A Fingerprint based Crypto-biometric System for Secure Communication

A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degree

of
BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING

Submitted by:
Mukul Anand Sharma
130001024,
Computer Science and Engineering,
Indian Institute of Technology Indore

Supervised by:
Dr. Somnath Dey,
Assistant Professor,
Computer Science and Engineering,
Indian Institute of Technology Indore

INDIAN INSTITUTE OF TECHNOLOGY INDORE
November 2016

CANDIDATE'S DECLARATION

I hereby declare that the project entitled **"A Fingerprint based Crypto-biometric system for Secure Communication"** submitted in partial fulfillment for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** is an authentic work.

The project was supervised by **Dr. Somnath Dey, Assistant Professor, Computer Science and Engineering, IIT Indore**.

Further, I declare that I have not submitted this work for the award of any other degree elsewhere.

Mukul Anand Sharma
130001024
Discipline of Computer Science and Engineering
Indian Institute of Technology Indore

CERTIFICATE
by BTP Guide

It is certified that the declaration made by the student is correct to the best of my knowledge and belief.

Dr. Somnath Dey,
Assistant Professor,
Discipline of Computer Science and Engineering,
IIT Indore

PREFACE

This report on "A Fingerprint based Crypto-biometric system for Secure Communication" is prepared under the supervision of Dr. Somnath Dey, Assistant Professor, Computer Science and Engineering, IIT Indore and in coordination with Apurv Goel, Bachelors Student, Computer Science and Engineering, IIT Indore.

Through this report, I have tried to provide a detailed description of the techniques that have been used to design crypto-biometric systems for secure communication. Further, I have designed a crypto-biometric system of my own to establish a secure communication among users over a non-secure channel. I have tried to implement this proposed system to the best of my abilities.

I have put my best efforts to explain the proposed system. Implementation and testing of the proposed system are also discussed.

ACKNOWLEDGEMENTS

I would like to express my gratitude towards **Dr. Somnath Dey** for constantly supporting and motivating me throughout the project and for his valuable feedback which helped us in the course of the project.

Also, I would like to thank **Apurv Goel** for coordinating and working with me on this project and making this complete.

I also appreciate the efforts and time given by **Rudresh Dwivedi**, PhD scholar under Dr. Somnath Dey, who guided me many times.

I would also like to thank my family members, friends and colleagues who have been a constant source of motivation. Finally, I offer sincere thanks to everyone who knowingly or unknowingly helped me complete this project.

Mukul Anand Sharma
130001024
Discipline of Computer Science and Engineering
IIT Indore

ABSTRACT

To ensure the secure transmission of data, cryptography is treated as the most effective solution. Cryptographic key is an important entity in this process. In general, randomly generated cryptographic key (of 256 bits) is difficult to remember. However, such a key needs to be stored in a protected place or transported through a shared communication line which, in fact, poses another threat to security. As an alternative to this, researchers advocate the generation of cryptographic key using the biometric traits of both sender and receiver during the sessions of communication, thus avoiding key storing and at the same time without compromising the strength in security. Nevertheless, the biometric-based cryptographic key generation has some difficulties: privacy of biometrics, sharing of biometric data between both communicating users (i.e., sender and receiver), and generating revocable key from irrevocable biometric. This work addresses the above-mentioned concerns.

In this work, a framework for secure communication between two users using fingerprint based crypto-biometric system has been proposed. For this, public key cryptography has been used in this approach to generate cryptographic key from fingerprint biometrics of both communicating users. Diffie-Hellman algorithm of public key cryptograpy is used for generating a common cryptographic key. This fingerprint-based cryptographic key can be applied in symmetric cryptography where session based unique key is required. In this approach, revocable key for symmetric cryptography is generated from irrevocable fingerprint. The core advantage of this approach is that biometric data is neither stored nor shared which ensures the security of biometric data, and perfect forward secrecy is achieved using session keys. This work also ensures the long term security of messages communicated between two users.

CONTRIBUTION

This project is done over a period of six months from June, 2016 to November, 2016 with complete survey, analysis and implementation of the proposed system. First two months were spent on literature and survey work. In next few months, we spent our time on design of proposed framework for secure communication using crypto-biometric system. We studied existing works in this domain and designed our framework with the goal to eradicate limitations of existing approaches. In the last two months, we hav devoted our time on implementation of proposed framework and its testing upon parameters like accuracy and security.

As I already mentioned that first two months we both did the literature survey. We proposed the framework for establishing secure communication among users and started designing it. Later, I implemented the feature extraction part and proposed algorithm was implemented by Apurv. We started testing it on various parameters like randomness of cryptographic key, accuracy of key etc. This report sums up the work we have done in the duration of this project.

Mukul Anand Sharma
130001024
Discipline of Computer Science and Engineering
IIT Indore

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Secure transmission of data over a network is crucial in communication technology. Privacy of shared information becomes an important issue in this regard. Generally, cryptography is used for ensuring privacy of data. Messages are encrypted before sharing over a network and decrypted at the other end by the receiver. There are mainly two types of cryptography as shown in Fig. 1.1 : symmetric and asymmetric cryptography. Symmetric cryptography is the one in which encryption and decryption keys are same while in asymmetric cryptography two keys are used, public key is used for encrypting the messages while private key is used for decrypting the messages. Examples of symmetric cryptography are DES, AES etc. and example of asymmetric cryptography is RSA algorithm [1].

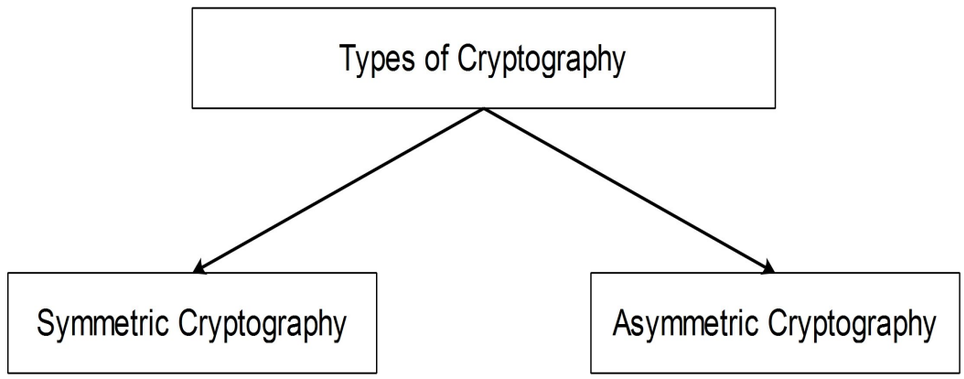

FIGURE 1.1: Types of Cryptography

For better security of a cryptographic system, keys used in encryption and decryption must be long enough to be unbreakable. Knowledge-based(key is remembered by user) and possession-based(key stored in smart card etc.) authentication systems are not secure due to the fact that long keys cause user inconvenience to remember and smart cards can be stolen or misplaced. Moreover, storing long keys on a system is costly and not secure. Biometrics-based authentication systems can alleviate the limitations of above mentioned systems [2].

A biometric is a measurable biological characteristic of a human being to uniquely identify one in the world. Fingerprints, facial structure, iris etc. are such unique characteristics which are used to generate biometrics for an individual [2]. Biometric traits are mainly used for authentication of a person's identity. Systems combining biometrics with cryptography are called Crypto-biometric systems.

Crypto-biometric systems are required to be revocable and secure in terms of privacy of an individual's biometric. If a person's biometric is compromised then it's useless forever. Moreover, biometrics are irrevocable but crypto-biometric systems are supposed to be revocable so that multiple instances of a person's biometric can be generated not just one. For this purpose cancelable biometrics are the solution [1]. Some transformation is applied on the original biometric template of an individual to generate cancelable biometric template such that this transformation is irreversible. This way, no one can get the original biometric from cancelable

biometric and multiple cancelable biometrics can be generated from a single original biometric in case one's cancelable biometric is compromised. This provides revocability to the irrevocable biometric.

When a user A wants to send a message to user B, A first encrypts the message using a key K and then sends this encrypted message to B. B can decrypt this message using key K only. For this, either the key K or some information to generate the same key K at both ends (A and B) have to be shared between two communicating users. In both cases, sharing of some information is required. So, there is need to securely share information over non-secure communication channel.

In crypto-biometric systems, biometric is integrated with cryptography using either key-generation techniques in which cryptographic key is generated from one's biometric or key-binding techniques in which cryptographic key is binded to a biometric and shared [3]. Most of the existing work focus either on key binding techniques or key generation techniques. Fuzzy vault [19] and fuzzy commitment schemes [6,7] are some good key-binding techniques. Few approaches have been proposed to generate cryptographic key from biometric traits [8-12].

Very little work has been proposed about a framework for secure communication on a network using crypto-biometric system. Barman et al. [17] proposed a system in which both sender and receiver exchange their cancellable biometrics using key-based steganography. Kanade et al. [18] proposed a crypto-biomtric system for establishing secure communication session between two clients. Their method involves CARA (Central Authority for Registration and Authentication) with which the clients are registered. Most of such work focuses on biometric based authentication techniques or using biometrics for generating cryptographic keys. Such approaches provide security for communication channel but also have some limitations. Storing of biometric templates is one of such issues which should be avoided. Security is very important for such systems which cannot be taken lightly. Ratha et al. [20] identified eight points of attack in biometric systems. A crypto-biometric system should be secure from all such possible attacks. Moreover, it should also provide privacy to biometrics of users along with generating revocable and non-invertible cryptographic key from biometric data of users.

This work aims to address above mentioned concerns. In this work, a complete framework for secure communication among users on a network using crypto-biometric system has been proposed to provide perfect forward secrecy. In cryptography, perfect forward secrecy is a property of secure communication protocols in which compromise of long-term keys does not compromise past session keys [42]. Perfect forward secrecy protects past sessions against future compromises of secret keys or passwords [43]. A crypto-biometric system for secure communication among users will require 1) generation of cancelable biometrics such that original biometric information of two users is never disclosed, 2) generation of unique cryptographic keys from cancellable biometrics of both sender and receiver, and 3) secure transmission of keys among users.

In this approach, public key cryptography has been used to generate revocable and non-invertible symmetric cryptographic key from fingerprints of users. For this, Diffie-Hellman algorithm of public key cryptography has been used to generate symmetric cryptographic keys. This approach starts with fingerprints of sender and receiver. Feature extraction of fingerprints is done using pair-minutiae vectors [21] and then revocable biometric templates are generated at both ends. This revocable template is hashed to generate a private key for the user. Diffie-Hellman algorithm is used to generate public keys from private keys of both sender and receiver which are shared and furthur used to generate a symmetric cryptographic key at both ends. This approach also involves a central authority(CA) for authentication of users. At the time of registration with CA, RSA key pair is generated for each user. RSA public key and identity of user along with its hash are signed

by CA. This information is later used by users to verify one another before setting up a connection.

This proposed system ensures the avoidance of biometric template storage, either in a central database or a smart-card. Revocability is provided to cryptographic key with transformation key to generate a revocable biometric template. Use of a central authority and public key cryptography algorithms like Diffie-Hellman and RSA provide security against various attacks including man in the middle attack. An overview of the proposed work is shown in Fig. 1.2.

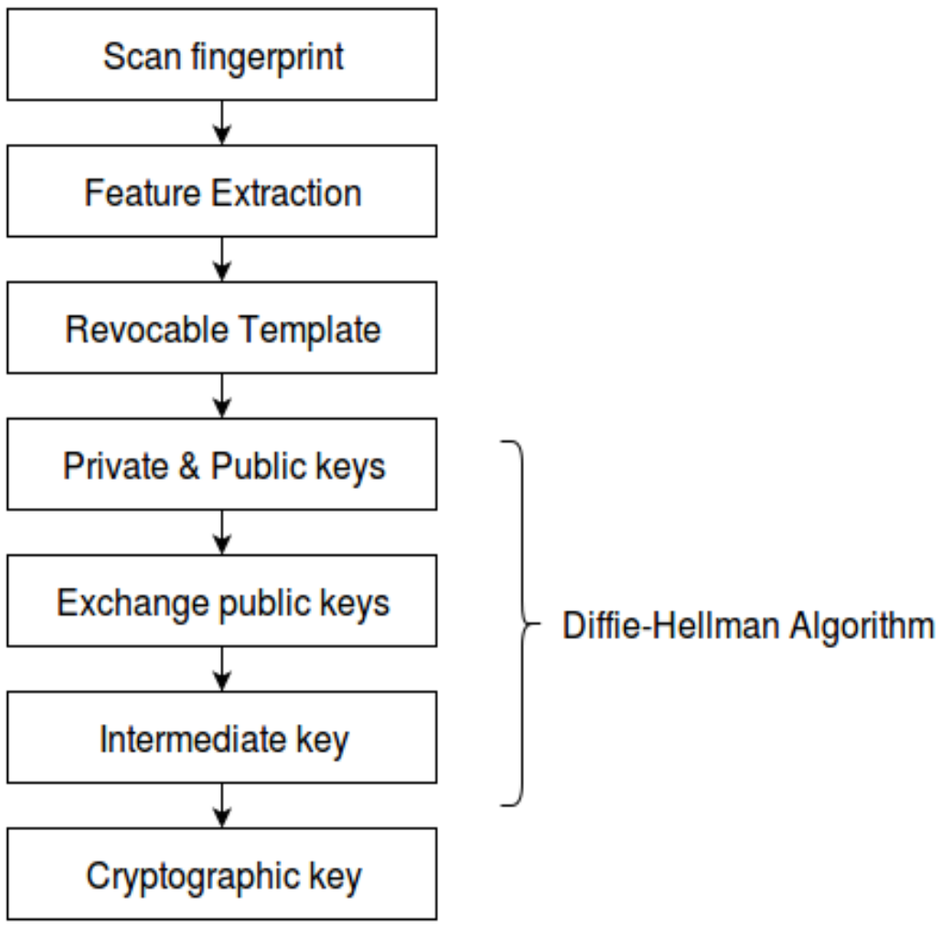

FIGURE 1.2: An overview of proposed framework

The rest of the thesis is organized as follows. A brief review of existing related work is given in section 2. Section 3 explains the proposed approach in detail. Experimental results and security analysis of this approach are given in section 4 and 5 respectively. Finally, the thesis is concluded in section 6.

Chapter 2

Related Work

As discussed in the previous section, a crypto-biometric system can be studied as per the following steps: 1) feature extraction, 2) cancelable template generation and 3) cryptographic key generation. Below is a brief discussion of some existing work related to each of these steps. Over the years, few work has been done on crypto-biometric systems for secure communication over a network. Some of these are also discussed below in brief.

2.1 Feature Extraction

Feature extraction plays a vital role in biometric authentication systems. Feature extraction is the process of defining a set of features, or image characteristics, which will most efficiently or meaningfully represent the information that is important for analysis and classification. In simple words, feature extraction is done so that one can represent a biometric image as a numeric value (feature vector). Feature vectors are required to be very accurate to represent a biometric image uniquely.

Only fingerprint based feature extraction techniques are discussed here. Several factors make feature extraction a very challenging problem [22]: image noise, skin conditions, distortions, rotation, displacement, etc. There are two well-known properties in fingerprints: large variability in different impressions of same finger (large intra-class variations) and much similarity between two images from different fingers (small inter-class variations) [4]. Minutiae based features are most widely used in fingerprint feature extraction techniques. Simplest features considered are minutiae point co-ordinates (x,y) and orientation angle (θ). A lot of work has been done on fingerprint minutiae matching techniques. Fingerprint minutiae matching methods can be divided into global and local minutiae matching methods. Global matching techniques try to align all minutiae points before matching while local matching techniques try to match local minutiae structures which are characterized by attributes that are invariant to global transformations like rotation and translation. Local matching techniques achieve distortion tolerance and low computational complexity.

Local matching is about extracting features based on local structure of minutiae points to achieve invariance regarding global transformations. Most of the existing work in this regard use NN (nearest neighbors) [23,24], minutiae triplets [25,26], minutiae cylinder [27,28], fixed radius based [29,30] and texture based [31,32] features as mentioned in survey [4]. Some non-minutiae based matching techniques use correlation-based fingerprint matching [].

2.2 Cancelable template generation

As discussed in chapter 1, cancelable biometric template generation is required to ensure privacy, security and revocability of biometric data. After Ratha et al.

[33] formally defined the problem of cancelable biometric, various types of approaches have come into picture regarding generation of cancelable biometric templates. Ratha et al.[5] proposed three transformations on feature domain to generate cancelable fingerprint templates : Cartesian, polar and functional transformations. Shuffling-based transformation using a user-specified random key are proposed based on iris biometric features to generate cancelable biometrics in [6,7]. In these methods, iris codes are divided into blocks and then shuffled using the user-specified random key to generate cancelable biometric template. Some bit-string type methods are proposed in [34,35]. In such methods, invariant features are extracted from fingerprint using minutiae based local feature extraction techniques, then quantized and bin indexed to generate a bit string. Farooq et al. [36] proposed triangle-based alignment-free method to generate cancelable templates in the form of bits strings.

2.3 Cryptographic key generation

There have been numerous proposals for generating cryptographic keys from biometrics. Generally, key generation techniques from biometric data follow similar design. A biometric template is generated from biometric data using some feature extraction technique. This biometric template is used for generating a unique key or binding a key to it which is regenerated later at the time of verification of biometric. Some examples of key generation systems are [8-12]. Some key regeneration(key binding) techniques are [6,7].

Few key generation techniques have been proposed over the years using various biometrics [8-12]. Davida et al. [40] proposed an approach that uses iris codes. Monrose et al. [8] proposed an approach to generate cryptographic key from user's voice while speaking a passphrase. Feng et al. [9] proposed an approach to generate private key using a user's online signature. Many other approaches have been proposed over the years using different types of biometrics such as face [10,12], iris [11,14,15], fingerprints [37,38] etc. In recent research [13-15], multimodal or multiple biometrics is used in crypto-biometric systems. Most of these proposed methods are for generating a key from cancelable biometrics.

Symmetric key generation has been the focus of most of the above mentioned work. These symmetric keys are further used for encryption and decryption purpose. Biometric data can also be used to generate key pairs for use in public-key cryptography. Sharda et al. [16] proposed an approach for generating RSA key pair using combination of fingerprints. Sayani et al. [39] propose an approach to generate RSA key pair using fingerprint biometric to strengthen the security of messages over a network. In some cases, unique key generated from biometric data can be used as a private key in a public key cryptography algorithm, public key can be generated using this private key.

2.4 Crypto-biometric system framework for secure communication among users

There exist few work related to frameworks for secure communication over a network using crypto-biometric systems.

Barman et al. [17] proposed a framework in which both sender and receiver exchange their cancelable biometrics using key-based steganography. Both cancelable templates are then merged together using concatenation-based feature level fusion technique to generate a combined template. Shuffle key is used to randomize the elements of the combined template and this shuffled template is fed to a

hash function to generate the cryptographic key. In this approach, shuffle key provides one more level of revocability. But, there are problems with this approach. First, biometric data is shared among users through non-secure communication channel. Second, this framework doesn't prevent man in the middle attack. If a person attacks the network and gets access to shared data then by intercepting both users' cancelable biometrics the attacker can act as another user using its own biometric and setting up two connections simultaneously, with sender and receiver both. Sender and receiver will never get to know if they are communicating to each other or some man in the middle. This way, their cancelable biometrics can also be used to setup connections with other users and communicating on behalf of them. Although, author has done analysis of this attack but it is based on the assumption that shuffle key is always securely transmitted beforehand using public key cryptography.

Kanade et al. [18] proposed a crypto-biometric system for establishing secure communication session between two clients. Their method involves CARA (Central Authority for Registration and Authentication) with which the clients are registered. CARA stores cancelable biometrics of users registered with it. Whenever two users want to communicate they first get authenticated by CARA which then provides session keys to both users. These session keys are then used for transformation on cancellable biometrics of one user and sent to other user. BSKGS protocol [41] is used then to generate cryptographic key from this. This approach uses CARA as authority which provides a single point of failure i.e. if an attacker gets access to CARA database then all users' biometric data is compromised which can be used to communicate with any user registered with CARA on behalf of any other user. This makes this system vulnerable, a user's original biometric is secured but their privacy is not.

Chapter 3

Proposed methodology

As discussed in previous chapter, not much work has been done on development of framework for secure communication between two users. Existing work in this regard possesses various problems which are also discussed in last chapter. This has been the motivation of the work in this project. The goal is to develop a framework which ensures secure communication between two users and overcomes the limitations of existing technologies discussed previously. In this section, the proposed work is discussed in detail.

An overview of the proposed work is given in Fig. 1.2. This approach starts with fingerprint biometric of sender and receiver. Initially, feature extraction of both sender's and receiver's biometric is done using pair-minutiae based feature extraction method. Feature binary string is obtained after quantization and binning. Next, a random key based permutation is applied on feature bit string to obtain a permuted binary string. This binary string is hashed using SHA256 to generate a 256 bit key input for next step which is diffie-hellman algorithm.

Diffie-Hellman algorithm is used for generating symmetric keys at both sender and receiver ends. Hashed revocable biometric templates of sender and receiver are termed as private keys of diffie-hellman algorithm. These private keys are fed to diffie-hellman algorithm along with two predefined parameters to generate public keys of sender and receiver. These public keys are then shared between sender and receiver. Diffie-Hellman algorithm uses user's own private key and other user's public key to generate a symmetric key at both users end. This key is termed the intermediate key which is further hashed to generate final cryptogrphic key in our technique. This key is then used for encryption and decryption of information to be shared between sender and receiver.

This system also involves authentication of users before starting communication among them. For this, a central authority (CA) for enrollment and verification of users has been proposed. At the time of registration, a user generates an RSA public-private key pair and shares this public key with CA along with some identification. CA registers the user with all this information and provides a signed certificate to the user. This certificate is used by users to verify each other before setting up the connection.

The above mentioned steps of the proposed framework are stated in detail in the following sections. Fig. 3.1 gives the detailed diagram of the proposed framework.

3.1 Feature extraction from fingerprint image

This is the first step in this framework for secure communication. As keys need to be generated from an individual's biometric, a few bits variation in the key generation can lead to wrong outcomes at the time of decryption of messages. So, such features are needed to be considered which are invariant to affine transformations of fingerprint images. For this purpose, feature extraction technique proposed by Wang in [21] has been used. They propose a transformed version of this technique

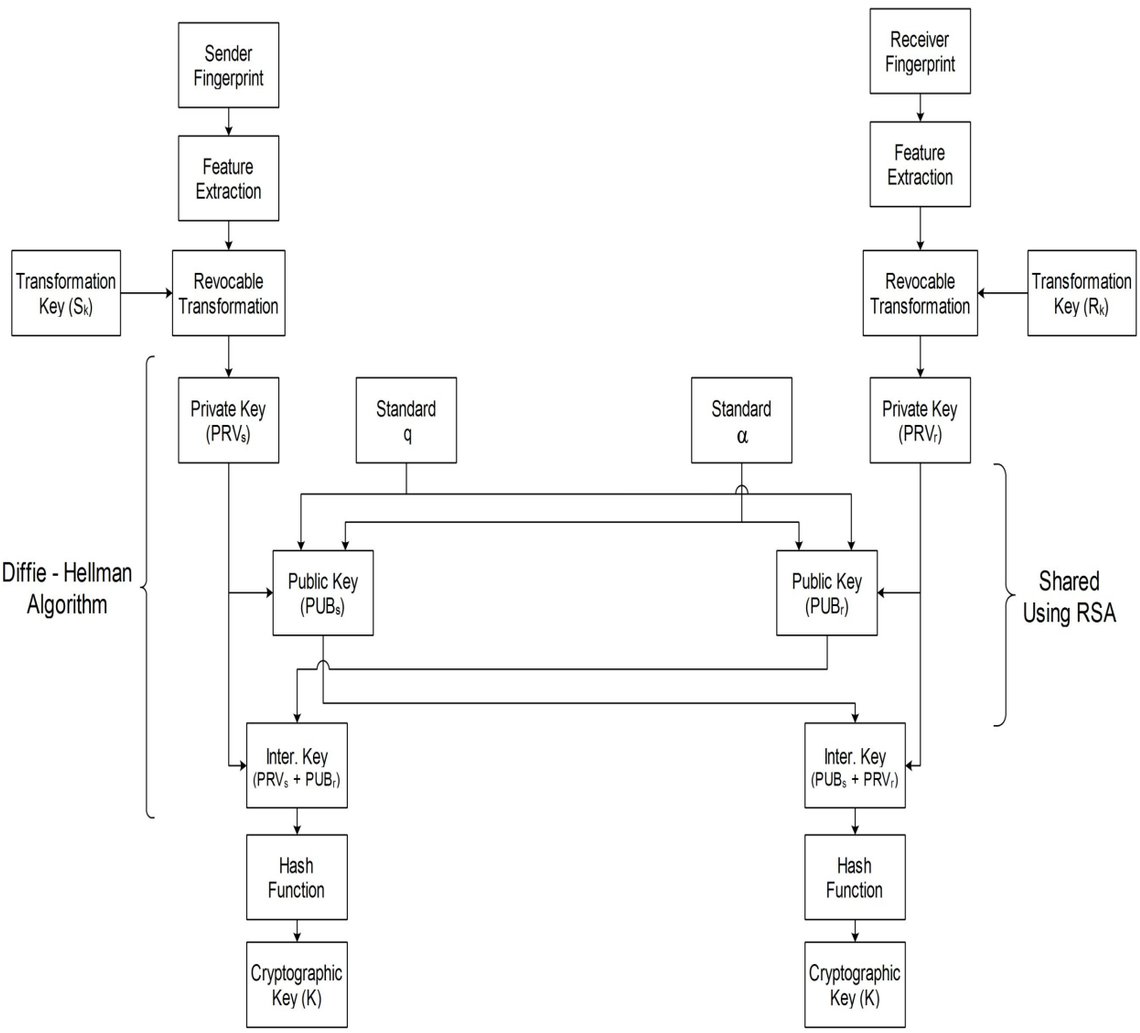

FIGURE 3.1: Proposed Crypto-biometric system framework

which was originally proposed by Jin in [35]. In this technique, feature extraction is done using minutiae-pairs, then quantization and binning to generate a binary-bit string. Details of this technique are as follows.

Let minutiae points are extracted from fingerprint image and a set of minutiae is selected such that distance between a set of minutiae is not less than a small threshold. Let this set be denoted by

$$M = \{M_{\mathrm{k}}(x_{\mathrm{k}}, y_{\mathrm{k}}, \theta_{\mathrm{k}})\}^{\mathrm{m}}{}_{\mathrm{k=1}} \tag{3.1}$$

where m is number of minutiae in set, x_k, y_k, θ_k are x,y coordinates and orientation of k^{th} minutiae, respectively. A pair minutiae vector V_{ij} can be formed by pairing up two minutiae M_i and M_j from set M. There will be *(m(m-1)/2)* pairs constituting the set V which can be expressed as

$$V = \{V_{\mathrm{ij}} : 1 \leq i, j \leq mandi \neq j\} \tag{3.2}$$

where each V_{ij} is triplet of distance and relative angles of minutiae pair (M_i, M_j), assuming the reference direction of line segment connecting minutiae pair is from M_i to M_j. Hence, V_{ij} is defined as

$$V_{\mathrm{ij}} = \{L, \alpha_{\mathrm{i}}, \beta_{\mathrm{j}}\} \tag{3.3}$$

where L is the distance between minutiae pairs M_i and M_j, α_i is the angle between reference direction of line segment joining M_i and M_j and the orientation of M_i in the counter-clockwise direction, and β_j defined analogously. Range of α and β is between 0 and 2π. Fig. 5 illustrates this triplet formation.

To determine V_{ij}, the following two quantities X and Y are calculated first :

$$X = (x_\mathrm{j} - x_\mathrm{i})cos\theta_\mathrm{i} + (y_\mathrm{j} - y_\mathrm{i})sin\theta_\mathrm{i}$$

$$Y = (x_\mathrm{j} - x_\mathrm{i})sin\theta_\mathrm{i} - (y_\mathrm{j} - y_\mathrm{i})cos\theta_\mathrm{i}$$

Based on X and Y, V_{ij} = (L,α_i,β_j) is obtained then

$$L = \sqrt{X^2 + Y^2} \tag{3.4}$$

$$\alpha_\mathrm{i} = arctanY \div X \tag{3.5}$$

$$\beta_\mathrm{j} = \alpha_\mathrm{i} + \theta_\mathrm{j} - \theta_\mathrm{i} \tag{3.6}$$

Here, pair-minutiae vector V is obtained. Next, quantization is applied on each V_{ij} in V. Quantization step size is finalised for each term in triplet (L,α_i,β_j) and each term is represented in binary notation. Suppose n_l,n_α,n_β are number of bits required to represent L,α,β in binary notation respectively. Then the total number of bits to represent each V_{ij} in V will be

$$n = n_\mathrm{l} + n_\alpha + n_\beta \tag{3.7}$$

Thus, for each pair-minutiae vector V_{ij} in V, a binary representation $V_{ij}^{(b)}$ of n bits can be found. $V^{(b)}$ set of $V_{ij}^{(b)}$ can be denoted as

$$V^{(\mathrm{b})} = \{V_{\mathrm{ij}}^{(\mathrm{b})} : 1 \leq i, j \leq m \& i \neq j\} \tag{3.8}$$

Based on the experiments done in [21], n=15 bits gives the most optimal results, 5 bits for each L, α, β. In this work too, n=15 has been used.

Next, binning is applied on binarized pair-minutiae vector set. Since there are 2^n possible combinations of n bits, binning from 00...0 to 11...1 is done. For each $V_{ij}^{(b)}$ in $V^{(b)}$, index a bin by 1 if V_{ij} falls in it. Only the bins indexed once are assigned 1 and all other bins are assigned 0. At the end of this process, a binary string h_k of length 2^n is obtained in which 1's correspond to the unique occurrence of those $V_{ij}^{(b)}$.

This binary string h_k is considered as the feature vector. This bit string is generated using pair-minutiae features and is invariant to affine transformations in the fingerprint image.

3.2 Generation of revocable biometric template

In crypto-biometric systems, biometrics of users are transformed to cancelable templates to ensure security of biometric data. Cancelable template generation provides revocability and non-invertibility to the original irrevocable biometric of a user. This template is used for sharing or matching purpose which can easily go in hands of adversaries. Cancelable template generation is essential to provide security to biometric data of users from such adversaries.

However, in this proposed framework, there is no need to share or store biometric data. The process of key generation for sender and receiver takes fingerprint image

as input and outputs a public key which is shared among users and intermediate key. During this process, no information is saved on a disk which an adversary can get their hands on. So, we need to make the feature bit string of biometric revocable only. DH algorithm automatically provides non-invertibility to the input biometric data so there is no need to perform it here. Hence, only a revocable biometric template is generated from feature bit string of a user.

For providing revocability to the feature bit string generated after feature extraction step in previous subsection, permutation is applied on the bit string based on a random key. A random key is generated at user end which is termed as transformation key for revocable template generation. This transformation key is used as seed value to generate random numbers upto feature bit string length. Bits corresponding to these random numbers are swapped with bits at positions strating from the start and incrementing with each random number.

Lets understand this with an example. Say we have a bit string of length 16 bits. We take T as transformation key which is used as seed to generate random numbers from 1 to 16. Say first random number generated is 5. We swap bit at 5th position with 1st position bit. Let next random number be 11. Now, 2^{nd} bit is swapped with 11th bit. This process continues upto 16 random number generations and each bit starting from the start upto last bit is swapped with bit at position equal to new random number. This way, a bit string is generated which is permutation of the original bit string. Using this technique, all possible permutations of original bit string can be generated. Also, for a given seed value, same set of random numbers are generated in the same order. So, this seed is termed as transformation key of feature bit string because transformation on a feature bit string using the same transformation key provides us the same permuted bit string.

This method ensures the generation of a revocable template from feature bit string of the user's biometric. As this method depends on randomly generated transformation key T for cancelable template generation, T provides revocability to the biometric template with respect to a biometric. A user can use different transformation key T to generate a different template from same feature bit string.

In this proposed work, this revocable biometric template is hashed using SHA256 hash to generate a 256-bit private key for the user. This key is never shared and remains with the user only. Next we apply Diffie-Hellman(DH) algorithm using this private key to generate a public key for the users. Diffie-Hellman algorithm is discussed in next subsection and use of this algorithm in our proposed work follows after.

3.3 Generation of public key and cryptographic key using Diffie-Hellman algorithm

Diffie-Hellman is an algorithm which is used to enable users to securely exchange a cryptographic key over public channels. This is one of the first public-key protocols as originally conceptualized by Ralph Merkle and named after Whitfield Diffie and Martin Hellman. The Diffie-Hellman algorithm depends for its effectiveness on the difficulty of computing discrete logarithms [1].

Diffie-Hellman algorithm is stated as :

1. Take a prime number q
2. Take an integer α such that α < q and α is a primitive root of q
3. For user A, select a random integer X_A < q
4. Calculate $Y_A = \alpha^{X_A} \bmod q$
5. For user B, select a random integer X_B < q

6. Calculate $Y_B = \alpha^{X_B} \bmod q$
7. Y_A & Y_B are public keys and $X_A \& X_B$ are private keys of user A & B respectively
8. Public keys Y_A & Y_B are shared among each other
9. Calculation of encryption key K by A as $K = (Y_B)^{X_A} \bmod q$
10. Calculation of decryption key by B as $K = (Y_A)^{X_B} \bmod q$
11. K is the symmetric cryptographic key

The security of Diffie-Hellman lies in the fact that, while it is easy to calculate exponentials modulo a prime, it is very difficult to calculate discrete logarithms. For large primes, the latter task is considered infeasible.

3.4 Generation of cryptogaphic key using Diffie-Hellman (DH) algorithm

As discussed in section 2.3, most of the existing key generation techniques focus on generating symmetric keys. Problem with these techniques is that they require to share this key through non-secure communication channel. In most of these methods, they assume a pre-established link setup between sender and receiver to share the cryptographic key. Some techniques generate symmetric keys at both ends but they share biometric data between sender and receiver. Sharing of biometric data should be avoided in crypto-biometric systems as it violates the privacy and security of user's biometric information.

Here, the proposed approach is a method of generating symmetric keys at both sender and receiver ends using public-key cryptography algorithm, Diffie-Hellman(DH) algorithm. In this approach, revocable biometric template of a user is hashed to get private key of the user which is never shared over the network. Advantages of using DH algorithm for key generation and sharing are : no sharing of biometric data is required and only public key is transmitted over the network which even if compromised, doesn't reveal any information about the private key i.e., revocable biometric data. This algorithm itself provides non-invertibility on biometric template.

Revocable biometric templates(binary strings) of both sender and receiver are available. As discussed in subsection 4.1, n is taken to be 15 in our approach to get a feature bit string of length 2^{15}. Same is the length of revocable template which is premuted feature bit string. This large binary string needs to be mapped into a smaller one which can be used as key input for DH algorithm. For this, SHA256 hash has been used here. Permuted binary string is hashed using SHA256 to generate a 256-bit key. This key is termed as private key of the user. This way, private keys of sender (PRV_S) and receiver (PRV_R) are generated. Next comes the cryptographic key generation part.

As discussed in previous section, DH algorithm requires a large prime number q and its primitive root α. These parameters are not required to be generated in each session, we can also use fixed value of these parameters over a large number of sessions(Appendix A). In this approach, DH parameters of RFC 3526: 2048-bit MODP group[45] have been used. With private keys PRV_S, PRV_R, q and α, step 4 and 5 of DH algorithm in section 4.3 are applied to generate public keys PUB_S and PUB_R of sender and receiver respectively.

Public keys PUB_S and PUB_R are then shared between sender and receiver. Once both sender and receiver have each other's public keys, DH algorithm is applied at both ends to generate a secret key using own private key and other's public key, as explained in step 9 and 10 of DH algorithm in section 4.3. This way, both

sender and receiver derive a secret key. This key is termed as intermediate key for the communication setup. In our approach, size of this key is 2048 bits. This intermediate key is hashed using SHA256 to generate a 256-bit key which is the final cryptographic key. This key is then used for encryption and decryption of messages between sender and receiver.

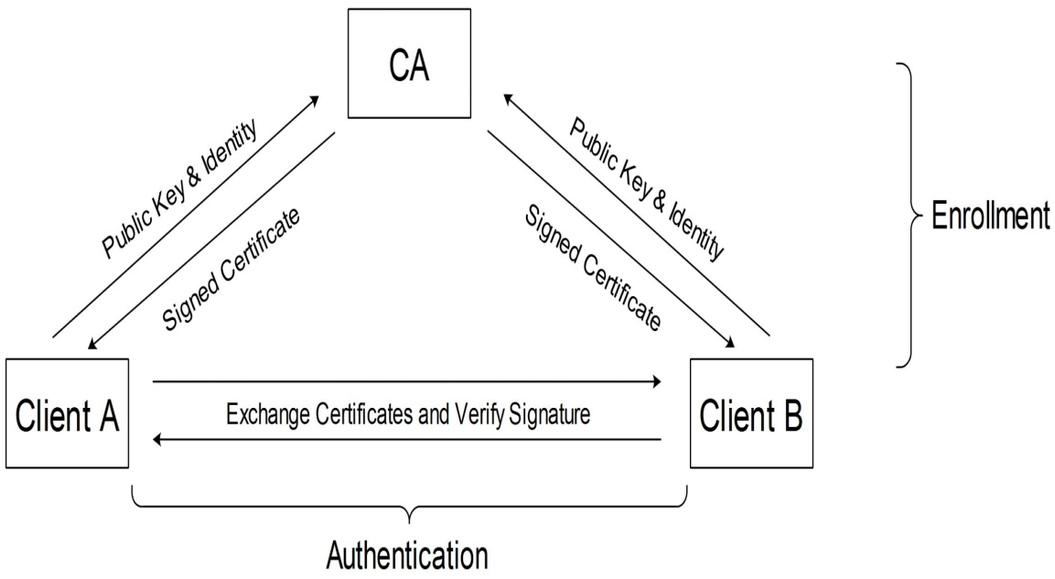

FIGURE 3.2: Enrollment and Authentication using CA

3.5 Authentication using CA

The proposed crypto-biometric system also involves a central authority (CA) with which all users need to be registered. When a new user joins the system, CA requires to enroll it first. An overview of this is given in Fig. 3.1.

CA has a pre generated RSA public-private key pair of its own. Public key of CA is shared among all users in the system. This enrollment phase can be divided into following steps :

1. User generates its own set of RSA public-private key pair and sends its public key along with its identification to the CA after encrypting it with public key of CA.
2. CA identifies the user using this information and stores this identification in its database.
3. CA computes hash of public key and identification of the user and encrypts this hash, public key and identification of the user using its private key.
4. This encrypted message is termed as certificate of the user and is sent to the user.

All users enroll with the CA to get their certificates. These certificates are used for verification of other users before setting up a connection with them. Suppose user A wants to communicate with user B. Verification phase before setting up this connection can be divided into following steps :

1. A sends its certificate to the user B with a request to initiate the communication.
2. B decrypts A's certificate with CA's public key and computes hash of A's public key and identification. This hash is matched with hash in the certificate to verify that this certificate is indeed signed by the CA.

3. B then identifies A using its identification and then send its certificate to A.
4. A does the same steps as B to verify B. Once verified by each other, they can start setting up the communication using proposed approach discussed in above subsections.

Chapter 4

Experimental setup and results

In previous section, the proposed work on a framework for secure communication using crypto-biometric system was discussed. In this section, implementation of this work and experimental results are discussed. Database and experimental setup are discussed first. Next, results of experiments for a number of cases are discussed. Results are calculated and compared with existing techniques and also for all possible cases of attacks.

4.1 Database

The proposed method was evaluated using the FVC2002 databases [44] (DB1, DB2, DB3).

A unique pair of fingerprints is taken from a database as fingerprints of sender and receiver. Genuine cryptographic key is generated using these fingerprints. Rest of the pairs of fingerprints of same subset of the same FVC database are taken to generate imposter key. This way, all pairs of genuine fingerprints and corresponding pairs of imposter fingerpritns are taken. For testing accuracy of key generation, 8 instances of a fingerprint are used for each fingerprint in databases DB1, DB2 and DB3.

4.2 Experiments and results

In this section, experimentation of proposed system is discussed upon the following parameters : randomness of private key for different transformation keys, randomness of cryptographic keys on using imposter pair of fingerprints, and accuracy of cryptographic key generated using genuine and imposter fingerprints. These results are discussed as follows.

4.2.1 Randomness of private key for different transformation key

For this experimentation, first instance of fingerprint for all subjects in set A of databases DB1, DB2 and DB3 has been used. For each fingerprint, private key is calculated for a transformation key T. Then, private key is calculated for 30 different randomly generated transformation keys for this fingerprint and hamming distances of private key against the first private key generated using transformation key T are calculated. This way, 30 hamming distances are calculated for each subject in set A of databases DB1, DB2 and DB3. Since, set A contains 100 subjects, a total of 9000 hamming distances are calculated, 3000 for each database. Histogram of these results is plotted which is shown in Fig. 4.1. It can be observed from the histogram that mean hamming distance is 50.03% which means that average hamming distance between two private keys generated using different transformation keys is 128 bits. Hamming distances are spread between the range of 37.11% to

64.06% with a standard deviation of 0.031. For change in transformation keys, 40% to 60% bits of private key are different in 99.85% of cases. So, on changing transformation key for a given biometric, upto 128 bits of private key are changed in most of the cases.

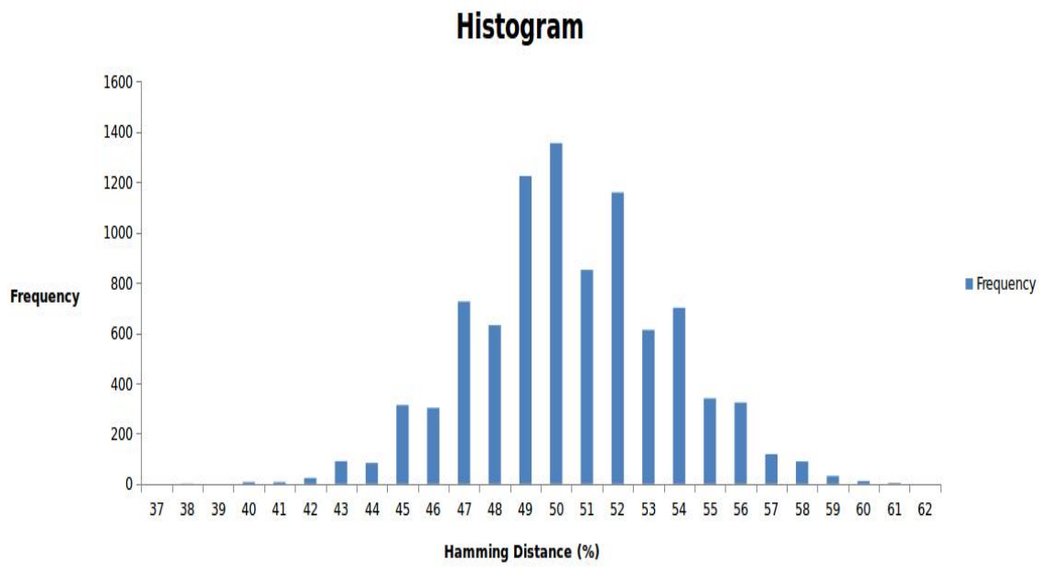

FIGURE 4.1: Hamming distances among private keys for different transformation keys

4.2.2 Randomness of cryptographic key for imposter pair of fingerprints

For this experimentation, set A of databases DB1, DB2 and DB3 are considered. Set A has 100 fingerprints of 100 subjects in each database. These fingerprints are divided into 50 unique pairs. For each pair, that particular pair is considered genuine while rest of the pairs are considered imposter pairs of fingerprints. This way, a total of 50*49=2450 possible combinations are possible for each set A. Hamming distances of cryptographic keys generated using genuine and imposter keys are calculated. This is calculated for each combination of genuine and imposter pair of fingerprints in databases DB1, DB2 and DB3. Hence, a total of 2450*3=7350 hamming distances have been calculated which are plotted as histograms in Fig. 4.2. It can be observed from the histogram that mean hamming distance is 49.94% which means that average hamming distance between genuine and imposter keys is 128 bits. Hamming distances are spread between the range of 37.89% to 61.72% with a standard deviation of 0.031. As per the quantity of imposter keys, 40% to 60% of the bits of genuine keys are different from 99.89% imposter keys. A small number (0.04%) of imposter keys have unmatched bits below 40%. Hence, an imposter cannot get more than 128 bits of 256-bit cryptographic keys in most of the cases.

4.2.3 Accuracy of cryptographic key generation

In this experiment, accuracy of cryptographic key for different instances of same fingerprint is tested and compared with cryptographic key for imposter fingerprints. For this, actual experimentation is done on the feature bit strings generated from fingerprints. Set A of databases DB1, DB2 and DB3 are considered. Each such set A has 100 subjects with 8 instances of each subject's fingerprint. First two fingerprints of each subject are taken as genuine pair of fingerprints and rest

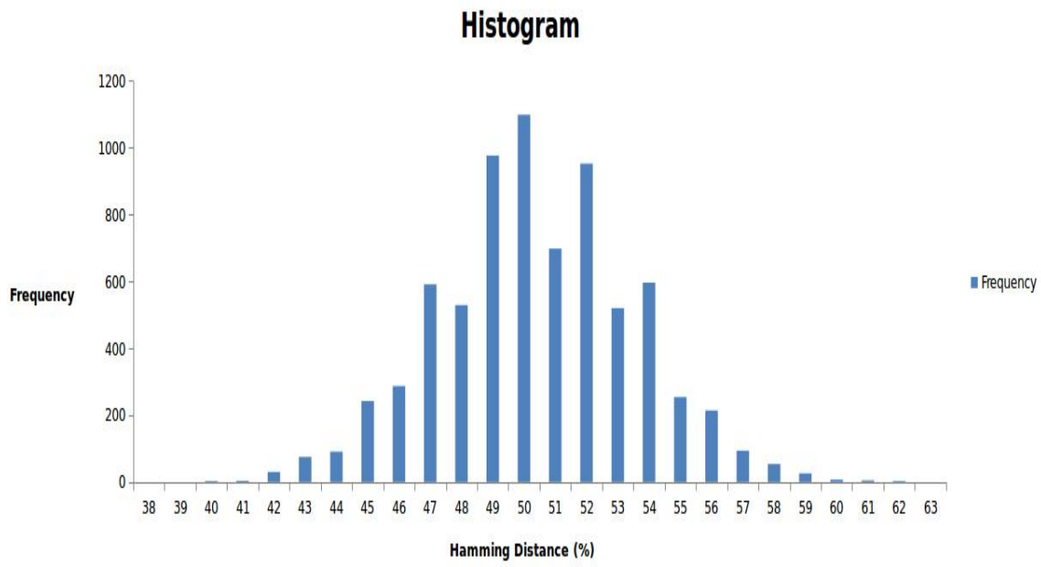

FIGURE 4.2: Hamming distances between genuine and imposter keys

of the subject's first fingerprints as imposter fingerprints. For each subject, hamming distances of feature strings generated from genuine pair of fingerprints (first two) and hamming distances of feature strings generated from first genuine fingerprint and rest of the imposter fingerprints are calculated. This way, a total of 100 hamming distances of genuine pair of fingerprints and 99*100=9900 hamming distances of a genuine and an imposter pair of fingerprint have been calculated for each database. These hamming distances actually tell about the difference in number of bits in feature strings generated from a pair of fingerprints. For a genuine pair of fingerprints, hamming distances must be minimal while for a pair genuine and imposter fingerprint, hamming distance must be higher. Average of these values is taken and result is termed as error rate in key generation. This experiment in divided into two: 1) transformation key is different for imposter fingerprints and same for genuine fingerprints and 2) transformation key is same for imposter fingerprints. Experimentation was done for different values of n and best results were obtained for n=12. Experimental results are shown in table 4.1.

TABLE 4.1: Error rate in key generation (in %)

Database	Genuine	Imposter	
		Same transformation key	Different transformation key
DB1	9.28	15.61	31.06
DB2	11.74	17.89	36.51
DB3	9.29	16.65	24.39

As it is clear from the results, for genuine pair of fingerprints average error rate in key generation lies in range of 9.2% to 11.8% while for imposter fingerprints, it is much larger. For first case where transformation keys are different for imposter fingerprints, error rate lies in the range of 24.3% to 36.5%. For second case where transformation keys are same for imposter fingerprints, error rate reduces in comparison to first case but still is significantly higher which is in the range of 15.6% to 17.9%. From these results, it can be stated that if an imposter fingerprint is used for generating same key as of a genuine fingerprint, it cannot be done as in most of the cases the imposter key will be different from genuine key by atleast15-16%. In case of genuine pair of fingerprints i.e., different instances of fingerprint of the same subject, error still remains in the key generation but is lower and there is scope of further optimizing it using other feature extraction techniques to reach an error rate of 0%.

Chapter 5

Security Analysis

In last chapter, experiments on randomness of cyptographic key in various scenarios using the proposed system were discussed. Hamming distance histograms were plotted for each scenario. This section focuses on security of the proposed system. Privacy of biometrics of users is discussed first. Next comes the discussion about security of proposed system in various possible cases of attack.

5.1 Privacy of biometrics

The core benefit of this approach is that biometrics of users are neither stored nor shared in the whole system. Two users can communicate with each other without the worry of storing or sharing their biometric data. Also, there is no information in the system that adversaries can get their hands on that can reveal the biometric information of users in any way. Let's discuss this below in detail.

Starting with the feature extraction of biometric data of users, feature vector extracted from the biometrics is permuted to get the private key of the user. This private key is fed to DH algorithm to generate public key which is shared then. Public key of other user is combined with private key to generate a symmetric key. Now, in this whole process of generating this symmetric cryptographic key, no intermediate transformation is stored anywhere. So, no adversary can get any information related to biometric from this whole process. We consider public key and cryptographic key generated from DH algorithm to be non-invertible which ensures the security of biometric data of users. Security of DH algorithm is discussed in later subsections. This ensures that there is no such information that someone can get their hands on that can reveal the biometrics of users.

5.2 Security of Diffie-Hellman algorithm

The security of Diffie-Hellman key exchange lies in the fact that, while it is relatively easy to calculate exponentials modulo a prime, it is very difficult to calculate discrete logarithms. For larger primes, the latter task is considered infeasible [1]. Lets understand this with an example. DH algorithm requires two parameters q and α, let prime number q be 353 and its primitive root α be 3. A and B select private keys $X_{\mathrm{A}} = 97 and X_{\mathrm{B}} = 233$ respectively. Now, the public keys

$$Y_{\mathrm{A}} = 3^{97} mod 353 = 40$$

$$Y_{\mathrm{B}} = 3^{233} mod 353 = 248$$

After exchanging public keys, common secret key is

$$K = (Y_{\mathrm{A}})^{X_{\mathrm{B}}} mod 353 = 248^{97} mod 353 = 160$$

$$K = (Y_{\mathrm{B}})^{X_{\mathrm{A}}} mod 353 = 40^{233} mod 353 = 160$$

Assuming that an attacker has q, α, Y_A and Y_B, to determine the secret key K he can use brute-force in this example. Attacker needs to solve the equation $3^a mod 353 = 40$ or $3^b mod 353 = 248$. But with larger numbers, this problem becomes impractical. Hence, even if an attacker gets access to the public keys, private keys cannot be generated from them. Same applies on the cryptographic keys, private keys cannot be generated from public and cryptographic keys. So, DH algorithm performs a non-invertible transformation on the private keys.

5.3 Security of cryptographic key

In this approach, cryptographic key is generated from biometric of sender and receiver using DH algorithm. This key is valid for only for a session and is destroyed as soon as the session is destroyed. This key is never shared so there is no way to get this key by attacking the network. There can arise different possible attacks on the proposed system. Next, the security of this system against such attacks is discussed in detail.

5.3.1 Network attack

In this attack, an attacker cracks the network security and can get all the information shared over the network. In this system, public keys of sender and receiver are the only information shared over the network before generating a secure session key. If an attacker gets public keys of both sender and receiver, no information can be reverse-engineered using these keys as discussed in section 5.2. So, this proposed system is secure against such attacks.

5.3.2 Attack on a host

This type of attack considers the possibility that an attacker attacks a user/host in the network and gets all the information available at the user end. In this system, a user stores transformation key, cryptographic session key, RSA public-private key pair and authentication certificate. With all this information, the attacker gets access to that particular session only, nothing else. An attacker can log all the encrypted transmissions and can get stored messages of that user, but the attacker still can't decrypt these messages. As cryptographic keys are changed in each session and are not related in any way except that they are generated from original biometric of user, an attacker can access messages of that session only. For decrypting messages of previous communications and to encrypt messages of future conversations, an attacker still needs original biometric of the user which we can assume will be secure with the user. This way, access to cryptographic key of a session gives access to messages of that session only, neither the previous nor the future communications. This property is called perfect forward secrecy which the proposed system achieves.

5.3.3 Replay attack

Use of session keys used in this system prevents the replay attacks. For each session between two users, a different cryptographic key is generated and destroyed as soon as the session is over. If an eavesdropper wants to make replay attack using a message previously transmitted by legal users, then it will fail because cryptographic key is changed and retransmitted message will make no sense to the user now. If an eavesdropper uses one of the public keys shared between two users and makes replay attack using this, cryptographic key cannot be generated because it

requires own private key along with public key to generate final cryptographic key. This way, the proposed approach resists replay attacks in the system.

5.3.4 Man in the middle attack

In this approach, before starting communication setup two users need to verify each other. This verification takes place using certificates provided by trusted certification authority (CA). Two users before setting up the communication verify each other by sharing their authentication certificates and checking each others identification. This verification before the communication setup makes sure that a user is communicating with the correct user, not a man-in-the-middle (MitM). At the time of verification, if a MitM eavesdrops two users certificate and sends his certificate to both of them to setup two communications, he will be verified as himself not the other user with which a user wishes to setup the communication. This way, a MitM attack will be revealed. Even for this, a MitM needs to have a authentication certificate provided by the CA which can only be provided to him after verifying his identity. So, no unknown user can get the certificate without identifying himself, once identified he cannot make MitM as his identity will be revealed in any such attack. Use of CA in the proposed approach to verify users using certificates prevents the system from MitM attacks.

Chapter 6

Conclusions

Cryptographic key generation and subsequently its maintenance are the two important issues in traditional cryptography. This work addresses these issues and provides a novel approach to generate symmetric cryptographic keys using fingerprint biometrics of sender and receiver.

Biometric data of users is neither stored nor shared at any point in the communication which provides privacy and security to biometric data. Revocability is provided to the irrevocable biometric data using transformation keys. Also, this work ensures long term security of messages shared among users by achieving perfect forward secrecy. Various attacks like replay attack, man-in-the-middle attack etc. are resisted by the proposed crypto-biometric system. Thus, this proposed approach provides an effective solution to the need of session-based secure communication setup for transmitting messages over an insecure communication channel.

There is a scope of optimizing this approach in future. For generating cryptographic keys from biometric data, accuracy is vital. For this, invariant features are extracted from biometrics of users and further used to generate keys. For multiple instances of a biometric of a single user, feature extraction technique should give the exact same output. To achieve an accuracy of 100% is a very tough job but there is scope of increasing accuracy in our approach. Future work can be done on this part to achieve as much accuracy as we can get in order to get accurate cryptographic key from biometric data of users.

Bibliography

[1] William Stallings: *Cryptography and Network Security : Principles and Practice*, Fifth Edition (Prentice Hall 2011).

[2] U Uludag, S Pankanti, S Prabhakar, AK Jain: *Biometric Cryptosystems: Issues and Challenges*, Proc. IEEE. 92(6), 948–960 (2004).

[3] Christian Rathgeb and Andreas Uhl: *A survey on biometric cryptosystems and cancellable biometrics*, EURASIP Journal on Information Security 2011

[4] Daniel Peralta , Mikel Galar , Isaac Triguero , Daniel Paternain , Salvador García , Edurne Barrenechea , José M. Benítez , Humberto Bustince , Francisco Herrera: *A survey on fingerprint minutiae-based local matching for verification and identification*, Information Sciences: an International Journal, v.315 n.C, p.67-87, September 2015

[5] NK Ratha, S Chikkerur, JH Connell, RM Bolle: *Generating Cancellable Fingerprint Templates*, IEEE Trans. Pattern Anal. Mach. Intell. 29(4), 561–572 (2007)

[6] 6. S Kanade, D Camara, E Krichen, D Petrovska-Delacretaz, B Dorizzi, Evry F, in Proceedings of 6th Biometrics Symposium (BSYM 2008). *Three Factor Scheme for Biometric-Based Cryptographic Key Regeneration Using Iris* (Tampa, Florida, USA, 2008), pp. 59–64

[7] 7. S Kanade, D Petrovska-Delacr~etaz, B Dorizzi, in Proceedings of Fourth IEEE International Conference on Biometrics: Theory Applications andSystems (BTAS), Washington, DC, USA, 2010. *Generating and sharing biometrics based session keys for secure cryptographic applications*, 2010), pp. 1–7

[8] 8. F Monrose, MK Reiter, Q Li, S Wetzel, in Proceedings of IEEE Symposium on Security and Privacy.*Cryptographic key generation from voice* (IEEE Computer Society Washington, DC USA, 2001), pp. 202–213

[9] H Feng, CC Wah: *Private key generation from on-line handwritten signatures*. Inform Manag. Comput. Secur. 10(4), 159–164 (2002)

[10] B Chen, V Chandran, in Proceedings of 9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications. *Biometric Based Cryptographic Key Generation from Faces* (Glenelg Australia, 2007), pp. 394–401

[11] C Rathgeb, A Uhl.: *Context-based biometric key generation for Iris*. IET Comput. Vis. 5(6), 389–397 (2011)

[12] C Yao-Jen,WZhang, T Chen, in Proceedings of IEEE International Conference on Multimedia and Expo (ICME'04), Taipei, 2004, Vol. 3. *Biometrics-based cryptographic key generation* (IEEE, 2004), pp. 2203– 2206

[13] S Dutta, A Kar, BN Chatterji, NC Mahanti, in Proc. Adv. Concepts Intell. Vis. Syst., LNCS 5259. *Network Security Using Biometric And Cryptography* (Springer Berlin Heidelberg, 2008), pp. 38–44

[14] A Jagadeesan, K Duraiswamy, Secured Cryptographic Key Generation from Multimodal Biometrics: *Feature Level Fusion of Fingerprint and Iris*. Int. J. Comput. Sci. Inform. Secur. 7(2), 28–37 (2010)

[15] A Jagadeesan, T Thillaikkarasi, K Duraiswamy: *Cryptographic Key Generation from Multiple Biometrics Modalities: Fusing Minutiae with Iris Feature*. Int. J. Comput. Appl. 2(6), 16–26 (2010)

[16] Sharda Singh and Dr. J. A. Laxminarayana: *RSA Key Generation Using Combination of Fingerprints*, National Conference on Advances in Engineering, Technology & Management, 2015

[17] Subhas Barman, Debasis Samanta and Samiran Chattopadhyay: *Fingerprint-based crypto-biometric system for network security*, EURASIP Journal on Information Security, 2015

[18] Sanjay G. Kanade, Dijana Petrovska-Delacretaz and Bernadette Dorizzi: *A Novel Crypto-Biometric Scheme for Establishing Secure Communication Sessions Between Two Clients*, International Conference of the Biometrics Special Interest Group (BIOSIG), Gesellschaft fOr Informatik e.V., 2012

[19] K Nandakumar, A Jain, S Pankanti: *Fingerprint-based fuzzy vault: Implementation and performance*. IEEE Trans. Inf. Forensics Secur. 2(4), 744–757 (2007)

[20] A. K. Jain, A. Ross and U. Uludag: *Biometric Template Security: Challenges and Solutions*, Proceedings of European Signal Processing Conference (EUSIPCO), (Antalya, Turkey), September 2005

[21] Song Wang , Jiankun Hu: *Alignment-free cancelable fingerprint template design: A densely infinite-to-one mapping (DITOM) approach*, Pattern Recognition, v.45 n.12, p.4129-4137, December, 2012

[22] D. Maltoni, D. Maio, A.K. Jain, S. Prabhakar: *Handbook of Fingerprint Recognition, second ed.*, Springer Publishing Company, Incorporated, 2009.

[23] A. Bengueddoudj, S. Akrouf, F. Belhadj, D. Nada: *Improving fingerprint minutiae matching using local and global structures*, in: 8th International Workshop on Systems, Signal Processing and Their Applications, WoSSPA 2013, 2013, pp. 279–282.

[24] S. Bistarelli, F. Santini, A. Vaccarelli: *An asymmetric fingerprint matching algorithm for java card*, Pattern Anal. Appl. (2006) 359–376.

[25] X. Tan, B. Bhanu: *A robust two step approach for fingerprint identification*, Pattern Recogn. Lett. 24 (2003) 2127–2134.

[26] Y. Ying, H. Zhang, X. Yang: *A method based on delaunay triangulation for fingerprint matching*, in: SPIE Conference on Biometric Technology for Human Identification II, 2005.

[27] R. Cappelli, M. Ferrara, D. Maltoni: *Minutia cylinder-code: a new representation and matching technique for fingerprint recognition*, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2010) 2128–2141.

[28] M.H. Izadi, L. Mirmohamadsadeghi, A. Drygajlo: *Introduction of cylinder quality measure into minutia cylinder-code based fingerprint matching*, in: 2012 IEEE 5th International Conference on Biometrics: Theory, Applications and Systems, BTAS 2012, 2012, pp. 353–358.

[29] X. Chen, J. Tian, X. Yang: *A new algorithm for distorted fingerprints matching based on normalized fuzzy similarity measure*, IEEE Trans. Image Process. 15 (2006) 767–776.

[30] J. Bringer, V. Despiegel: *Binary feature vector fingerprint representation from minutiae vicinities*, in: IEEE 4th International Conference on Biometrics: Theory, Applications and Systems, BTAS 2010, 2010.

[31] G. Fang, S.N. Srihari, H. Srinivasan, P. Phatak: *Use of ridge points in partial fingerprint matching*, in: SPIE: Biometric Technology for Human Identification IV, 2007.

[32] A. Balti, M. Sayadi, F. Fnaiech: *Improved features for fingerprint identification*, in: Proceedings of the Mediterranean Electrotechnical Conference – MELECON, 2012, pp. 878–883.

[33] NK Ratha, JH Connell, R Bolle: *Enhancing Security and Privacy in Biometric-Based Authentication System*. IBM Syst. J. 40(3), 614–634 (2001)

[34] Z.Jin, A.Teoh, T.S.Ong, C.Tee: *Secure minutiae-based fingerprint templates using random triangle hashing*, in First International Visual Informatics Conference, 2009,pp.521–531.

[35] Z.Jin, A.Teoh, T.S.Ong, C.Tee: *A revocable fingerprint template for security and privacy preserving*, KSII Transactions on Internet and Information Systems 4(6)(2010)1327–1341.

[36] F.Farooq, R.Bolle, J.Tsai-Yang, N.Ratha: *Anonymous and revocable fingerprint recognition*, in IEEE Conference on Computer Vision and Pattern Recognition, 2007,pp.1–7.

[37] SVK Gaddam, M Lal: *Efficient Cancellable Biometric Key Generation Scheme for Cryptography*. Int. J. Netw. Secur. 11(2), 57–65 (2010)

[38] N Lalithamani, KP Soman, in The 2nd International Conference on Computer Science and Information Technology. *Towards Generating Irrevocable Key for Cryptography from Cancelable Fingerprints* (Beijing China, 2009), pp. 563–568

[39] Sayani Chandra, Sayan Paul, Bidyutmala Saha, Sourish Mitra: *Generate an Encryption Key by using Biometric Cryptosystems to secure transferring of Data over a Network*: IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 12, Issue 1 (May. - Jun. 2013), PP 16-22.

[40] Davida, G.I., Frankel, Y., and Matt, B.J.: *On enabling secure applications through off-line biometric identification*, in proceedings of the 1998 IEEE Symposium on Security and Privacy (May 1998), pp.148-157.

[41] S. Kanade, D. Petrovska-Delacretaz, and B. Dorizzi: *Multi-biometrics Based Crypto-biometric Session Key Generation and Sharing Protocol*, in ACM Workshop on Mumtimedia and Security (MM&Sec), 2011.

[42] Menzies, Alfred; van Oorscot, Paul C.; Vanstone, SCOTT (1997). *Handbook of Applied Cryptography*. CRC Pres. ISBN 0-8493-8523-7.

[43] Wu, Thomas (1997-11-11). *The Secure Remote Password Protocol*. Internet Society Symposium on Network and Distributed System Security.

[44] *Fingerprint Verification Competition FVC2002*, [Online]. Available: http://bias.csr.unibo.it/fvc2002

[45] *RFC 3526 "More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)"*, [Online]. Available: https://www.ietf.org/rfc/rfc3526

Appendices

Appendix A

Using same DH parameters for multiple sessions

There is no known bad side-effect to sharing the same DH parameters with other people / servers. There has been some theoretical concern about how an attacker might invest a lot of work into "breaking" discrete logarithm modulo a specific prime and then reusing the intermediate calculations to quickly break many DH instances which work modulo that specific prime. In that sense, reusing common DH parameters might imply an extra power of nuisance for the attacker. But, this kind of cost sharing does not work for all DL-breaking algorithms, and this also assumes that the attacker could break DL modulo the prime, i.e. the prime was too short or "special form". This would be a much more pressing reason not to use that prime: not that it is shared, but that it is weak.

Thus, using existing, shared DH parameters is not a problem as long as you can make sure that the said parameters have not been specially "cooked" to allow for a fast(er) break. This usually means that the generating algorithm has been fully specified and can be verified to have been faithfully followed.

Using custom DH parameters should work with all clients, because there is no implementation advantage to tying an implementation to a specific modulus (contrary to elliptic curves). As long as your modulus fits in the size requirements of a specific implementation, things ought to be fine (some older implementations have trouble with sizes beyond 1024 bits; other won't like a modulus whose size is not a multiple of 32 or 64 bits).

Appendix B

DH parameters

In this project, following parameters (RFC 3526 : 2048-bit MODP group) were used for implementing DH algorithm.

Prime number (for modulo) q:

FFFFFFFFFFFFFFFFC90FDAA22168C234C4C6628B80DC1CD1

29024E088A67CC74020BBEA63B139B22514A08798E3404DD

EF9519B3CD3A431B302B0A6DF25F14374FE1356D6D51C245

E485B576625E7EC6F44C42E9A637ED6B0BFF5CB6F406B7ED

EE386BFB5A899FA5AE9F24117C4B1FE649286651ECE45B3D

C2007CB8A163BF0598DA48361C55D39A69163FA8FD24CF5F

83655D23DCA3AD961C62F356208552BB9ED529077096966D

670C354E4ABC9804F1746C08CA18217C32905E462E36CE3B

E39E772C180E86039B2783A2EC07A28FB5C55DF06F4C52C9

DE2BCBF6955817183995497CEA956AE515D2261898FA0510

15728E5A8AACAA68FFFFFFFFFFFFFFFF

Primitive root (generator) α: 2