. (2012*a*), Secure hierarchical identity-based identification without random oracles, *in* D. Gollmann and F. Freiling, eds, 'Information Security', Vol. 7483 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 258–273.

Fujioka, A., Saito, T. and Xagawa, K. (2012*b*), Security enhancements by or-proof in identity-based identification, *in* F. Bao, P. Samarati and J. Zhou, eds, 'Applied Cryptography and Network Security', Vol. 7341 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 135–152.

Fujisaki, E., Nishimaki, R. and Tanaka, K. (2009), On the insecurity of the fiat-shamir signatures with iterative hash functions, *in* J. Pieprzyk and F. Zhang, eds, 'Provable Security', Vol. 5848 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 118–128.

Fujisaki, E. and Okamoto, T. (2013), 'Secure integration of asymmetric and symmetric encryption schemes', *Journal of Cryptology* **26**(1), 80–101.

Goh, E.-J., Jarecki, S., Katz, J. and Wang, N. (2007), 'Efficient signature schemes with tight reductions to the diffie-hellman problems', *Journal of Cryptology* **20**(4), 493–514.

Goldreich, O. (2009), *Foundations of Cryptography*, 1st edn, Cambirdge University Press, Inc.

Goldwasser, S. and Kalai, Y. (2003), On the (in)security of the fiat-shamir paradigm, *in* '44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.', pp. 102–113.

Goyal, V., Pandey, O., Sahai, A. and Waters, B. (2006), Attribute-based encryption for fine-grained access control of encrypted data, *in* 'Proceedings of the 13th ACM conference on Computer and communications security', CCS '06, ACM, New York, NY, USA, pp. 89–98.

Hofheinz, D. and Kiltz, E. (2012), 'Programmable hash functions and their applications', *Journal of Cryptology* **25**(3), 484–527.

Jain, A., Nandakumar, K. and Nagar, A. (2008), 'Biometric template security', *EURASIP Journal on Advances in Signal Processing* **2008**(1), 579416.

Jin, Z., Teoh, A. B. J., Ong, T. S. and Tee, C. (2010), 'A revocable fingerprint template for security and privacy preserving.', *TIIS* **4**(6), 1327–1342.

Kevenaar, T. A. M., Schrijen, G.-J., van der Veen, M., Akkermans, A. H. M. and Zuo, F. (2005), Face recognition with renewable and privacy preserving binary templates, *in* 'Fourth IEEE Workshop on Automatic Identification Advanced Technologies, 2005.', pp. 21–26.

Kurosawa, K. and Heng, S.-H. (2004), From digital signature to id-based identification/signature, *in* 'Public Key Cryptography', pp. 248–261.

Kurosawa, K. and Heng, S.-H. (2005), Identity-based identification without random oracles, *in* 'ICCSA (2)', pp. 603–613.

Kurosawa, K. and Heng, S.-H. (2006), The power of identification schemes, *in* 'Public Key Cryptography', pp. 364–377.

Lee, C. and Kim, J. (2010), 'Cancelable fingerprint templates using minutiae-based bit-strings', *Journal of Network and Computer Applications* **33**(3), 236 – 246. ¡ce:title¿Recent Advances and Future Directions in Biometrics Personal Identification¡/ce:title¿.

Lim, M.-H., Teoh, A. B. J. and Toh, K.-A. (2012), 'An efficient dynamic reliability-dependent bit allocation for biometric discretization', *Pattern Recognition* **45**(5), 1960 – 1971.

Liu, X., Miao, Q. and Li, D. (2007), A biometric identity based signature scheme with convenient verification, *in* 'Future Generation Communication and Networking (FGCN 2007)', Vol. 1, pp. 113–117.

Menezes, A. J., Vanstone, S. A. and Oorschot, P. C. V. (1996), *Handbook of Applied Cryptography*, 1st edn, CRC Press, Inc., Boca Raton, FL, USA.

Menezes, A., Okamoto, T. and Vanstone, S. (1991), Reducing elliptic curve logarithms to logarithms in a finite field, *in* 'Proceedings of the Twenty-third Annual ACM Symposium on Theory of Computing', STOC '91, ACM, New York, NY, USA, pp. 80–89.

Nielsen, J. (2002), Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case, *in* M. Yung, ed., 'Advances in Cryptology CRYPTO 2002', Vol. 2442 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 111–126.

Numayama, A., Isshiki, T. and Tanaka, K. (2008), Security of digital signature schemes in weakened random oracle models, *in* R. Cramer, ed., 'Public Key Cryptography PKC 2008', Vol. 4939 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 268–287.

Rckert, M. (2010), Adaptively secure identity-based identification from lattices without random oracles, *in* J. Garay and R. Prisco, eds, 'Security and Cryptography for Networks', Vol. 6280 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 345–362.

Ren, Y., Gu, D., Wang, S. and Zhang, X. (2010), 'New fuzzy identity-based encryption in the standard model', *Informatica* **21**(3), 393–407.

Sahai, A. and Waters, B. (2005), Fuzzy identity-based encryption, *in* R. Cramer, ed., 'Advances in Cryptology EUROCRYPT 2005', Vol. 3494 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 457–473.

Sakai, R. and Kasahara, M. (2003), 'Id based cryptosystems with pairing on elliptic curve', Cryptology ePrint Archive, Report 2003/054.

Sarier, N. (2008), A new biometric identity based encryption scheme, *in* 'Young Computer Scientists, 2008. ICYCS 2008. The 9th International Conference for', pp. 2061–2066.

Sarier, N. (2009), A new approach for biometric template storage and remote authentication, *in* M. Tistarelli and M. Nixon, eds, 'Advances in Biometrics', Vol. 5558 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 909–918.

Sarier, N. (2010*a*), Biometric identity based signature revisited, *in* F. Martinelli and B. Preneel, eds, 'Public Key Infrastructures, Services and Applications', Vol. 6391 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 271–285.

Sarier, N. (2010*b*), Generic constructions of biometric identity based encryption systems, *in* P. Samarati, M. Tunstall, J. Posegga, K. Markantonakis and D. Sauveron, eds, 'Information Security Theory and Practices. Security and Privacy of Pervasive Systems and Smart Devices', Vol. 6033 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 90–105.

Sarier, N. D. (2011), 'A new biometric identity based encryption scheme secure against dos attacks', *Security and Communication Networks* **4**(1), 23–32.

Schnorr, C. (1990), Efficient identification and signatures for smart cards, *in* J.-J. Quisquater and J. Vandewalle, eds, 'Advances in Cryptology EUROCRYPT 89', Vol. 434 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 688–689.

Shamir, A. (1985), Identity-based cryptosystems and signature schemes, *in* G. Blakley and D. Chaum, eds, 'Advances in Cryptology', Vol. 196 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 47–53.

Shi, W., Jang, I. and Yoo, H. S. (2010), 'An Improved Fuzzy Identity-Based Encryption Scheme With Constant Size Ciphertext', *International Journal of Digital Content Technology and Its Applications* **4**, 7–14.

Tan, S.-Y., Chin, J.-J., Heng, S.-H. and Goi, B.-M. (2013), 'An improved efficient provable secure identity-based identification scheme in the standard model', *KSII Transactions on Internet and Information Systems (TIIS)* **7**(4), 910–922.

Tan, S.-Y., Heng, S.-H. and Goi, B.-M. (2010), Java implementation for pairing-based cryptosystems, *in* D. Taniar, O. Gervasi, B. Murgante, E. Pardede and B. Apduhan, eds, 'Computational Science and Its Applications ICCSA 2010', Vol. 6019 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 188–198.

Tan, S.-Y., Heng, S.-H., Goi, B.-M. and Moon, S. (2009), Fuzzy identity-based identification scheme, *in* D. lzak, T.-h. Kim, J. Ma, W.-C. Fang, F. Sandnes, B.-H. Kang and B. Gu, eds, 'U- and E-Service, Science and Technology', Vol. 62 of *Communications in Computer and Information Science*, Springer Berlin Heidelberg, pp. 123–130.

Teoh, A., Goh, A. and Ngo, D. (2006), 'Random multispace quantization as an analytic mechanism for biohashing of biometric and random identity inputs', *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **28**(12), 1892–1901.

Thorncharoensri, P., Susilo, W. and Mu, Y. (2009), Identity-based identification scheme secure against concurrent-reset attacks without random oracles, *in* H. Youm and M. Yung, eds, 'Information Security Applications', Vol. 5932 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 94–108.

Tuyls, P., Akkermans, A., Kevenaar, T., Schrijen, G.-J., Bazen, A. and Veldhuis, R. (2005), Practical biometric authentication with template protection, *in* T. Kanade, A. Jain and N. Ratha, eds, 'Audio- and Video-Based Biometric Person Authentication', Vol. 3546 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 436–446.

Wang, C. (2012), 'A provable secure fuzzy identity based signature scheme', *Science China Information Sciences* **55**(9), 2139–2148.
URL: *http://dx.doi.org/10.1007/s11432-011-4454-x*

Wang, C., Chen, W. and Liu, Y. (2009), A fuzzy identity based signature scheme, *in* 'International Conference on E-Business and Information System Security, 2009. EBISS '09.', pp. 1–5.

Wang, C. and Kim, J.-H. (2009), Two constructions of fuzzy identity based signature, *in* '2nd International Conference on Biomedical Engineering and Informatics, 2009. BMEI '09.', pp. 1–5.

Xi, K., Ahmad, T., Han, F. and Hu, J. (2011), 'A fingerprint based bio-cryptographic security protocol designed for client/server authentication in mobile computing environment', *Security and Communication Networks* **4**(5), 487–499.

Xi, K. and Hu, J. (2009), Biometric mobile template protection: A composite feature based fingerprint fuzzy vault., *in* 'ICC', IEEE, pp. 1–5.

Xi, K. and Hu, J. (2010), Bio-cryptography, *in* P. Stavroulakis and M. Stamp, eds, 'Handbook of Information and Communication Security', Springer Berlin Heidelberg, pp. 129–157.

Xiong, H., Chen, Y., Zhu, G. and Qin, Z. (2014), 'Analysis and improvement of a provable secure fuzzy identity-based signature scheme', *Science China Information Sciences* **57**(9), 1–5.
URL: *http://dx.doi.org/10.1007/s11432-014-5152-2*

Yang, C., Zheng, S., Wang, L., Tian, M., Gu, L. and Yang, Y. (2014), 'A fuzzy identity-based signature scheme from lattices in the standard model', *Mathematical Problems in Engineering* **2014**. doi:10.1155/2014/391276.

Yang, G., Chen, J., Wong, D. S., Deng, X. and Wang, D. (2008), 'A new framework for the design and analysis of identity-based identification schemes', *Theoretical Computer Science* **407**(13), 370 – 388.

Yang, P., Cao, Z. and Dong, X. (2011), 'Fuzzy identity based signature with applications to biometric authentication', *Computers Electrical Engineering* **37**(4), 532 – 540.

Yang, Y., Hu, Y.-P., Zhang, L.-Y. and Sun, C.-h. (2011), 'Cca2 secure biometric identity based encryption with constant-size ciphertext', *Journal of Zhejiang University SCIENCE C* **12**(10), 819–827.

Yao, Y. and Li, Z. (2014), 'A novel fuzzy identity based signature scheme based on the short integer solution problem', *Comput. Electr. Eng.* **40**(6), 1930–1939.

URL: *http://dx.doi.org/10.1016/j.compeleceng.2013.09.005*

PUBLICATIONS LIST

1. Tan, S.-Y., Jin, Z., Teoh, Andrew B.-J., Goi, B.-M. & Heng, S.-H. (2012), 'On the realization of fuzzy identity-based identification scheme using fingerprint biometrics', *Journal of Security and Communication Networks*, **5**(12), 1312-1324, Impact Factor: 0.311.

2. Tan, S.-Y., Jin, Z. & Teoh, Andrew B.-J. (2012), 'Argument on biometrics identity-based encryption schemes', *Journal of Security and Communication Networks*, **6**(11), 1344-1352, Impact Factor: 0.311.

3. Tan, S.-Y., Chin, J.-J., Heng, S.-H. & Goi, B.-M. (2013), 'An improved efficient provable secure identity-based identification scheme in the standard model', *KSII Transaction on Internet and Information Systems*, **7**(4), 910-922, Impact Factor: 0.560.

4. Tan, S.-Y., Heng, S.-H. and Goi, B.-M. (2011), On the security of two fuzzy identity-based signature schemes, *in* 'IFIP International Conference on New Technologies, Mobility and Security(NTMS)', pp. 1-5.

5. Tan, S.-Y., Heng, S.-H., Phan, R.-W. and Goi, B.-M. (2011), A variant of schnorr identity-based identification scheme with tight reduction, in T.-h. Kim, H. Adeli, D. Slezak, F. Sandnes, X. Song, K.-i. Chung and K. Arnett, eds, 'Future Generation Information Technology', Vol. 7105 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 361-370.

6. Tan, S.-Y., Heng, S.-H., Goi, B.-M. & Phan, Rapahel C.-W. (2014), Schnorr Identity-Based Identification Scheme with Tight Reduction, Manuscript.

7. Tan, S.-Y., Heng, S.-H., Goi, B.-M. & Phan, Rapahel C.-W. (2014), Tight Security for Signature Schemes Provably Secure in the Random Oracle Model, Manuscript.

8. Goi, B.-M., Tan, S.-Y. & Yap, W.-S.(2014), Database-less and Certificate-less Multi-factor Authentication Methods, Patent Application, in filing.