The 7th International Telecommunications Symposium (ITS 2010)

Providing Higher Entropy Cryptographic Keys by the Use of Biometrics

D. P. B. A. Camara and V. C. da Rocha Jr.
Communications Research Group - CODEC
Department of Electronics and Systems
Federal University of Pernambuco
Recife, PE, BRAZIL, 50711-970, PO Box 7800
e-mail: dpbac@ieee.org, vcr@ufpe.br

***Abstract*— In this paper modifications are investigated to a previously introduced Key Regeneration system used to obtain cryptographic keys from biometric data, specifically from the iris. These modifications improve the performance of the previous system making possible the regeneration of longer and higher entropy cryptographic keys. The modified system was evaluated on two public databases: CBS-BiosecureV1 and NIST-ICE 2005. On NIST-ICE 2005 it is possible to regenerate a 287 binary digit cryptographic key with an estimated entropy of 231 bits at 0.28% False Acceptance Rate (FAR) and 6.30% False Rejection Rate (FRR). The cryptographic keys regenerated possess enough length and entropy to be used on practical cryptosystems like, for example, the AES.**

Keywords*— *Biometrics, cryptography, error-correcting codes, security.

I. Introduction

Biometrics verification techniques have been used for many decades providing authentication/identification of an individual based on his unique characteristics, e.g., fingerprint, iris, voice, hand geometry etc. In particular, the use of biometrics has grown significantly these last decades raising important concerns about the individual privacy and data confidentiality since conventional biometric solutions require direct storing of user personal data [1]. On the other hand, cryptography is able to assure high data privacy as long as the cryptographic key is secret, long and as random as possible to provide the required security level (e.g., the Advanced Encryption Standard (AES) was designed to support encryption keys of length 128, 192 or 256 bits [2]). However, classical cryptographic keys can not assure that the person using it is actually the genuine user (non-repudiation). The complementary nature of these two important and widely used security tools stimulated many researchers to investigate new techniques capable of combining biometrics and cryptography in order to provide privacy to biometric data and obtain cryptographic keys truly linked to the user. The main drawback of this combination is the inherent variabilities in biometric data because so far cryptographic systems require exactitude to work properly. One of the approaches used to obtain cryptographic keys from biometrics, known as *Key Regeneration*, deals with this drawback using *Error-Correcting Code* (ECC) techniques.

In this paper we propose a modification to the scheme introduced in [3] making it possible the regeneration of cryptographic keys that are longer and have higher estimated entropy than the ones obtained by other systems, showing also good biometric performance for real world applications. Experiments were performed on CBS-BiosecureV1 [4] and NIST-ICE 2005 [5] databases and, for example, 287 binary digit keys with 231 bit estimated entropy at 0.28% false acceptance rate (FAR) and 6.30% false rejection rate (FRR)* were regenerated on the ICE-NIST2005 database.

This paper is organized as follows. Section II provides the necessary background for understanding this paper. The new proposal is introduced in Section III. In Section IV we describe the experiments performed and present the results obtained. In Section V we present a security analysis of the proposed scheme. Summing up, in Section VI we present some conclusions about this work as well as some suggestions for future research.

II. Background

Basically three approaches are used to combine cryptography and biometrics, namely *Cancelable Biometrics*, *Key Generation* and *Key Regeneration*.

The *Key Regeneration* approach has been considered the most effective way to combine biometrics and cryptography in order to obtain cryptographic keys strongly linked to the user (non-repudiation), allowing key revocability and key diversity†. This approach also provides privacy to the biometric data. ECC techniques are used in order to deal with biometrics inherent variability [3], [6]–[10].

In 1999 Juels and Wattenberg [11] proposed the use of ECC to deal with this variability in order to obtain cryptographic keys. However, no practical ECC technique was proposed.

Dodis et al. introduced in 2004 [7] the idea of fuzzy extractor, a primitive capable to extract nearly uniform randomness R from its biometric input in an error-tolerant way assuring that R will be the same if the input changes but remains close. They observed that the construction proposed in [11] can be used to obtain a nearly optimal fuzzy extractor. As in [11] no specific ECC technique was proposed on this paper [7].

In 2006 Hao et al. [8] introduced a Key Regeneration system based in iris using concatenated ECC [12, page 740]

*FAR and FRR are parameters used to measure the performance of biometric systems, where FAR is the measure of the likelihood that genuine users will be rejected by the system and FRR is the measure that false users will be accepted by the system.

†Different keys are associated with different applications using the same biometric data.

combining a Hadamard code and a Reed-Solomon (RS) code. As explained in [8], the Hadamard code is used to deal with background errors (random errors) caused for example by camera noise, iris distortion, image-capture effects that cannot be effectively corrected by the preprocessing phase while the RS code deals with burst errors caused for example by eyelashes, eyelids and reflections. This system is able to regenerate 140 binary digit keys with estimated entropy of 44 bits at 0.47% FRR and 0% FAR over a 700-image proprietary database. The same scheme tested over a public database, NIST-ICE 2005 [5], showed very high FRR, e.g. 19.41% for 42 bit key. Kanade et al. [3] improved the system proposed by Hao et al. inserting two new mechanisms but maintaining the ECC technique that suits the mixed error structure presented by the iris. As a result, it is possible to regenerate 198 binary digit cryptographic keys with estimated entropy 83 bits at 0.055% FAR and 1.04% FRR on NIST-ICE 2005 database [5]. Although, the scheme proposed by Kanade et al. [3] presents better results than the previous ones (e.g., [8], [9]) the entropy obtained is not enough for real cryptographic applications [2], [13], [14] because for a key length of 198 bits the enemy needs just 2^{83} trials to break the system, against 2^{197} trials on average required to break a truly random key. Another paper by Kanade et al. [10] showed a different scheme also using concatenation of RS and Hadamard codes providing 94 bit entropy cryptographic keys. The key length in [10] depends on the hash function chosen since the cryptographic key is the hash value of the reference iris code.

Cryptographic keys obtained by Key Regeneration approach are subject to some constraints because of the required performance of the biometric system, e.g., the FAR and FRR parameters, and the ECC technique used. Every biometric recognition system has a built-in acceptance threshold, which when raised both decreases FAR and increases FRR. The choice of this threshold is usually done based on the application, e.g., high security applications requires low FAR while commercial applications requires low FRR.

The use of ECC in Key Regeneration systems is very peculiar. In order to choose the appropriate ECC technique the behavior of biometrics variability of certain biometric characteristic must be observed, e.g. the iris presents mixed random and burst errors. Moreover, the error-correcting capability of the code must be enough to correct *intra-user variations*, e.g. differences between error bits for the same eye, but unable to correct *inter-user variations*, e.g. differences between different eyes. Many researchers are considering the use of the iris for cryptographic key regeneration since irises are considered the most attractive biometric feature because of their high level of accessible entropy [15, Chapter 3], e.g. the iris appears to contain on the order of 249 bits of entropy using Daugman method [16] to extract the iris code[‡] while a fingerprint contain at most 20 bits of entropy. In addition, irises recognition systems are the most accurate, especially for large databases and also present the best search speed.

[‡]Iris code is the denomination given to a bit string obtained after procedures of iris capture, localization, normalization and feature extraction [17, Chapter 3].

III. Modifications Proposed

In this section we present modifications to the iris based Key Regeneration system introduced in [3]. As indicated in Section IV, these modifications make it possible the regeneration of longer and stronger cryptographic keys. In order to make the proposal clearer we start by describing the technique introduced in [3].

A. Kanade et al. scheme

Fig. 1 shows the Key Regeneration system proposed in [3]. During the *enrollment (key generation) phase* a random cryptographic key K is generated and encoded by the concatenation of RS and Hadamard codes (Fig. 2) resulting in θ_{ps}, denominated *pseudo-iris code*. The hash value of K, $h(K)$, is stored in a smart card while K is discarded.

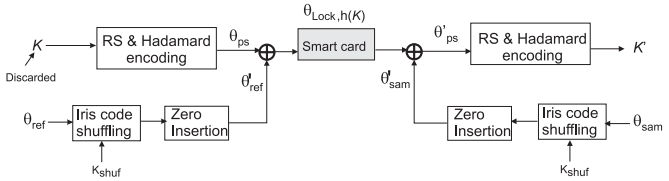

Fig. 1. Key Regeneration using smart card, iris and password.

The user presents his iris to the biometric system and his reference iris code, θ_{ref}, is extracted[§]. A 198 binary digit random generated *user specific shuffling key*, K_{shuf}, is used to determine how the iris code is shuffled (Fig. 3). After the iris code shuffling some zeros are inserted uniformly and the string is truncated in order to obtain a 1,952 bit string, called the *modified reference iris code*, θ'_{ref}. This modified reference iris code, θ'_{ref} is then combined with θ_{ps} by a bitwise exclusive-or (XOR) operation, resulting in

$$\theta_{lock} = \theta_{ps} \oplus \theta'_{ref}. \quad (1)$$

K_{shuf}, θ_{lock} and $h(K)$ are stored in a smart card protected by a password. During the *verification (regeneration key) phase* the user presents his iris and his smart card containing K_{shuf}, θ_{lock} and $h(K)$ to the system.

A *sample iris code* is extracted, θ_{sam} and passes through the same iris code shuffling, zero insertion and truncation operations as performed during the enrollment phase. The *modified sample iris code*, θ'_{sam}, obtained by this procedure is so combined with θ_{lock} (Eq.1) by a bitwise XOR operation, resulting in:

$$\theta^*_{ps} = \theta'_{sam} \oplus \theta_{lock} = \theta'_{sam} \oplus \theta'_{ref} \oplus \theta_{ps} = e \oplus \theta_{ps}, \quad (2)$$

where e indicates the errors between the two iris codes.

θ^*_{ps} is decoded by the concatenation of Hadamard and RS codes resulting in K' that is hashed and compared with $h(K)$. If $h(K')$=$h(K)$ it means that K=K' with high probability, as

[§]OSIRIS (Open Source Iris System) developed under the Biosecure project [17, pp.34-40] is used to extract a 1,188 bit iris code.

a consequence the cryptographic key is considered valid and can be used successfully by the cryptosystem. Notice that the user identity is also verified assuring non-repudiation of the key.

1) RS and Hadamard codes in the scheme: The concatenated code used in the scheme is formed by a t_s-error-correcting (n_s,k_s) RS code with symbols from $GF(2^m)$ and a t_{HC}-error-correcting $(2^k,k+1)$ Hadamard code, denoted respectively, by $RS(n_s,k_s,t_s)$ and $HC(2^k,k+1,t_{HC})$ where n_s is the number of m bit blocks after encoding and k_s is the number of m bit blocks before encoding, k is the order of the Hadamard matrix that is obtained by the Sylvester method. Observe that in order to make the two codes work properly in concatenated form, it is required to set $m = k+1$ (Fig. 2).

RS codes are MDS (*Maximum Distance Separable*) [12, pp.238], i.e., $d_{RS} = 2t_s+1 = n_s-k_s+1$ and thus,

$$n_s - k_s = 2t_s \tag{3}$$

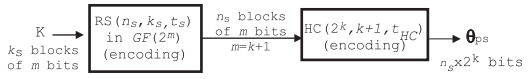

Fig. 2. Concatenated code, $RS(n_s,k_s,t_s)$ and $HC(2^k,k+1,t_{HC})$.

More details about these codes can be obtained in [12], [18].

2) Iris Code Shuffling: The *Iris code shuffling* permutes the bits of the iris code based on the user specific shuffling key, K_{shuf}. This shuffling procedure consists of, first, dividing the 1,188 bit θ_{ref} in 198 blocks of 6 bits, then the iris code blocks are aligned with the 198 bit K_{shuf} and the blocks where the K_{shuf} bit is one are sorted first and the remaining blocks are sorted at the end (Fig. 3).

Because the shuffling scheme is based on a user specific shuffling key, when a genuine user uses his shuffling key, the shuffling is the same and no errors are introduced; however, if an impostor uses his shuffling key, the iris code is shuffled in a different way and errors are introduced. In this manner the separation between genuine and impostor Hamming distance distribution is increased improving the biometric performance of the system ([3, Fig. 3]). Another advantage is that K_{shuf} is kept secret, encrypted in an smart card, for example, and so acts as an auxiliary secret for the Key Regeneration scheme improving the security and also as another factor that provides revocability to the system.

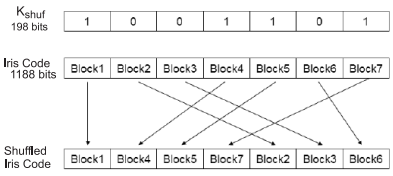

Fig. 3. Iris code shuffling

3) Zero insertion: The Hadamard code corrects up to $2^{k-2}-1$ errors in 2^k bits which means that its error-correcting capability is limited to 25%. Experiments showed that this error-correcting capability is not enough to deal with variabilities present in the iris [19]. The insertion of zeros in the shuffled iris code¶ is able to adjust the number of errors to match the error-correcting capability of the concatenated code to the desirable level. By zero insertion the total number of errors remains the same, but the number of errors per block decreases. Suppose there are p errors in a n bit block, where n denotes the length of θ_{ps}. If q zeros are uniformly inserted in this block, there will be p errors in $(n+q)$ bits. The error ratio decreases from p/n to $p/(n+q)$, and if at most 25% of the bits in each codeword of the Hadamard code are in error, they can be corrected, and all the p errors can thus be corrected.

B. New Proposal

The cryptographic key length $||K|| = m \cdot k_s$ is a function of the parameters of the code and the length of the modified iris code and is expressed as

$$||K|| \quad = \quad m \cdot (n_s - 2t_s) = m \cdot \left(\frac{\| \theta'_{ref} \|}{2^k} - 2t_s \right). \tag{4}$$

The output of the concatenated code, θ_{ps}, has its length limited by the length of the modified iris code and must be equal to $||\theta'_{ref}||$, consequently limiting $||K||$ (Fig. 1).

Zero insertion is important to improve the error-correcting performance while maintaining the ECC structure. However, inserting more zeros can increase security vulnerability, i.e., it can cause security issues by leaking an amount of information that could be explored by the enemy, since by (Eq. 1), $\theta_{lock} = \theta_{ps}$ in the positions where zeros are inserted.

Therefore, in order to increase the length of the code output and consequently the cryptographic key length we proposed to duplicate the modified iris code length by applying twice the iris code shuffling and the zero insertion. The resulting modified iris code is formed by the concatenation of two modified iris codes (e.g., $\theta'_{ref} = \theta'_{ref1}|\theta'_{ref2}$) (Fig. 4).

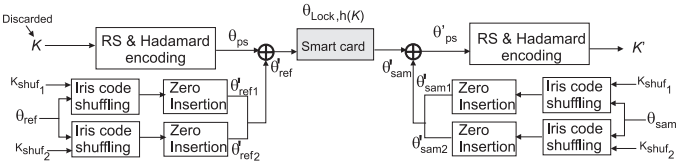

Fig. 4. Key Regeneration using smart card, iris and password applying the iris code shuffling twice and zero insertion to increase cryptographic key length and entropy.

Two 198 bit shuffling keys must be stored in the smart card, instead of one key only. Experiments were also performed considering two 99 bit shuffling keys in order to maintain the same storage use. Many trials were performed over different codes by varying the code parameters m and n_s and also observing the amount of inserted zeros. More details about the experiments and results are given in the next section.

¶In the system introduced by Kanade et al. [3] after each 3 bits of the 1,188 bit shuffled iris code 2 zeros are inserted and the the resulting 1,980 bit string has its last 28 bits removed to form the final 1,952 bit modified iris code.

IV. Experiments and Results

For the experiments different values of code parameters (n_s, m) were chosen taking in account factors as cryptographic key length and estimated error-correction capability. The amount of inserted zeros is limited in such a way that its relation to the length of the modified iris code is less or equal to the one used in [3]. In fact we have tried to insert as few zeros as possible to avoid information leak. These parameters were kept fixed while the system was tested for different values of t_s. As illustrated in Table I lower values of t_s result in longer keys but with higher FRR and vice-versa. Thus t_s acts as a second level threshold by adjusting which we can fine tune the system performance.

CBS-BiosecureV1 and NIST-ICE 2005 database were used to evaluate the system.

- CBS-BiosecureV1 is part of the CBS (Casia-Biosecure) database and contains 1200 images from 60 eyes of 30 persons with 20 images. On this database 6000 genuine comparisons, and 6000 impostor comparisons were performed.
- NIST-ICE 2005 database consists of 2,953 images from 244 different eyes. Two experiment were carried out for this database: Exp-1 - with right eyes which consists of 1,425 images of right irises from 124 users resulting in 12,214 genuine comparisons, and 1,002,386 impostor comparisons and Exp-2 - with left eyes that consists of 1,528 images of left irises from 120 users resulting in 14,653 genuine comparisons, and 1,151,975 impostor comparisons.

In this section we will show the best results obtained at this moment considering: (1) FAR as close to zero as possible since we are considering a security application, (2) low FRR to avoid user annoyance, (3) cryptographic key lengths and (4) entropy equal to or higher than the ones required by actual cryptosystems. The estimation of the entropy, H, is done using the same criterion used by Hao et al. [8] and Kanade et al. [3], based on the sphere-packing bound [18]. More details are given in Section V.

Tables I and II show results in terms of FAR, FRR and cryptographic key length, $||K||$, obtained by an experiment performed on CBS-BiosecureV1 and NIST-ICE 2005 databases, respectively. In these experiments the parameters for the ECC are $n_s = 51, m = 7$, varying t_s. Four zeros are inserted after each 12 bits and additional 12 zeros are appended to the end of the iris code. A 31.91% error correcting capability is achieved after zero insertion. The estimated entropy is 231 bits (Eq. 6).

The use of two 99 bit K_{shuf} increases FRR and decreases FAR in a small amount. Therefore, it is an option to maintain the same amount of data stored as in [3].

TABLE I
Results in terms of False Acceptance Rate), False Rejection Rate and cryptographic key length, $||K||$, on CBS-BiosecureV1 database.

t_s	1	2	3	4	**5**	6
Experiments using two 198 bit K_{shuf}						
FRR(%)	15.02	11.03	9.12	7.92	**7.13**	6.5
FAR(%)	0	0	0	0.02	**0**	0.08
Experiments using two 99 bit K_{shuf}						
FRR(%)	17.17	12.7	10.08	8.93	**8.07**	7.33
FAR(%)	0	0	0	0	**0**	0.03
||K||	343	329	315	301	**287**	273

TABLE II
Results in terms of False Acceptance Rate, False Rejection Rate and cryptographic key length, $||K||$, on NIST-ICE 2005 database.

t_s	1	2	3	4	**5**	6
Experiments using two 198 bit K_{shuf}						
Exp-1						
FRR(%)	24.08	15.91	10.96	8.20	**6.30**	4.88
FAR(%)	0	0.01	0.03	0.13	**0.28**	0.62
Exp-2						
FRR(%)	25.70	17.86	13.03	10.09	**7.96**	6.32
FAR(%)	0	0.01	0.03	0.07	**0.19**	0.38
Experiments using two 99 bit K_{shuf}						
Exp-1						
FRR(%)	26.76	17.78	12.81	9.55	**7.25**	5.49
FAR(%)	0	0	0.01	0.05	**0.13**	0.29
Exp-2						
FRR(%)	29.04	19.87	15.07	11.2	**8.78**	7.34
FAR(%)	0	0.01	0.02	0.04	**0.13**	0.31
||K||	343	329	315	301	**287**	273

V. Security Analysis

The Key Regeneration system makes use of all three factors used for authentication: (a) what the user knows (e.g. password), (b) what the user possesses (e.g. smart card) and (c) what the user is (e.g. biometrics), in order to provide a higher level of security since authentication systems that incorporate all three factors are more secure than the systems that incorporate only one or two of these factors [20].

Considering that the attacker can obtain the smart card, the security of the system will rely on the iris and the user shuffling key, K_{shuf}. Supposing that the enemy was able to guess the correct 198 bit random number, the enemy must also provide the correct iris code. In order to set a lower bound on the number M of trials necessary for the enemy to find the right iris code we consider the worst case assuming that the enemy knows all the correlations within the user's iris. It has been proved that these correlations exist but it is not clear yet how they can be exploited [16]. Therefore, by considering the sphere packing bound [18, p. 19] it follows that

$$M \geq \frac{2^z}{\sum_{i=0}^{w} \binom{z}{i}} \simeq \frac{2^z}{\binom{z}{w}}, \qquad (5)$$

where $z = 1,172$, i.e., the uncertainty provided by the iris code extracted using OSIRIS and shuffled is $z = 1,172$ [16] and $w = \frac{t_{HC}}{n_{HC}} \times z$.

In order to illustrate the meaning of this bound consider an experiment presented in Table II where $n_s = 51$ and $m = 7$ over the NIST-ICE 2005 database. For this experiment the estimated error correction capability is 31.91%, so $w = 0.3191 \times 1,172 = 374$. By Eq. (5) $M \simeq 2^{231}$ which means that the enemy must

try to find a 1,172 bit string within 231 bits Hamming distance of the key. In other words, the entropy provided by the system is $\log_2(M) = 231$ bits, i.e,

$$H \simeq \log_2 M. \quad (6)$$

Table III compares actual iris based cryptographic key regeneration algorithms. It can be observed that our proposal (in bold) achieves better values of cryptographic key length and entropy. However, the FRR is higher than the ones achieved by the other schemes. As indicated in Table III, for the system introduced in [3] it is possible to regenerate 282 bit cryptographic keys with estimated entropy of 83 bits at 9.54% FRR and 0% FAR on the ICE-NIST database. Except for FRR, the result in bold confirms that the modification we have introduced improved the performance of the system proposed by Kanade et al. [3].

TABLE III
COMPARISON BETWEEN IRIS BASED CRYPTOGRAPHIC KEY REGENERATION ALGORITHMS (VALUES OF FAR AND FRR ARE IN %); ECC – ERROR-CORRECTING CODE, RSH – REED-SOLOMON AND HADAMARD CODE, RMP – REED-MULLER AND PRODUCT CODE

Ref.	ECC	Key bits	Entropy	FRR	FAR	Database
[8]	RSH	140	44	0.47	0	proprietary
[9]	RMP	42	ND	5.62	10^{-5}	ICE
[3]	RSH	198	83	1.04	0.055	ICE-Exp-1
[3]	RSH	282	83	8.42	0	ICE-Exp-1
[10]	RSH	variable	94	0.76	0.096	ICE-Exp-1
-	**RSH**	**287**	**231**	**6.30**	**0.28**	**ICE-Exp-1**

In order to improve the smart card content security the maximum number of login attempts before lockout can be limited. We suggest the possibility of using another biometric feature of the same individual to unlock the smart card instead of a password.

VI. CONCLUSIONS

This paper introduces a modification in the system proposed in [3] making it possible the regeneration of 287 binary digit cryptographic keys with estimated entropy of 231 bits on the NIST-ICE 2005 database. Table III shows that with the proposed modification we are able to regenerate cryptographic keys longer and stronger than the ones obtained by other proposals [3], [8]–[10]. It is worth noticing that the key length and entropy obtained can be used by real cryptosystems. The FAR is kept near to zero which is important for security applications as the one considered here. There is still room to reduce the FRR, however as can be seen in Table III our current proposal produces better results than the system in [3].

The proposed system will also be evaluated on CBS-CasiaV2 database that is a subset of CASIA Version 2 database that contain 1200 images from 60 eyes of 30 persons with 20 images from each eye. The testing protocol consists on 6000 genuine comparisons, and 6000 impostor comparisons.

The results obtained so far showed good improvements except for the FRR. Because of that we are also considering other possible scenarios, for example, by taking into account other codes, i.e., other values for m and n_s, and the use of three shuffling keys, all of the same length, either 198 binary digits or 66 binary digits long. We intend to evaluate the new scenarios on CBS [4] and NIST-ICE 2005 [5] database.

ACKNOWLEDGEMENTS

The authors acknowledge partial support from the Pernambuco State Foundation to Support Science and Technology - FACEPE and the Brazilian National Council for Scientific and Technological Development - CNPq under Grants APQ-0055-3.04/09 and No. 306612/2007-0, respectively.

REFERENCES

[1] A. K. Jain, P. Flynn and A. A. Ross, *Handbook of Biometrics*, Springer, 2008.
[2] "Advanced encryption standard (AES)", Federal Information Processing Standards Publication 197, National Institute of Standards and Technology, November 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.
[3] S. Kanade, D. Camara, E. Krichen, D. Petrovska-Delacrétaz and B. Dorizzi, "Three factor scheme for biometric-based cryptographic key regeneration using iris," *The 6th Biometrics Symposium 2008 (BSYM2008)*, pp. 59 - 64, 2008.
[4] "BioSecure Network of Excellence," www.biosecure.info.
[5] National Institute of Science and Technology (NIST), "Iris Challenge Evaluation," 2005, http://iris.nist.gov/ice.
[6] A. Juels and M. Sudan, "A fuzzy vault scheme," *Proc. IEEE Int. Symp. Information Theory*, p. 408, 2002.
[7] Y. Dodis, L. Reyzin and A. Smith, "Fuzzy Extractors : How to generate strong keys from Biometrics and other noisy data", *Advances in Cryptology - Eurocrypt 2004*, vol. 3027, pp. 523-540, May 2004.
[8] F. Hao, R. Anderson and J. Daugman, "Combining crypto with biometrics effectively", *IEEE Transactions on Computers*, vol. 55, No. 9, pp. 1081-1088, 2006.
[9] J. Bringer, H. Chabanne, G. Cohen, B. Kindarji and G. Zmor, "Theoretical and practical boundaries of binary secure sketches," *IEEE Transactions on Information Forensics and Security*, Vol. 3, No. 4, pp. 673-683, December 2008.
[10] S. Kanade, D. Camara, D. Petrovska-Delacrétaz and B. Dorizzi, "Application of biometrics to obtain high entropy cryptographic keys", *Proceedings of World Academy of Science, Engineering and Technology*, Hong Kong, China, Vol. 39, pp. 264 - 268, March 2009. http://www.waset.org/pwaset/v39/v39-45.pdf
[11] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proceedings of the Sixth ACM Conference on Computer and Communication Security (CCCS)*, pp. 28-36, 1999.
[12] S. Lin, D. J. Costello, *Error Control Coding*, 2nd Edition, Prentice Hall, 2004.
[13] B. Schneier, *Applied Cryptography : Protocols, Algorithms and Source Code in C*, Second Edition, Wiley Publishing, Inc., 1996.
[14] A. J. Menezes, P. C. Van Oorschot e S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press Series, 1996.
[15] P. Tuyls, B. Skoric and T. Kevenaar, *Security with Noisy Data: On Private Biometrics, Secure Key Storage and Anti-Counterfeiting*, Springer 2007.
[16] J. Daugman, "The importance of being random: Statistical principles of iris recognition," *Pattern Recognition*, vol. 36, no. 2, pp. 279-291, February 2003.
[17] D. Petrovska-Dlacretaz, G. Cholet, B. Dorizzi; *Guide to Biometric Reference Systems and Performance Evaluation*, Springer, 2009.
[18] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North Holland, 1988.
[19] J. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1148-1161, November 1993.
[20] W. E. Burr, D. F. Dodson, and W. T. Polk, "Electronic authentication guideline: Recommendations of the National Institute of Standards and Technology," April 2006 http://csrc.nist.gov/publications/nistpubs/800-63/SP800-63V1_0_2.pdf.