

# Biometric Cryptography: Key Generation Using Feature and Parametric Aggregation

Christopher Ralph Costanzo  
School of Engineering and Applied Sciences  
Department of Computer Science  
The George Washington University  
14 October 2004

## **Abstract**

An approach is described for generating a cryptographic key from an individual's biometric for use in proven symmetric cipher algorithms. The proposed approach uses a method referred to as Biometric Aggregation. The encryption process begins with the acquisition of the required biometric samples. Features and parameters are extracted from these samples and used to derive a biometric key that can be used to encrypt a plaintext message and its header information. The decryption process starts with the acquisition of additional biometric samples from which the same features and parameters are extracted and used to produce a "noisy" key as done in the encryption process. Next, a small set of permutations of the "noisy" key are computed. These keys are used to decrypt the header information and determine the validity of the key. If the header is determined to be valid, then the rest of the message is decrypted. The proposed approach eliminates the need for biometric matching algorithms, reduces the cost associated with lost keys, and addresses non-repudiation issues. This paper reports on work in progress.

## **1. Introduction**

This paper proposes a technique for generating keys for symmetric cipher algorithms, such as the widely used Data Encryption Standard (DES) and 3-DES, although it can be extended to asymmetric algorithms as well. There are several problems which must be addressed in order to generate a useful biometric cryptographic key. This paper will consider those associated with (1) the entropy (strength) of the biometric key, (2) uniqueness of biometric key, and the (3) stability of the biometric key.

1. *Key Entropy (strength).* Instead of developing simply longer cryptographic keys to resist brute force attacks, a more intelligent approach might be to aggregate features and parameters from an individual in such a way that their correlation generates a key that is much stronger than the individual size of the actual key.
2. *Key Uniqueness.* The uniqueness of a biometric key will be determined by the uniqueness of the individual biometric characteristics used in the key. Instead of trying to find a single unique feature, a biometric key needs to find only a collection of somewhat unique features or parameters that when assembled collectively create a unique profile for an individual. The incorporation of a

simple passphrase will improve the accuracy of the biometric key by incorporating “something you are” with “something you know.”

3. *Key Stability.* A major problem with biometric identification is that individual’s enrollment template and sample template can vary from session to session. This variation can occur for a number of reasons including different environments (e.g. lighting, orientation, emotional state) or physical changes (e.g. facial hair, glasses, cuts). If a set of relatively stable features can be determined and the amount of variation can be reduced to an acceptable number of bits, then it might be possible for a valid user to search a limited key space to recover an encrypted transmission while making a brute force search by an attacker remain difficult if not impossible.

The motivation for this approach comes from studying the improvements being made in biometrics and computer processing speeds as well as the limitations associated with existing cryptographic operational requirements. Biometric cryptography does not require that a complete solution to the biometrics problem (e.g. find a trait that will correctly identify a user under all conditions) be found.

## **2. Previous Work**

There has been relatively little work done on generating keys using biometrics to date. This is primarily because biometrics does not produce the same matching templates every time a sample is collected and also because cryptography relies on a stable and unique key to encrypt and decrypt messages. There are two approaches, key release and key generation, which have been proposed to address incorporation of biometrics into cryptography (Uludag 2004).

Key release algorithms described in the literature (Soutar 1998; Clancy 2003; Roginsky 2004) require that (1) the cryptographic key is stored as part of the user’s database, (2) requires access to biometric templates for matching, and (3) user authentication and key release are completely decoupled. While this technique does work, there are several problems that result. One problem is that there is no way to ensure who produced the key. The user could deliberately choose a known weak key. Second, if the key is stored in a database, the information could be hacked by spoofing the matcher, and third an enrollment process is required to store the template.

Key generation approaches avoid some of the problems associated with key release approaches by (1) binding the secret key to the biometric information and (2) not requiring access to a biometric template. Unfortunately, even key generation approaches so far described in the literature (Davida G. I. 1998; Davida G. I. 1999; Juels 1999; Monroe 1999; Monroe 2001; Juels 2002; Clancy 2003; Linnartz 2003) required prealigned sample representations, intensive calculations, and more complicated systems than their key release counterparts.

### 3. Application Description

Figure 1 illustrates the system structure for Biometric Cryptography. The encryption and decryption processes are described in this illustration. The encryption process takes as input a plaintext message and header and uses a biometric identifier as the cipher key. The decryption process is similar to the encryption process except that it is responsible for computing a limited number of permutations of the sample key with the hope that one of the permutations will match the original encryption key.

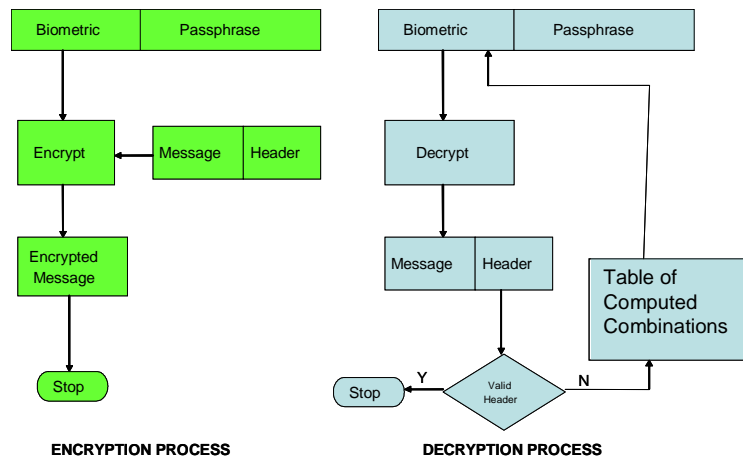


Figure 1. Biometric Cryptography Process

### 4. Biometric Aggregation

A novel approach to defining a biometric key is currently being explored. Biometric Aggregation is an extension of the *AdaBoosting* concept where aggregation of smaller well defined classifiers can provide a more accurate classification than any single classifier (Freund 1996; Maclin 1997). Figure 2 provides a more detailed description of the biometric identifier being used to encrypt and decrypt the message and header.

Fingerprint Classifiers / Parameters (Each Hand, $X_1$ bits)	Faceprint Classifiers / Parameters (Each Hand, $X_2$ bits)	Palmprint Classifiers / Parameters (Each Hand, $X_3$ bits)	Iris Classifiers / Parameters (Each Hand, $X_4$ bits)	Retina Classifiers / Parameters (Each Hand, $X_5$ bits)	Speechprint Classifiers / Parameters (Each Hand, $X_6$ bits)
--	--	--	---	---	--

Figure 2. Biometric Aggregate Identifier

It should be noted that a brute force attack could be attempted against each component of this key to find a weakness. However, if each part of the key could be considered to be strong enough (e.g. 56 bit DES key can be cascaded) and since each part of the key is related, the overall key strength should be greater than the minimum bit length of a constituent.

## 5. Simulations and Evaluations

Some simulations and evaluations have been performed using fingerprints that indicate that the proposed approach has merit. The use of fingerprints was chosen for several reasons. First, it is the most studied biometric to date and large databases of fingerprint data are available for analysis. Second, fingerprint recognition has been shown to be effective for distinguishing between users. Third, fingerprint recognition is becoming widely accepted as the biometric most suitable for large scale implementation (IBG 2001). The two areas being investigated in this research are the use of fingerprint classifiers (e.g. loop, arch, whorl) and fingerprint parameters (e.g. ridge count between minutiae).

It is necessary to estimate how much variation across a key can be allowed in the proposed Biometric Aggregation approach. To understand this, a table of possible combinations a key might have was computed using equation 1 below (Devore 1995):

$$(Equation\ 1) \quad \binom{n}{k} = \frac{n!}{k!(n-k)!} = \frac{P_{k,n}}{k!}$$

These results are shown in Table 1 for several typical symmetric key lengths.

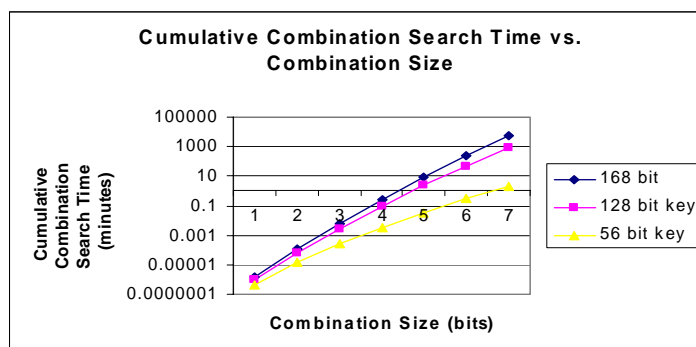
Realistically, this research assumes that there will be some random number of bits that might vary from sample to sample and therefore the entire key space associated with some maximum number of varying bits would have to be searched and therefore a cumulative chart adding up all of the previous combinations was tabulated as well as summarized in Table 2. Finally, a graph relating the amount of time needed to search the cumulative combinatorial key space was computed assuming that each encryption (and decryption) would take 1 uS. This is plotted in Figure 3.

Combination Size	Total number of combinations		
	168 bit key	128 bit key	56 bit key
1	168	128	56
2	14028	8128	1540
3	776216	341376	27720
4	32018910	10668000	367290
5	1050220248	264566400	3819816
6	28530983404	5423611200	32468436
7	6.60288E+11	94525795200	231917400

**Table 1. Total Number Combinations vs. Combination Size**

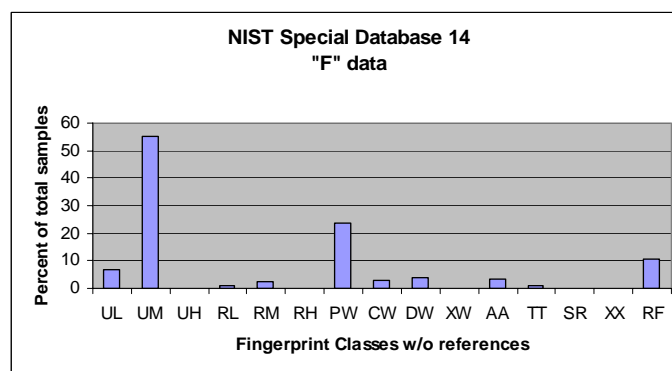
Combination Size	Cumulative Combinations			Combination Search Time (1 Encrypt/uS) E&D (minutes)	Combination Search Time (1 Encrypt/uS) E&D (minutes)	Combination Search Time (1 Encrypt/uS) E&D (minutes)
	168 bit key	128 bit key	56 bit key	168 bit	128 bit key	56 bit key
1	168	128	56	0.0000014	1.06667E-06	4.66667E-07
2	14196	8256	1596	0.0001183	0.0000688	0.0000133
3	790412	349632	29316	0.006586767	0.0029136	0.0002443
4	32809322	11017632	396606	0.273411017	0.0918136	0.00330505
5	1083029570	275584032	4216422	9.025246417	2.2965336	0.03513685
6	29614012974	5699195232	36684858	246.7834415	47.4932936	0.30570715
7	6.89902E+11	1.00225E+11	268602258	5749.187384	835.2082536	2.23835215

**Table 2. Cumulative Combinations vs. Combination Size**



**Figure 3. Cumulative Search Time vs. Combination Size**

The National Institute of Standards and Technology (NIST) has compiled a fingerprint database (i.e. Special Database 14) from which Figure 4 and Table 3 were compiled. Special database 14 contains a total of 54,000 individual fingerprints which were collected from the same individuals at two different times. The one group of 27,000 fingerprints are labeled “F” data (e.g. for file) and the other group of 27,000 fingerprints are labeled as “S” data (e.g. for search).



**Figure 4. Fingerprint Classes**

Table 3 describes the number of fingerprints in the NIST Special Database 14 as a function of classification without any references. From this table, it can be shown that the maximum class variance is in the RF class which shows a 2.5% variance.

Class	Units	Class	Units	Class	Units
UL	1680	UL	1772	UL	92
UM	13450	UM	13663	UM	213
UH	2	UH	2	UH	0
RL	256	RL	265	RL	9
RM	538	RM	549	RM	11
RH	1	RH	0	RH	1
PW	5757	PW	5987	PW	230
CW	726	CW	714	CW	12
DW	889	DW	913	DW	24
XW	20	XW	20	XW	0
AA	864	AA	900	AA	36
TT	230	TT	314	TT	230
SR	21	SR	24	SR	3
XX	0	XX	0	XX	0
RF	2566	RF	1877	RF	689
<b>TOTAL</b>	<b>27000</b>	<b>TOTAL</b>	<b>27000</b>	<b>TOTAL</b>	<b>27000</b>

F data                      S data                      Difference

**Table 3**

Using the same fingerprints, NIST has generated a classifier scheme with references. The total number of classifiers is 115 and the “F” data is shown in Table 4 below.

Class	Units	Class	Units	Class	Units	Class	Units	Class	Units
1	166	26	27	51	45	76	3	TT	816
2	374	27	9	52	111	77	0	XX	0
3	456	28	12	53	100	78	0	SR	21
4	519	29	5	54	77	79	0	PI	2513
5	613	30	6	55	53	80	1	CI	503
6	627	31	0	56	43	81	0	DI	665
7	708	32	1	57	33	82	2	XI	9
8	729	33	0	58	38	83	0	PM	1305
9	808	34	0	59	40	84	0	CM	31
10	915	35	1	60	43	85	0	DM	67
11	1069	36	0	61	33	86	0	XM	8
12	1189	37	0	62	37	87	0	PO	2366
13	1195	38	0	63	54	88	0	CO	486
14	1187	39	0	64	40	89	0	DO	364
15	1084	40	0	65	40	90	0	XO	14
16	1050	41	0	66	42	91	0		
17	858	42	0	67	44	92	0		
18	672	43	0	68	30	93	0		
19	496	44	0	69	25	94	0		
20	422	45	0	70	12	95	0		
21	259	46	0	71	14	96	0		
22	191	47	0	72	11	97	0		
23	111	48	0	73	7	98	0		
24	78	49	0	74	4	99	0		
25	37	50	0	75	4	AA	972		

**Table 4. NIST Special Database 14 “F” data**

Computing the variance between the “F” and “S” data for this new classification scheme, the variance largest variance is contained in the PI class and is measured to be 0.4 %. As the number of classifiers is increased the variation is reduced as one would expect.

## **6. Conclusion**

A method for generating biometric cryptographic keys for symmetric cipher systems has been presented. The approach takes advantage of computer processing speeds, biometric data, and standard encryption algorithms to provide a novel way of generating cipher keys without having to remember complicated sequences which might be lost, stolen, or even guessed. In addition, this approach provides some auditing capabilities to minimize first person attacks perpetrated by those users who deliberately generate weak keys. Additional work will be performed to see if fingerprint ridge count between minutiae points or other parameter or classifier may serve as a more stable and unique fingerprint biometric feature.

## **7. Acknowledgement**

I would like to thank Dr. Lance Hoffman and Dr. Ross Micheals for their support, guidance, and encouragement. I would also like to thank Dr. Poorvi Vora, Dr. Daniel Ryan, and Dr. Diane Martin for their valuable participation on my Doctoral Committee.

## **8. References**

- Clancy, T. C., N. Kiyavash and D.J. Lin (2003). "Secure smartcard-based fingerprint authentication." Proceedings ACM SIGMM 2003 Multimedia, Biometrics Methods and Workshop: 45-52.
- Davida G. I., Y. F., and B.J. Matt (1998). "On enabling secure applications through off-line biometric identification." Proceedings of the IEEE Privacy and Security: 148-157.
- Davida G. I., Y. F., B.J. Matt and R. Peralta (1999). "On the relation of error correction and cryptography to an offline biometric based identification scheme." Proceedings Workshop Coding and Cryptography: 129-138.
- Devore, J. I. (1995). Probability and Statistics for Engineering and the Sciences, Duxbury Press.
- Freund, Y., R. Schapire (1996). Experiments with a new boosting algorithm. Proceedings of the 13<sup>th</sup> International Conference of Machine Learning: 148-156.
- IBG (2001). Biometrics Explained, International Biometric Group. **2002**.
- Juels, A., M. Wattenberg (1999). A fuzzy commitment scheme. Proceedings of the 6th ACM Conference of Computer and Communications Security.
- Juels, A. a. M. S. (2002). A fuzzy vault scheme. Proceedings IEEE International Symposium on Information Theory.
- Linnartz, J. a. P. T. (2003). "New shielding functions to enhance privacy and prevent misuse of biometric templates." Proceedings of the 4th International Conference on Audio and Video Based Person Authentication: 393-402.
- Maclin, R., David Opitz (1997). An empirical evaluation of bagging and boosting. Proceedings of the 14<sup>th</sup> National Conference on Artificial Intelligence: 546-551.
- Monrose, F., Michael K. Reiter, Qi Li, Susanne Wetzel (2001). Cryptographic Key Generation from Voice. IEEE Symposium on Security and Privacy.
- Monrose, F., Michael K. Reiter, Susanne Wetzel (1999). "Password hardening based on keystroke dynamics." Proceedings of the 6th ACM Conference of Computer and Communications Security: 73-82.
- Roginsky, A. (2004). A New Method for Generating RSA Keys, International Business Machines Consulting Group.
- Soutar, C., D. Roberge, S.A. Stojanov, R. Gilroy, and B.V.K. Vijaya Kumar (1998). "Biometric encryption using image processing." Proceedings of the SPIE - Optical Security and Counterfeit Deterrence Techniques II **3314**: 178-188.
- Uludag, U., Sharath Pankanti, Salil Prabhakar, Anil Jain (2004). "Biometric Cryptosystems: Issues and Challenges." Proceedings of the IEEE **92**(6): 948-960.