BIOMETRIC BASED CRYPTO-KEY GENERATION FROM FACES

[1] Indu Verma [2] Sanjay Kumar Jain
[1] Research Scholar , [2] Associate Professor
[1] School of Computer Applications
[1] ITM University, Gwalior , MP, India

***Abstract*:** Several biometrics like face, iris, retina, etc., are used in rendering security to the information or key. Instead of using PINs and passwords as Crypto-Keys that are either easy to forget or vulnerable to dictionary attacks, easy-to-carry and difficult-to-transfer keys can be generated based on user-specific biometric information. In this paper, a framework is proposed to generate stable Crypto-Keys from biometric data that is unstable in nature. The proposed framework differs from prior work in that user-dependent transforms are utilized to generate more compact and distinguishable features. Thereby, a longer and stable bit stream can be generated as the Crypto-Key. Experiments are performed on one face database to verify the feasibility of the proposed framework. The preliminary result is very encouraging.

The generation of Crypto-Key from biometrics is used generally to secure the system. This paper examines the possibility of using biometric attribute to overcome common problems in having a single biometric scheme for authentication. Here key generated by using Face biometric factors. Crypto-Key is an important entity in this process. In general, randomly generated Crypto-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 Crypto-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. Elements of combined template are shuffled using shuffle key and hash of the shuffled template generates a unique session key.In this paper, we first propose a simple and effective protocol to securely share such crypto-biometric keys. Moreover, we propose another protocol to generate and share session keys which are valid for only one communication session. This protocol achieves mutual authentication between the client and the server without the need of trusted third party certificates. This protocol also facilitates easy online updating of templates. For experimentation,We have tested our work using the face images from publicly available face databases, protocols are evaluated for biometric verification performance on a subset of the NIST-FRGCv2 face database.

Keywords – Biometrics, DNA, Face, Fingerprint, , Hand geometry, Iris, Retina, Crypto-Key

I.INTRODUCTION

Information security and a secure transmission of data become very important in information and communication technology. A third party can trap data or steal important data stored in a computer. To prevent this, it is advocated to encrypt the messages to provide information security. This type of protection is usually provided using cryptography. In cryptography, a key (K1) is used to encrypt a message (called plaintext P) with encryption algorithm (E) into ciphertext (C). The ciphertext is converted into plaintext using a key (K2) and decryption algorithm (D). Crypto-Key generation and subsequently its maintenance are the two important issues in traditional cryptography. A Crypto-Key should be generated in such a way that it is hard enough to guess and then it should be managed without any overhead of users. This work addresses these issues and propose a novel approach to generate random Crypto-Key using face biometric of sender and receiver.

This work aims to address the above-mentioned concerns and proposes a solution to develop a crypto- biometric system. Our proposed solution includes the following: 1) how to generate cancelable face tem- plate so that biometric features of neither communicators are never disclosed to anyone, 2) how to generate a unique Crypto-Key for encryption (decryption) of messages using the cancelable face templates of both sender and receiver, and 3) how to generate revocable session key from irrevocable biometric traits prior to each session. In this paper, we propose an approach to generate, share, and update Crypto-Key for symmetric cryptography from the faces of sender and receiver at their sites for encryption and decryption, respectively. Initially, sender shares two secret keys namely stego key (Kg) and shuffle key (Kshuf) with receiver. Stego key is generated from a password (pwd) by sender and receiver using pseudo random number generator (PRNG) . Shuffle key (Kshuf) is generated randomly, which is a binary stream of bits and stored in token. In this work, sender shares Kshuf and pwd with receiver using public key cryptography. With our proposed approach, asymmetric cryptography is proposed to exchange an initial shuffle key Kshuf and a password pwd between sender and receiver. For session keys, we propose biometric-based Crypto-Key generation to establish a link of users

biometric with Crypto-Key. In our approach, biometrics of both communicating parties are integrated to generate crypto- graphic keys so that we can avoid the complex random number generation and alleviate the issue of storing the random Crypto-Keys in the custody of sender and

receiver. Moreover, revocable key generation in every session and protecting the privacy of biometric templates are the challenge which has been addressed in this work. Both sender and receiver exchange their cancelable face template with each other 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. Finally, Crypto-Key is generated from this shuffled tem- plate using a hash function.

Issues

Crypto-biometric system, however, has some issues. Any biometric system needs to provide biometric tem- plate protection which confirms the privacy and security of biometric data . The biometric data used in a bio- metric system should not leak any information about the biometric features. It is also required to provide revocability to the irrevocable biometric data. In password-based authentication systems or token-based authentication systems, passwords or tokens are easy to change while it is compromised. But, biometric traits are inherent and fixed forever, that is, the biometric data is irrevocable . The owner of biometric traits is not able to revoke her bio- metric when it is compromised. As a result, the biometric data become useless forever . To overcome this problem, it demands a cancelable transformation of biometric template to provide revocability to the irrevocable biometric. Simultaneously, it would ensure the privacy of biometric data , so that the transformed template does not leak any information about the original template. Moreover, biometric data is required to be transmitted over non-secure communication channels for remote use. Therefore, there is a need to generate Crypto-Key, which is revocable and non-invertible from the biometrics of two different users without compromising the privacy and security of the biometrics involved in key generation process. In a real-time scenario, from a set of *B* biometric attributes, a person may not be able to produce a subset of attributes. *We will* need to deal *with* such situations and clearly define the acceptable level of identification. For e*x*ample, assume an application defines *B*={ *Faces ,iris, voice, password*}. *I*f on any one particular instance, an authentic person may not be able to produce all the attributes of *B* but rather a subset *S*={ *Faces, password* }.*T*his could be due to change in physical attributes of a person, or due to external influential factors. Correlating, *S with B* is a ma*j*or challenge. *I*ncomplete and erroneous input must be distinguished. A genuine person might furnish incomplete biometric data. *I*n such cases the system must decide if the identification process has sufficient information to authenticate a person. *T*his is extremely critical in those cases *w*hen hybrid biometric data is used for key generation that is used *w*ith standard crypto-algorithms.

II.BACKGROUND

To secure biometric templates many techniques are there. These techniques are categorized into two classes: Template Transformation. These techniques modify the biometric template with a user specific key so that it is complicated to recover the original template from the transformed template. Throughout authentication, the same transformation is applied to the biometric query and the matching is performed in the transformed domain to evade exposure of the original biometric template. Generally the secure template should satisfy the properties like:

(i) Non-invariability—specified a secure template, it must be computationally not easy to find a biometric feature set that will match with the particular template.

(ii) Revocability— specified two secure templates generated from the same biometric data, it must be computationally tough to identify that they are consequent from the same data or obtain the original biometric data.

Moreover, biometric systems possess problems of their own such as non-revocability, non-diversity, and possibility of privacy compromise which should be taken into consideration. Revocability is a desired property for a user verification system which implies that if the authenticator (e.g., password) is compromised, it can be replaced with a fresh one. The old authenticator can no longer be used in that system for authentication. Since biometric traits are permanently associated with the user, they cannot be replaced and thus lack the property of revocability. Additionally, the templates generated from the same biometric trait of a user stored in different biometric systems are similar, and can be cross linked together compromising user privacy.

Previous works related Literature survey

Our work consists of mainly three sub-tasks: i) transformation of biometric template, ii) secure transmission of biometric data, and iii) crypto-biometric system. Biometric systems require a transformation of biometric template to ensure privacy, security, and revocability of biometric data. The technique which can meet this requirement is called cancelable or revocable biometric. This privacy enhancement problem is identified, and conceptual frameworks of biometric templates are presented in formally defined the problem of cancelable biometric. Biometric data transmission. There are many work reported in the current literature where the biometric data is transmitted over communication channels for the purpose of remote authentication. Existing work consider hiding of biometric data within another

media called cover media using data hiding technique.

Our work is inspired from a number of previous works related to cancellable biometrics and the generation of Crypto-Key from cancellable biometric features. A brief review of some of the works is given below.

In order to solve these problems of non-revocability, non- diversity, and possibility of privacy compromise, a new re- search area, called cancelable biometric systems [2], [3], [4], [5], has emerged. Cancellable biometrics proffers a greater level of privacy by facilitating more that one template for the same biometric data and thus the non-linkability of user's data stored in diverse databases. The measurement of the success of a particular transformation and matching algorithm for faces. A key dependant geometric transform was employed on the features obtained from a face, so as to pro- duce a key-dependent cancellable template for the finger- print. Besides, they have also studied the performance of an authentication system that utilizes the cancellable face matching algorithm detection purposes. Experimental evaluation of the system was carried out and the results illustrated that it was possible to bring about a good performance when the matching algorithm remains unaltered. Unfortunately, when used for cryptographic purposes, both classical and cancelable biometric systems have one drawback in common: the verification result of these systems is a one-bit information which results in a weak link between biometrics and cryptography.

Biometrics based Crypto-Key generation and regeneration systems try to strengthen this link between biometrics and cryptography. In key generation systems, a stable bit- string called crypto-biometric key is extracted from biometric data. Some examples of key generation systems are [6], [7], [8].

A realistic and secure way to incorporate the iris biometric into cryptographic applications. They deliberated on the error patterns within iris codes and developed a two-layer error correction technique that merges Hadamard and Reed-Solomon codes. The key was produced from the iris image of the subject through the auxiliary error correction data that do not disclose the key and can be saved in a tamper-resistant token like a smart card. The evaluation of the methodology was performed with the aid of samples from 70 dif- ferent eyes, 10 samples being obtained from every eye. It was established that an error-free key can be reproduced reliably from genuine iris codes with a success rate of 99.5 percent. It is possible to produce up to 140 bits of bio- metric key, more than adequate for 128-bit AES. Since the crypto-biometric key is a multi-bit string, the entropy is higher than using the classical biometric verifica- tion system. Another strategy to obtain Crypto-Keys using biometrics is to bind a random key to the reference biometric data and then regenerate it with the help of another biometric sample. We denote this strategy as biometrics based Crypto-Key regeneration (also known as key binding) [9], [10], [11], [12], [13].

The keys derived using such crypto-biometric systems can be used in cryptography. Cryptographic systems are mainly of two types: symmetric-key cryptography in which the encryption and decryption keys are the same, and public-key (also called asymmetric) cryptography where the encryption and decryption keys are different but are mathematically related. Since all the entities participating in a cryptographically secure communication session must have correct keys, management of these keys is a critical issue. The crypto- biometric systems listed above do not mention any specific key management/sharing methodologies and rely on conventional cryptography for the purpose. The proposals which deal with biometrics based key sharing and authentication protocols are briefly described below.

Boyen et al. [14] proposed a biometrics based remote authentication protocol using the fuzzy extractor scheme. The one-time biometric authentication protocol of Ueshige and Sakurai [15] creates biometric authentication based secure sessions.

The application of handwritten signature to cryptography was analyze don basis of recent works displaying the likelihood of key generation by means of biometrics. A cryptographic construction called the fuzzy vault was employed in the signature-based key generation scheme. The analysis and evaluation of the usability of distinctive signature features appropriate for the fuzzy vault was carried out. Results of experimental evaluation were reported. The reports also included the error rates to release the secret data with the aid of both random and skilled forgeries from the MCYT database.

Similarly, a scheme for biometric based authentication in which the biometric comparison is carried out in encrypted domain.The fuzzy extractors along with public key cryptography for secure authentication.

The problem with these protocols is that they require storage of reference biometric templates. Additionally, the scheme in [17] requires a secure link to be established between the different components of the system. Moreover, the schemes in [15] and [16] can only verify the identity of the person; they cannot generate keys required for se- cure communication. Moreover, the keys obtained with the schemes in [14] and [17] are the same for all the sessions. Using the same key for encryption of a large amount of data can make some cryptanalytic attacks easier. Therefore, most of the practical systems, e.g., the Transport Layer Security (TLS) protocol [18], employs a session specific symmetric key for secure communication. In TLS, the session key is temporarily generated in every session and shared through public-key cryptography. The session key is a one-time key valid precisely for a single communication session.

The "Secure Ad-hoc Pairing with Biometrics: SAfE" protocol proposed [19] can be used to establish a secure link between two parties. Keys are obtained from biometrics with the help of the fuzzy extractor scheme. The drawback of this protocol is that it shares the biometric data between the two parties and requires mutual trust among them. Moreover, it also requires a secure channel for exchanging the biometric data. Following the concept of session keys, Scheirer and Boult [20], [21] proposed "bipartite biotokens".

An on-line signature-based biometric authentication system, where non invertible transformations were applied to the acquired signature functions ruling out the possibility to derive the original biometrics from the stored templates at the same time maintaining the same recognition performances of an unprotected system [21]. Precisely the probability of producing cancellable templates from the same original data, thereby proffering an appropriate solution to privacy concerns and security problems was intensely explored.

They combined their earlier proposal of revocable biotokens with fuzzy vault which enables to securely share keys using biometrics. In this scheme, a series of transformations is shared between the client and the server. A new transformation (in succession) is applied in every communication session. The bipartite biotokens are session specific and make it possible to share session specific data between two parties. In this paper, we propose two novel protocols: the first protocol enables secure sharing of keys generated using the crypto-biometric systems. The second proposal is for generating and sharing Crypto-Keys which are valid precisely for one session. This protocol: (1) facilitates se- cure generation and sharing of session keys, (2) possesses cancelability/revocability and privacy protection because the templates stored in the database are revocable, (3) achieves mutual authentication: the server can authenticate the client and the client can also authenticate the server without the need of trusted third party certificates, and (4) can carry out secure online update of templates. The difference between our proposal and the scheme in [21] is in the key regeneration approach. The key regeneration system used in our protocol is a hybrid system combining a shuffling based cancelable biometric system with fuzzy commitment scheme whereas [21] uses fuzzy vault scheme. Two-factor cancellable formulation that facilitates data distortion in a revocable yet non-reversible manner by first converting the raw biometric data into a fixed-length feature vector followed by the projection of the feature vector onto a sequence of random subspaces that were obtained from a user-specific Pseudorandom Number (PRN). The process was revocable making the replacement of biometrics seem as easy as replacing PRNs. This formulation was confirmed under numerous scenarios (normal, stolen PRN, and compromised biometrics scenarios) with the aid of 2400 Facial Recognition Technology face images. Moreover, our scheme does not need any public key to initiate the key sharing process. The biometric verification performance of the proposed system is evaluated on a subset of the NIST- FRGCv2 face image database [22].

The rest of this paper is organized as follows: the biometrics based key regeneration system is summarized and then the proposed key sharing protocol is described in Section II. The novel protocol for session key generation and sharing is described in Section III. Experimental evaluation related to biometric performance along with some security analysis is given in Section IV. Finally, the Section V sets out our conclusions and perspectives.

Performance Metrics for Biometric Systems

The different performance metrics for evaluating the biometric system are as follows

False Acceptance Rate (FAR) : The FAR is defined as the probability that a user making a false claim about his/her identity will be verified as that false identity. The importance of the FAR is the strength of the matching algorithm. The stronger the algorithm, the less likely that a false authentication will happen.

FRR (False Rejection Rate): The FRR is defined as the probability that a user making a true claim about his/her identity will be rejected as him/herself. The strength of the FRR is the robustness of the algorithm. The more accurate the matching algorithm, the less likely a false rejection will happen.

Crossover Error Rate (CER): The rate at which both the accept and reject errors are equal. A lower value for the CER is desired for a biometric system in order to be considered more accurate as well as convenient for its users.

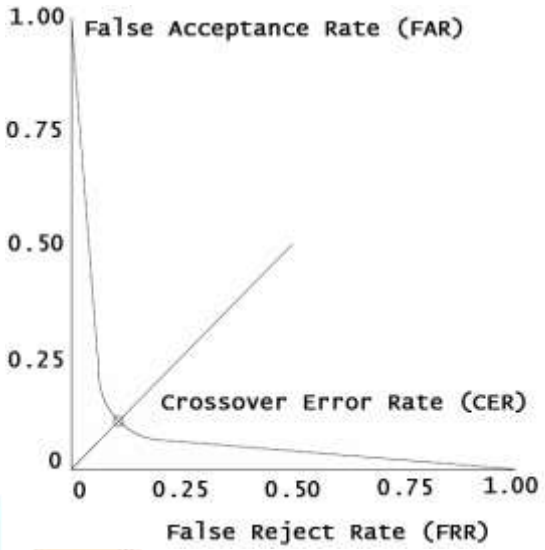
Failure to Enroll Rate (FER): The rate at which attempts to create a template from an input is not successful. This is most commonly caused by low quality inputs that are insufficiently distinctive biometric samples or from a system design that makes it difficult to provide consistent biometric data.

Failure to Capture Rate (FCR): Applicable for automated systems, the probability that the system fails to detect a biometric input when presented correctly.

Template Capacity: The number of unique users that can be represented by its contents

Tradeoff: Larger the FER, lower the FAR and FRR; and vice-versa.

The graphical representation of relationship between FRR, FAR and CER is shown in Fig.2



Biometric System Evaluation

The biometric system can be evaluated with the help of these factors

Universality: Can everyone provide the considered biometric.

Uniqueness:How well the biometric separates individually from another.

Permanence :Stability along life time.

Collectability :Ease of capture for measurement.

Performance :Accuracy, speed, and robustness of technology used.

Acceptability :Degree of approval of a technology by end user.

Circumvention :How hard to fool the system.

The various fusion levels for a multi-biometric system is summarized in this section. They are as follows

In feature level fusion new feature vector is constructed with high dimensionality. The newly formed vector is more discriminative than individuals.

Score Level Fusion

In this level matching scores are collected from every individual and then combine together.

Decision Level Fusion

In decision level fusion final results are combined together. We use feature level fusion for multibiometric cryptosystem. Unlike passwords and tokens, compromised multibiometric templates are not recoverable. Because of this, multibiometric template security is very necessary thing. In this paper, we propose a scheme to protect all the templates of user in multibiometric system.

Levels of Fusion in Multi-Biometric Systems

III. PROPOSED APPROACH

Encryption using Chaotic Map

In this paper we use an algorithmm using chaotic map. There are two steps for [2] encryption, in first step we introduces a chaotic map using henon map.

$a(x+1)=1-ka(x)^2+g(x)$ $g(x+1)=sa(x)$

where k=1.4 s=0.3 to illustrate chaotic manner.

Minutiae Points Extraction

Matrix and Key Generation from Minutiae Points

The key generation algorithm is as follows: Assumptions

Mp – Minutiae point set Sp – Size of Mp

KL-Key Length Kv – Key Vector

Lk-Length of Key Vector

Z□ (X, Y) - Co-ordinate of a minutiae point.

Step 1: Read the Minutiae Points

Step 2: Fin the point H with highest X+Y.

Step 3: Draw a line from origin (0, 0) to the H and call it as L.

Step 4: Sort the Minutiae points and store in an array A.

Step 5: Value= KL/Np

Vector = KL%Np

Step 6: For i=1 to value

For j=1 to Sp

Read point X from Array A and Check the point whether it is above or below the line L.

If it is above the line or on the line put value as '0' else value is '1'. Store them in an array K.

Final Key: - Append the key vector of length vector to value of K.

IV.KEY GENERATION METHOD

Cryptographic systems require a secret key or a random number which must be tied to an individual through an identifier. This identifier indeed could be a globally unique user id or biometric data. Generating user ID-based key or random number is straightforward and the techniques could easily be found in literature.[32] But generating user-based Crypto-Keys includes several of approaches.

A. User Dependant Key Generation

PRNG (pseudo random number generator). The resulting pseudorandom number can be used directly as a key or adjusted with user-dependent data. User-dependent key may consist of user ID or biometric data. In order to make the key depends on a specific user, two ways could be applied. First the key generation algorithm could be modified by using the user dependent data. Second PRNG could be modified. PRNG Modification is accomplished using a front-end or back- end approach. In front-end manner, the definition of the seed value (which is used to create a random key) is extended to include a user-specific data component. In back-end manner, pseudorandom numbers are treated as intermediate values and processed further.

In this section we will describe three methods where user-specific data is biometric data. Biometric template of user is denoted by T.

1)Method 1: This method is based on pairing the biometric data with random numbers. The seed value of PRNG consists of a secret random value R and T, seed=(R, T). In order to eliminate any structure in the seed a complex function f is applied. Then the seed value is defined as seed=f(R, T) where f is the one-to-one mixing function. By the way, created pseudorandom numbers are not adversely affected by the composition of the seed value.

2)Method 2: In this method R and T are inputs to a more complex function that generates an n-bit pseudorandom number S which could be used directly as a key or as an input to key generation algorithm.

The algorithm is as follows:

•Generate a secret pseudorandom number R by using PRNG.

•Let Z=H(R,T) || H(R+1,T) || H(R+2,T) || ... ||

H(R+a ,T) where a=[n/h]-1. Here H is a strong collision-resistance one way hash function (such as SHA-1). H generates an h bit output from any length input. The symbol —||‖ denotes the concatenation operation.
•Let S be n specific (eg, leftmost) bits of Z.
Since H is a strong collision-resistance one-way hash function it is not feasible to derive either R or T from Z. This increases the security of the scheme. In practice this method is designed for the user to store the value of R and generate S from R and T on demand. S might be an encryption key. In this case, R might be encrypted and stored within a cryptographic subsystem.
3)Method 3:
In this method R and T are combined via simple function (XOR) to generate an n-bit secret pseudorandom number S. The algorithm is as follows:
• Let Z=H(R,T) || H(R+1,T) || H(R+2,T) || ... ||
H(R+a,T) where a=[n/h]-1.
•Let X be n specific bits of Z.
•Let R be an n-bit secret pseudorandom number, where R is either specified by the system or generated in his step using a PRNG.
•S=R (XOR) X.
4)Method 4: Whenever the user needs to encrypt or decrypt with S, T must
As can be seen, due to the hash function collision probability the previous three methods do not guarantee that a key or random number derived for a user will be unique. The probability of two users ending up with the same pseudorandom number is still present and will be quite small if n and h are chosen to be large. In this method, the user can prove or cannot deny that a key is one belonging to, or generated in, his/her designated space of keys or random numbers. In this method we assume that the value to be generated is n-bit long where (n > t). The algorithm is a two step process:
•Divide the space 2n into 2t subspaces. Note that each subspace correspond to a particular individual based the specific biometric data.
•Choose n-bit value at random from the user‘s subspace. The first step of the algorithm is realized by taking the first t bits from the biometric data representation and allow the remaining n - t bits to take any value. It would be advantageous to employ a mixing function to mix the user-dependent key or random number so that the secret entropy in it will be uniformly distributed over the entire key or random number.

V. Overview of Biometrics Based Key Regeneration Scheme

The biometrics based key regeneration scheme described in Fig. 1 is a hybrid system that combines a transformation based cancelable biometric system with fuzzy commitment based key regeneration scheme. It was proposed in our earlier work on iris based key regeneration [13]. In this scheme, a key $\mathbf{K}r$ is randomly generated and then encoded into a pseudo code θps using Error Correcting Codes (ECC). A cancelable transformation is applied on the reference biometric data θref of the user. This transformed data $\theta canc$ is then XOR*ed* with the pseudo code θps to obtain a locked code template $\theta lock$. At the time of key regeneration, a similar transformation is applied on the test biometric data $\theta test$ and then the cancelable data $\theta^{r}_{c\,anc}$ is XOR*ed* with the stored template $\theta lock$ to obtain $\theta^{r}_{p\,s}$. The two XOR operations transfer the errors between the reference and test biometric data onto the pseudo code ($\theta^{r}_{p\,s} = \theta lock \quad \theta^{r}_{c\,anc} = \theta ps\ \theta canc \quad \theta^{r}_{c\,anc} = \theta ps \quad e$). If the amount of errors *e* is less than the error correction capacity of the ECC, all these errors can be corrected after decoding. On successful error correction, a trial value of the random key $\mathbf{K}r$, denoted as $\mathbf{K}'_{r}$ is obtained. A comparison of the hash values of these two keys is carried out, and if they are the same, verification success is declared along with releasing the key. If the hash values are different, verification failure is declared.

In this Biometrics based session-key generation and sharing protocol the enrollment is securely carried out off-line and cancelable template is generated using the biometric data of the user and it is stored in the database at the server. The cancelable template from biometric data is θcanc The shuffling of the enrollment biometric data is done by θref ⊕ + shuffling key Ksh. The shuffling key Ksh is either stored on a smart card or can be generated from a password.

The algorithm for generating session key is as follows

When a client desires to securely communicate with the server, following steps are carried out: The client sends authentication request to the server.

The server sends acknowledgement to the client. Shuffled test biometric data θ′canc by client

1.Fresh biometric data θtest of the user is captured .

2.Shuffled using the shuffling key Ksh

3.θ′canc = θtest⊕ Ksh

User ID of the user is sent to the server not the biometric Locked code θlock is created by server

1.The server generates a random key Kr

2.Stored cancelable template θcanc.

3.θlock = E (Kr, θref) where E( ) indicates the encoding function.

The locked code θlock and H(H(Kr)) is sent to the client.

1.The client regenerates a trial value of the random key K′r

K′r = E-1 (θ′canc , θlock)

2.K′r is made as H(H(K′r))

If H(H(Kr)) = H(H(K′r)) – Server Authentic

then

H(K′r) is sent to server Server compares H(K′r) = H(Kr), to check the authenticity of the client.

If H(K′r) = H (Kr) – user Authentic - both parties - same key Kr.

Server sends the signal to start secure communication using the key Kr.

The Client & server share the same key which is a concept of symmetric key cryptography. The key is temporary and it is destroyed at the end of the communication session. Next communication session, a new key Kr will be randomly generated. The data being transferred through the channel during the protocol are Request, user ID, locked code θlock, hash values H(H(Kr)) and H(K′r) . None of the data reveal the biometric information.

The cancelable transformation used in our system is a shuffling scheme [13]. A randomly generated shuffling key $\mathbf{K}_{sh}$ is assigned to each identity and this key is used to randomize the biometric data of that user. This shuffling key is different for different users and is also different for

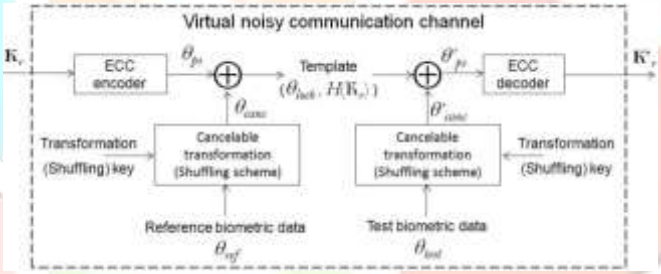


Fig. 1. Biometrics Based Key Regeneration Scheme

different applications. The biometric data is divided into blocks and these blocks are rearranged according to the shuffling key. Since the shuffling key is long, it needs to be stored on a smart card or should be generated using a password. The advantage of this shuffling scheme is that it increases only the impostor Hamming distances leaving the genuine Hamming distances intact. Hence, in addition to the properties of cancelability, revocability, template diversity, and privacy protection, the shuffling scheme also improves the verification performance of the system. The privacy protection provided by this system includes privacy of the user identity, privacy of user's biometric data, and privacy of the information stored in the system.

A. Secure Key Sharing Protocol

The crypto-biometric system described in the previous subsection (and all others summarized in Section I) fo- cuses on the problem of obtaining Crypto-Keys with the help of biometrics but it does not discuss about the usage of these keys. Basically, cryptographically secure communication requires that all the entities participating in communication should have the correct keys for encryption and/or decryption. This problem is addressed by key-sharing. The system described in Section II-A relies on conventional cryptographic techniques for key sharing. In this section, we propose a simple and effective protocol to securely share the crypto-biometric keys that can be obtained using this system.

We make following assumptions for the protocols:

- There is no trust between the client and the server. Therefore, the client will not share the authenticators (e.g., biometric data, passwords, etc.) with the server. The server will also not share the stored information with the client.

The communication link between the client and the server is unprotected. Therefore, the data being transferred through this should not leak information.

-
-
-

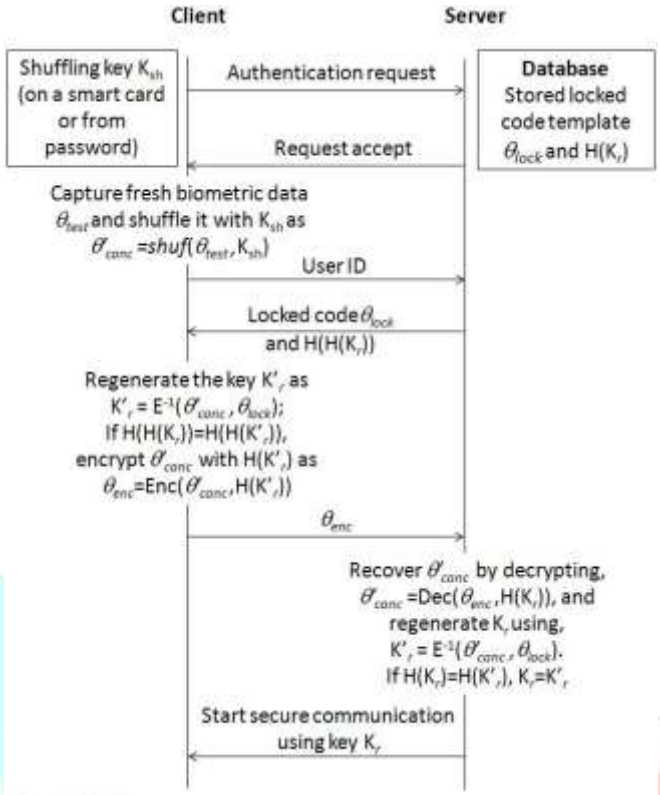


Fig. 2. The proposed protocol for biometrics based secure key sharing.

- Biometric data of the user should not be stored in the server or database to protect the user's privacy. The stored data should be revocable.
- The protocol should achieve mutual authentication be- tween the client and the server because none of them trust each other.

A schematic diagram of the proposed protocol for crypto- biometric key sharing is shown in Fig. 2. The enrollment process (not shown in the figure) is carried out off-line at a secure location. It is basically the same as described in Section II-A. A secure, locked code template θ_{lock} is created using a random key $\mathbf{K}r$, shuffling key $\mathbf{K}_{sh}$, and the reference biometric data θ_{ref}. This θ_{lock} along with the hash of the key $\mathbf{K}r$, i.e., $H(\mathbf{K}r)$, is stored in a database. The system can also employ a smart card to store the shuffling key $\mathbf{K}_{sh}$ in encrypted form. Otherwise, the shuffling key can be directly generated from a password.

At a later time, when the client needs a secure crypto- graphic key for communication, following steps are carried out:

1) The client sends the authentication request to the server.
2) The server responds with the request accept signal.
3) At the client side, fresh biometric data θ_{test} of the user is captured and shuffled using the shuffling key $\mathbf{K}_{sh}$ to obtain shuffled test code $\theta^r_{c\,anc}$. Only the user ID is sent to the server.
4) The server sends the locked code θ_{lock} along with the hash value $H(H(\mathbf{K}r))$ of the stored hash (i.e., hash of $H(\mathbf{K}r)$) of the user corresponding to the requested ID to the client.
5) At the client side, a key $\mathbf{K}^r_r$ is obtained from θ_{lock} and $\theta^r_{c\,anc}$ as, $\mathbf{K}^r_r = E^{-1}(\theta^r_{c\,anc}, \theta_{lock})$ where $E^{-1}()$ indicates the decoding function.
6) The client computes $H(H(\mathbf{K}^r_r))$ and compares it with the received $H(H(\mathbf{K}r))$ and if the two values are equal, the shuffled biometric data $\theta^r_{c\,anc}$ is encrypted using $H(\mathbf{K}^r_r)$ and the encrypted data is sent to the server.
7) The server decrypts the received data with $H(\mathbf{K}r)$ (which is stored in the database) to obtain $\theta^r_{c\,anc}$ and then regenerates the key $\mathbf{K}^r_r$ from θ_{lock} and $\theta^r_{c\,anc}$.
8) The server checks the hash values of the original and regenerated keys ($H(\mathbf{K}r)$ and $H(\mathbf{K}^r_r)$, respectively). If they are equal, it sends a start communication signal to the client.

Thus a secure channel is established between the client and the server through which secure communication can be carried out. Moreover, the protocol achieves biometric based secure authentication over an unsecured channel. The templates stored in the database are cancelable and the system possesses the properties of revocability, template diversity, and privacy protection.

VI. BIOMETRICS BASED SESSION-KEY GENERATION AND SHARING PROTOCOL

A. Session Key Generation and Sharing

The protocol described in the previous section is for sharing crypto-biometric keys which can be used in sym- metric cryptographic systems. A limitation of symmetric cryptographic systems is that if a large amount of data encrypted using a single key is available to an attacker, cryptanalytic attacks are made easier.

Public-key cryptographic systems use different (but mathematically related) keys for encryption and decryption which does not require secure key sharing. But, such systems are too slow for general purpose use (e.g., when large amount of data needs to be secured).

In order to overcome these shortcomings, many practical systems, such as the Transport Layer Security (TLS) [18] protocol[1], combine symmetric-key cryptography with public- key cryptography. In TLS, public-key cryptographic systems are employed to share a session key, and this session key is used in a symmetric cryptographic system during that communication session.

In this section, we propose a novel protocol to generate and share session keys based on biometrics. It makes use of the biometrics based key regeneration system described in Section II-A, but it can be generalized to accommodate any other key regeneration scheme. The enrollment is securely carried out off-line during which a cancelable template is generated from the enrollment biometric data of the user and is stored in the database at the server. In our case, the cancelable template is the shuffled biometric data θ_{canc} which is obtained by shuffling the enrollment biometric data θ_{ref} with a shuffling key $\mathbf{K}_{sh}$. The shuffling key $\mathbf{K}_{sh}$ is either stored on a smart card or can be generated from a password.

Figure 3 shows a schematic diagram of the proposed session key generation and sharing protocol. The channel between the client and the server is not secure, and hence, no private or sensitive information should be sent over the

[1]TLS is a widely used protocol, e.g., HTTPS (HyperText Transmission Protocol–Secure) uses TLS to secure World Wide Web traffic carried by HTTP. HTTPS is used for secure e-commerce applications such as online payments through internet, online banking applications, etc.

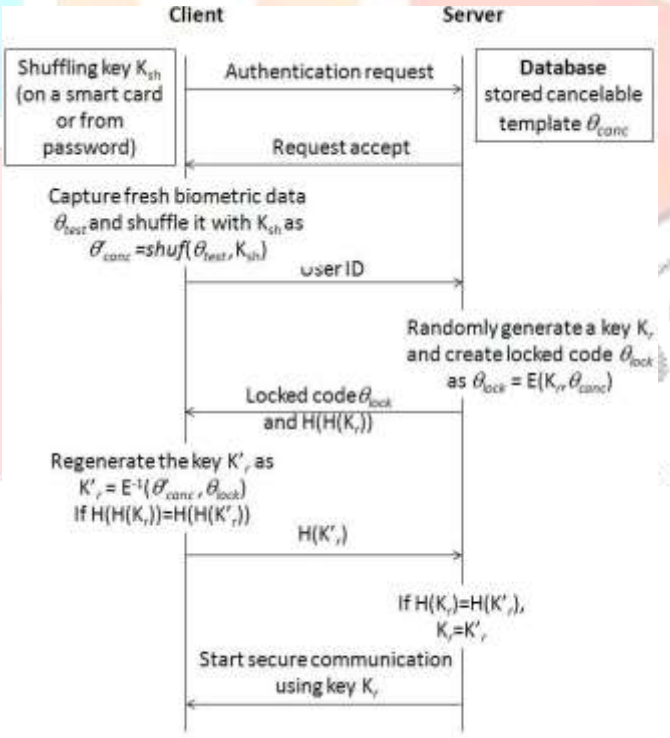


Fig. 3. The proposed protocol for generating and sharing biometrics based session keys.

network unless the channel is secured. When a client desires to securely communicate with the server, following steps are carried out:

1) The client sends authentication request to the server.
2) The server sends acknowledgement to the client.
3) Fresh biometric data θ_{test} of the user is captured and shuffled using the shuffling key $\mathbf{K}_{sh}$ to obtain shuffled test biometric data $\theta^{r}_{c\,anc}$ at the client side.
4) User ID of the user is sent to the server. Note that the biometric data is not sent to the server.

5) The server generates a random key $\mathbf{K}_r$ and a locked code θ_{lock} is created from $\mathbf{K}_r$ and the stored cancelable template θ_{canc}. This process of obtaining θ_{lock} is the same as shown in Fig. 1. It can be summarized as $\theta_{lock} = E(\mathbf{K}_r, \theta_{canc})$, where $E(\,)$ indicates the encoding function.
6) The locked code θ_{lock} is sent to the client. A double hashed version of the random key, i.e., $H(H(\mathbf{K}_r))$, is also sent to the client.
7) The client regenerates a trial value $\mathbf{K}_r^r$ of the random key using the locked code θ_{lock}, and the shuffled test biometric data $\theta_{c\ anc}^r$. This can be summarized as $\mathbf{K}_r^r = E^{-1}(\theta_{c\ anc}^r, \theta_{lock})$, where $E^{-1}(\,)$ indicates the decoding function. The regenerated key $\mathbf{K}_r^r$ is hashed twice to obtain $H(\mathbf{K}_r^r)$ and $H(H(\mathbf{K}_r^r))$.
8) The client compares $H(H(\mathbf{K}_r^r))$ with the received $H(H(\mathbf{K}_r))$, and if the two values are equal (which also confirms the server's authenticity), it sends the $H(\mathbf{K}_r^r)$ to the server. Server compares the received hash value $H(\mathbf{K}_r^r)$ with the hash value of the random key $\mathbf{K}_r$, i.e., with $H(\mathbf{K}_r)$, to check the authenticity of the user. If the two hash values are the same, it means that the user is authentic and has correctly received the randomly generated key $\mathbf{K}_r$. Thus, both the parties have the same key $\mathbf{K}_r$.
9) The key $\mathbf{K}_r$ is then treated as a session key and the server sends the signal to start secure communication using the key $\mathbf{K}_r$.

Thus, at the end of this protocol, the client as well as the server share the same key which can be used for symmetric- key cryptography. Note that, the key is temporary and is destroyed at the end of the communication session. In the next communication session, a new key $\mathbf{K}_r$ will be randomly generated and shared to be used as a session key.

The data being transferred through the channel during the protocol are request, user ID, locked code θ_{lock}, and the hash values $H(H(\mathbf{K}_r))$ and $H(\mathbf{K}_r^r)$, none of which reveal the biometric information. Moreover, the template stored in the database is cancelable which itself prevents cross-linking between biometric databases and protects user privacy.

As opposed to the popular and widely used cryptographic protocols such as HTTPS and TLS, the proposed protocol does not need a third party trusted certification authority. In HTTPS, the third party certification is used to confirm the server authenticity by using digital certificates. In our proposed protocol, client can confirm the authenticity of the server by comparing the double hashed values $H(H(\mathbf{K}_r^r))$ and $H(H(\mathbf{K}_r))$. This comparison can yield positive result if only if the server has generated a locked code θ_{lock} from the stored template θ_{canc} of the same user. On the other hand, the server authenticates the client by comparing the hash values $H(\mathbf{K}_r)$ and $H(\mathbf{K}_r^r)$. Thus, our protocol achieves mutual authentication without the need of third party certificates. The system described here employs strong authentication by combining biometrics with password (or smart card). Since the user is required to provide specific information in addition to biometric data, the system can resist replay attacks.

In this protocol, the randomly generated key $\mathbf{K}_r$ is en- coded using Error Correcting Codes (ECC). The ECC are required in order to cope with the biometric data variability. Note that, the error correction coding is applied at the time of authentication. Therefore, it is possible to accommodate different error correcting codes (compatible with the biomet- ric data) in the protocol. As it is done in the TLS, the client and server can negotiate on the choice of ECC and the error correction capacity to be used during authentication.

B. Online Template Update

Many systems (such as online banking services) require that the user authentication credentials be updated periodi- cally. In password based systems, this means that the user is asked to change his password periodically. On the other hand, the user may also wish to change his credentials.

The distributed nature of our proposed protocol allows the user and/or the system to update the template online. The

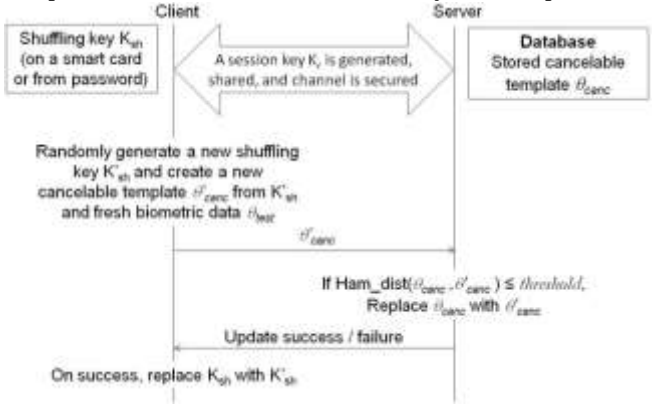


Fig. 4. Protocol showing online template update.

In the beginning of this protocol, the mutual authentication between the client and the server is carried out with the protocol shown in Fig. 3. Ham dist means Hamming distance.

template update procedure involves changing the cancelable template θ_{canc} by changing the reference biometric data θ_{ref} and the shuffling key $\mathbf{K}_{sh}$. The procedure for this template update is shown in Fig. 4.

The steps followed during the template update procedure are:

1) A secure communication channel is created between the client and the server by using the session key gen- eration and sharing protocol described in the previous subsection (shown in Fig. 3).
2) A new shuffling key $\mathbf{K}^r_{sh}$ is randomly generated at the client side and a cancelable template $\theta^r_{c\ anc}$ is obtained from the fresh test biometric data θ_{test} and $\mathbf{K}^r_{sh}$.
3) The new cancelable template $\theta^r_{c\ anc}$ is sent to the server through the encrypted channel.
4) The server compares the old template stored in the database θ_{canc} with the received cancelable template $\theta^r_{c\ anc}$. If the Hamming distance between the two is less than a threshold, the old template θ_{canc} is replaced with the new one $\theta^r_{c\ anc}$. Update success/failure mes- sage is sent to the client.
5) If the received message is success, the old shuffling key $\mathbf{K}_{sh}$ stored on the smart card is replaced with the new one $\mathbf{K}^r_{sh}$.

Note that, the template update process can be initialized by either the client or the server. Also the mutual authentication between client and server is carried out before initiating the template update procedure during session key generation and sharing. Additionally, the new cancelable data sent by the client is compared with the stored cancelable template before replacing. Therefore, even if an attacker and a genuine user collude to carry out a substitution attack, the system can resist it. Fig. 2. The proposed protocol for biometrics based secure key sharing.

VII.EXPERIMENTAL EVALUATION ON FACE BIOMETRICS, SECURITY ANALYSIS, AND DISCUSSION

A. Experimental Setup

The biometric verification performance of the proposed protocols is evaluated on face biometrics. In order to validate our proposal, we selected one set of error correcting codes (ECC) and tested the system at different levels of error correction. The ECC used in our system is a two level scheme in which the first level is comprised of BCH codes which performs much of the error correction. If there are error bursts (localized errors) in the biometric data, these cannot be corrected by the first level BCH codes. In the second level, the possible leftover localized errors are corrected with the Reed-Solomon (RS) codes.

The database used for evaluation is a subset of the NIST- FRGCv2 face database [22]. This subset contains 12 images from 250 subjects. Eight of these 12 images are recorded under controlled conditions while the remaining four are from non-controlled conditions. This subset is further split into development and evaluation sets with 125 subjects in each. Moreover, we use the images only from the controlled set which have smaller variations compared to those from the non-controlled set. If the images from non-controlled set are to be used, the error correction capacity of the ECC should be selected accordingly. In practice, an automated quality esti- mation module can be employed for the purpose. We carried out all possible comparisons between the images resulting in 3,500 genuine and 496,000 impostor comparisons on the development as well as the evaluation data sets.

A Gabor filter based approach [23] is followed to ex- tract features from face images. The face image is first geometrically normalized with the CSU Face Recognition Evaluation System [24], and then processed using log-Gabor filters having four scales and eight orientations with the MATLAB programs available at [25]. Magnitude of the filtered output is calculated, downsampled, and concatenated to form a 3,200-element feature vector. The median of the values in a feature vector is calculated and used as a threshold to binarize that feature vector. The binarization process yields a 3,200-bit binary feature vector called face code. The binarization process used is fairly simple.

B. Results

We first carried out comparisons between the binary face feature vectors from the development data set as described above. From the distributions of genuine and impostor Ham- ming distances, we found out that the amount of required error correction is nearly 21%. Therefore, BCH codes of di- mensions BCH(511,28,111) having 21.72% error correction capacity are selected. Here, 28 is the number of data bits to be encoded, 511 is the number of bits after encoding, and 111 is the number of bits that can be corrected. The RS-codes of size RS(24,k_S,t_S) are used where k_S and t_S are the number of input blocks and error correction capacity of the RS-codes, respectively. Each of these blocks is $m = 7$ bits. Four RS-codes output blocks are combined to form an input block of BCH codes. Various tests are conducted by changing the error correction capacity t_S of the RS codes.

During decoding, if the BCH code fails to correct the errors, it outputs a decoding failure flag. In such cases, the 28-bit output of the decoder, and therefore, all four RS- decoder input blocks, are treated as erasures. Assuming that there can be α errors and β erasures, the error correction capacity of RS codes is $2\alpha + \beta < d_{min}$, where d_{min} is the minimum distance of the RS-codes. Since we can predict the erasures for RS codes, they are operated in simultaneous error-erasure mode. The results in terms of False Acceptance Rate (FAR) and False Rejection Rate (FRR) on the evaluation subset are shown in Table I. Note that the FAR is zero

for all these tests. This reduced FAR is because of the shuffling scheme used in the system.

Table 1

VERIFICATION RESULTS OF THE PROPOSED SYSTEM ON NIST-FRGCV2 (FACE) EVALUATION SUBSET; SHUFFLING IS APPLIED ON THE BINARY FACE FEATURE VECTORS; ≈ 21.72% BCH ERROR CORRECTION CAPACITY; $n_S = 24$, $m = 7$.

t_S	**Key Length** (in	**FAR (in**	**FRR (in**
2	140	0	5.60
3	126	0	5.60
4	112	0	2.63
6	84	0	1.14
8	56	0	0.63

As described in Section III-A, the proposed protocol can choose the ECC and error correction capacity dynamically at the time of authentication. Therefore, if it is detected that the face image acquisition conditions are different than those at the time of enrollment (which will result in higher amount of variations), a different set of ECC along with higher error correction capacity can be selected which can still allow successful authentication.

The data that is transferred through the unprotected chan- nel during authentication is the user ID, locked code θ_{lock}, and the hash values $H(\mathbf{K}_r^r)$ and $H(H(\mathbf{K}_r))$. The hash values are obtained using state-of-the-art hash algorithm (such as SHA-256). The security analysis of the hash values is out of the scope of this work. Some possible attacks on the proposed protocol are discussed in the following section.

C. Attacks on the Proposed Session Key Generation and Sharing Protocol and Their Remedies

One of the simplest attacks against biometric systems is the *dictionary attack*. An attacker can run a database of images against the templates in order to obtain a false acceptance. In our system, biometric data shuffling is used which results in zero FAR. Therefore, the dictionary attack becomes ineffective. Note that, the zero FAR is reported by carrying out 496,000 impostor comparisons on the evaluation data set having 125 subjects (additionally, the FAR is zero on the development data set of the same size). The scalability of this system on a larger database with more number of subjects need to be studied further.

We define a possible attack against the proposed protocol denoted as *false rejection attack*. In this attack, an attacker tries to access the system and obtains the locked code θ_{lock}. Since he does not have the right credentials of the user, the system rejects him. By performing this task multiple times, he obtains multiple locked codes. In this protocol, the locked code is generated by XOR*ing* an encoded random key with the stored cancelable template. Thus, having multiple locked codes means multiple messages XOR*ed* with a single cancelable template. In this case, the attacker can decode the messages by breaking the XOR encryption.

This attack can be overcome as follows: when a rejection occurs, the server does not delete the locked code and the hash values but stores them along with the cancelable template. The next time the same user requests for authenti- cation, this stored locked code along with the hash value is sent to the client. In this way, if the user is an attacker, he will receive the same locked code in every login attempt. A new locked code will not be generated unless the previous session was completed successfully. Moreover, after every successful session, the template update process is followed so that the template is renewed. Thus, the stored cancelable template is used only once for creating the locked code.

The proposed protocol can replace the existing key sharing protocols. In order to have additional layer of security, the proposed protocol can also be integrated inside other cryptographic protocols such as TLS. Such classical pro- tocols can first be used to establish a secure connection between the client and the server. Then the protocol shown in Fig. 3 can be employed for biometrics based secure mutual authentication between the client and the server.

VIII. CONCLUSIONS AND FUTURE WORK

In this paper, a framework for biometric-based Crypto-Key generation is proposed. Contributions include a general approach for distinguishable feature generation and a stable key generation mechanism. In this paper, we proposed a novel protocol which enables a client and a server to share biometrics based Crypto-Keys securely to be used for symmetric key cryptography.

In our works, the privacy and security of face data are provided with cancelable template. Also, we propose a protocol with which key can be revoked thus address ing the limitation of irrevocability property of biometric trait. More significantly, there is no need to store the key, prior to communication. In fact, our protocol adds more security allowing to generate different keys in dif- ferent sessions. The proposed crypto-biometric system is resilient to many attacks such as known key attacks, replay attack, man- in-middle attacks, etc. Our proposed approach thus provides an effective solutions where we need a session-based Crypto-Key during message transmission over an insecure network channel.

Additionally, we proposed a novel protocol for generating and sharing session keys which can be used for secure communication. The use of session keys provides better security by limiting the amount of data encrypted with a symmetric key. The protocol achieves mutual authentication – a client can authenticate the server and the server can authenticate the client, without the need of costly third party certificates. Biometrics based user verification is effectively included in the protocol. Successful user verification yields a long key thus producing a strong link between the user identity and his Crypto-Keys. The session key is valid only for a particular communication session. The template stored in the database is cancelable which is obtained using the reference biometric data and an assigned secret key. Thus, the system possesses important properties such as revocability, template diversity, and privacy protection. Moreover, the protocol also facilitates easy online updating of the templates. The protocol can be used for high level security applications where user verification is mandatory. The protocol can further be developed to accommodate multiple biometric modalities for higher security.

IX. FUTURE WORK

Possible future directions include applying to other person-dependent biometric features (e.g. voices, audio- visual dynamics, iris pattern, etc.) and finding a good approach to set up the authentic range for each feature to achieve the optimal overall performance.

X.REFERENCES

[1] *Advanced Encryption Standard (AES)*, Federal Information Processing Standards Publication 197, Standard, November 2001.

[2] N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle, "Gener- ating cancelable face templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 561–572, April 2007.

[3] A. Lumini and L. Nanni, "An improved biohashing for human authen- tication," *Pattern Recognition*, vol. 40, no. 3, pp. 1057–1065, March 2007.

[4] T. E. Boult, W. J. Scheirer, and R. Woodworth, "Revocable face biotokens: Accuracy and security analysis," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[5] S. Kanade, D. Petrovska-Delacrétaz, and B. Dorizzi, "Cancelable Iris Biometrics and Using Error Correcting Codes to Reduce Variability in Biometric Data," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2009.

[6] G. I. Davida, Y. Frankel, and B. J. Matt, "On enabling secure applications through off-line biometric identification," in *Proceedings of the IEEE Symposium on Privacy and Security*, 1998, pp. 148–157.

[7] F. Monrose, M. Reiter, and R. Wetzel, "Password hardening based on keystroke dynamics," in *Proceedings of the Sixth ACM Conference on Computer and communication Security (CCCS)*, 1999, pp. 73–82.

[8] S. Argyropoulos, D. Tzovaras, D. Ioannidis, and M. G. Strintzis, "A Channel Coding Approach for Human Authentication From Gait Sequences," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 3, pp. 428 – 440, 2009.

[9] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proceedings of the Sixth ACM Conference on Computer and com- munication Security (CCCS)*, 1999, pp. 28–36.

[10] A. Juels and M. Sudan, "A fuzzy vault scheme," in *Proc. IEEE Int. Symp. Information Theory*, A. Lapidoth and E. Teletar, Eds. IEEE Press, 2002, p. 408.

[11] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Proceedings of the Eurocrypt 2004*, 2004, pp. 523–540.

[12] F. Hao, R. Anderson, and J. Daugman, "Combining crypto with biometrics effectively," *IEEE Transactions on Computers*, vol. 55, no. 9, pp. 1081–1088, 2006.

[13] S. Kanade, D. Camara, E. Krichen, D. Petrovska-Delacrétaz, and B. Dorizzi, "Three Factor Scheme for Biometric-Based Crypto-Key Regeneration Using Iris," in *The 6th Biometrics Symposium (BSYM)*, September 2008.

[14] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith, "Secure Remote Authentication Using Biometric Data," in *Eurocrypt*, 2005.

[15] Y. Ueshige and K. Sakurai, "A Proposal of One-Time Biometric Authentication," in *Security and Management*, H. R. Arabnia and S. Aissi, Eds., 2006.

[16] J. Bringer, H. Chabanne, M. Izabachène, D. Pointcheval, Q. Tang, and S. Zimmer, "An Application of the Goldwasser-Micali Cryptosystem to Biometric Authentication," in *The 12th Australasian Conference on Information Security and Privacy (ACISP '07)*, 2007.

[17] Q. Tang, J. Bringer, H. Chabanne, and D. Pointcheval, "A Formal Study of the Privacy Concerns in Biometric-Based Remote Authen- tication Schemes," in *Information Security Practice and Experience Conference (ISPEC)*, 2008.

[18] T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) Pro- tocol Version 1.2," Request for Comments: 5246, Internet Engineering Task Force (IETF), August 2008.

[19] I. Buhan, "Crypto-Keys from Noisy Data," Ph.D. dissertation, University of Twente, Netherlands, 2008.

[20] W. J. Scheirer and T. E. Boult, "Bio-Cryptographic Protocols with Bipartite Biotokens," in *Biometric Symposium*, 2008.

[21] ——, "Bipartite Biotokens: Definitions, Implementation, and Analy- sis," in *International Conference on Biometrics (ICB)*, 2009.

[22] National Institute of Science and Technology (NIST), "Face Recogni- tion Grand Challenge," 2005, http://www.frvt.org/FRGC/.
[23] M. Lades, J. C. Vorbrüuggen, J. Buhmann, J. Lange, C. v.d. Malsburg, R. P. Wüurtz, and W. Konen, "Distortion Invariant Object Recognition in the Dynamic Link Architecture," *IEEE Transactions on Computers*, vol. 42, no. 3, pp. 300–311, March 1993.
[24] J. R. Beveridge, D. Bolme, B. A. raper, and M. Teixeira, "The CSU Face Identification Evaluation System," *Machine Vision and Applications*, vol. 16, no. 2, pp. 128–138, 2005.
[25] P. Kovesi, "Matlab and octave functions for computer vision and image processing,"