Pre-print of article that appeared at the Biometrics Symposium 2007.

The published article can be accessed from:
http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4430534

# CRACKING FUZZY VAULTS AND BIOMETRIC ENCRYPTION

*Walter J. Scheirer and Terrance E. Boult**

Securics Inc. and University of Colorado at Colorado Springs
Colorado Springs, CO. lastname@vast.uccs.edu or lastname@securics.com

## ABSTRACT

This paper is a security analysis of leading privacy enhanced technologies (PETs) for biometrics including biometric fuzzy vaults (BFV) and biometric encryption (BE). The lack of published attacks, combined with various "proven" security properties has been taken by some as a sign that these technologies are ready for deployment. While some of the existing BFV and BE techniques do have "proven" security properties, those proofs make assumptions that may not, in general, be valid for biometric systems. We briefly review some of the other known attacks against BFV and BE techniques. We introduce three disturbing classes of attacks against PET techniques including attack via record multiplicity, surreptitious key-inversion attack, and novel blended substitution attacks. The paper ends with a discussion of the requirements for an architecture to address *the privacy and security requirements*.

## 1. INTRODUCTION

Biometrics, those unique traits that do not change significantly over a lifetime, present interesting challenges to the security researcher, because of their inherent properties. Unlike familiar cryptographic techniques, biometric data is inexact and can only be approximately matched, hence simple hashing cannot protect it. We also care a great deal about privacy - our biometrics may be used to identify who we are, in circumstances where anonymity is expected. Or, even worse, we may become the victims of identity theft if our biometrics are compromised. The most serious flaw of biometrics, however, is non-revocability. If a biometric is compromised, the user cannot simply generate a new one, as with passwords or PINs. Once a biometric is compromised, it can never be recovered.

Privacy enhancing technologies (PETs) have been introduced to enhance the privacy and security of biometrics, so that they may warrant widespread adoption. Ideally, PETs should make use of as little personal data possible for authentication or validation. Moreover, any personal data used should be protected in such a way that it is infeasible to recover the original data or spoof the user identity.

*Supported in part by NSF STTR #OII-0611283

Biometric template protection is an emerging PETs research area that seeks to compensate for insecurities inherent in biometric security. To protect a template, any transmitted secret must be secured, along with the accompanying biometric data. Recent work [1] has shown that recovering biometric data from templates is possible, thus leading us to a stronger motivation for protection. The critical nature of this area leads one to question the security put in place, as would be done for any traditional cryptographic system. Are the current offerings sufficiently secure for deployment? Unfortunately the answer is "not yet".

Work on attacks against biometric PETs is surprisingly limited. Only a few papers [2] [3], [4], defining attacks against current PETs research exist. These attacks do not span the extent of the weaknesses in current technology.

In this paper, we first define four general classes of attacks in Section 2, including three novel approaches: attack via record multiplicity (ARM), surreptitious key-inversion (SKI) attack, and new types of substitution attacks. We then present an overview of the operation of BFV and BE, followed by our security analysis in Sections 3 and 4. By examining weaknesses in existing systems, we will be able define the requirements for secure biometric systems. We conclude with an enhanced set of requirements for PETs architectures.

## 2. CLASSES OF ATTACK

In order to motivate PETs technology, we first look at attacks against the database in a standard biometric security system. The database is a common target of attack, and certainly a fruitful path of attack against standard biometrics. As depicted in Figure 1, a traditional biometric system will store the original templates in a database, for use in authentication/identification comparisons. If an attacker can gain access to the database (despite its security measures), all template data ($X$) can be compromised.

At the very least, encryption should be utilized to protect enrolled templates in the database, with keys stored by a trusted third party off-site. From a privacy point of view, function creep and owner abuse must be considered - neither of which is reduced by standard encryption. Since decryption is somewhat expensive, and needed on each match, it is likely system operators will often keep many/all records decrypted in memory.

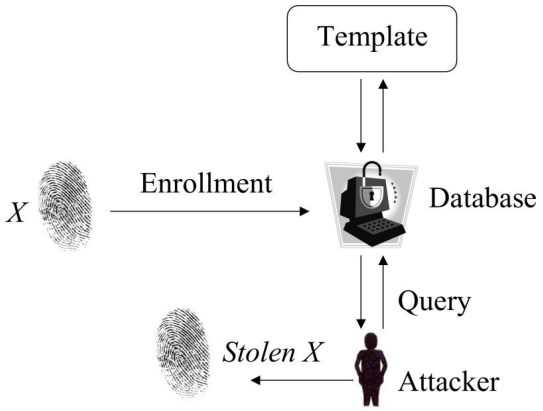


**Fig. 1**. An attacker compromises the database, and is able to retrieve the template $X$

Unfortunately, illicit access to databases with "private" information has become commonplace, with over 150 million financial/personnel records lost in 2006. Even with template encryption, using a worm or virus an attacker might get access to decrypted data in memory or intercept a key intended to unlock an encrypted template, especially given the frequency of use. PETs systems attempt to raise the bar a step further by altering the template before storage in such a way that it cannot be linked back to the user's original biometric data and such that matching or other operations take place in the encoded space. We must presume that an attacker could gain access to the database, and retrieve any data stored there.

### 2.1. Attacks via Record Multiplicity

While many PETs for biometrics have attempted a formal analysis of their security, a significant oversight has been the issue of the risk from attacks that use multiple records. Since to be useful biometric-based tokens need to be used in many independent locations, they need to be non-linkable (to protect privacy) and not subject to combination. But combination issues must be explicitly addressed, and are well known in standard cryptography. A classic "perfect security" model uses a one-time pad. Its security, however, strictly depends on the single use of the pad. Some PETs depend on a random one-time pad model, and ensure the pad is used only once. But it is important to realize that in the one-time pad model, the pad or data are, in a deep sense, interchangeable. In a traditional application of a one-time pad for cryptography, no one would think of sending the same message with multiple different pads, so it is not discussed in that literature. Logically, however, it is the same as using the same pad with many messages, and hence violates the single-use assumptions and hence the perfect security.

Referring to Figure 2, we see multiple enrollments for the same set of biometric template data $X$. Each enrollment has its own secret $\kappa$, resulting in multiple different encodings ($F_1(\kappa_1)$ to $F_n(\kappa_n)$), which are subsequently transmitted and stored by various systems with the same implementation.

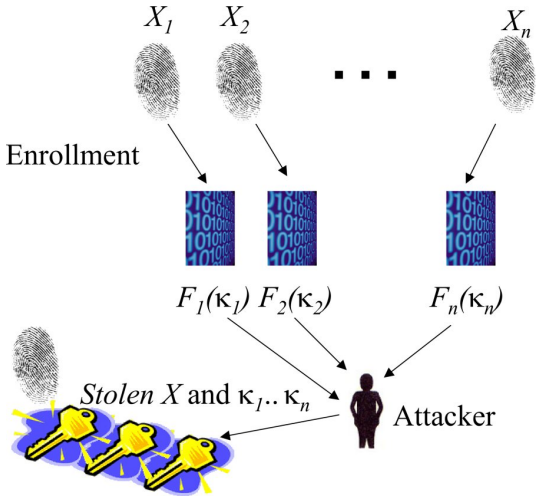


**Fig. 2**. Attack via Record Multiplicity (ARM). An attacker collects multiple enrollment templates, and is able to combine the data and at a minimum link records, and in the most dangerous case can retrieve the template $X$ and the secret $\kappa$

In an Attack via Record Multiplicity (ARM), if an attacker can harvest several of these encodings, it may be possible to correlate the data contained within between encodings to link the databases or, in some cases to directly retrieve $X$ and $\kappa_1 \ldots \kappa_n$. Examples are provided in Sections 3 and 4.

The work of [4], discussed more in Section 3.1, could be viewed as doing a ARM analysis for one class of algorithms – the community needs more of that type of work.

### 2.2. Surreptitious Key-Inversion Attack

In BFV and BE, the stated goal of the system is the release of a secret key. To be useful, this key needs to be used for something, and if it leaves the vault in plain text form opens up a new range of attacks. Even if it only leaves in encrypted form, it opens up a range of insider and system-owner attacks. Figure 3 shows this with encoded data $F(\kappa)$ and an intercepted secret $\kappa$. By knowing $\kappa$, an attacker can decode the biometric template data $X$ by identifying values related to $\kappa$. An example is presented in Section 3.

Scenarios in which an attacker can recover $\kappa$ are not hard to imagine. For example, if $\kappa$ is an ID for user login or a standard cryptographic secret key, it might be possible to intercept it as it is submitted or used in a further portion of an

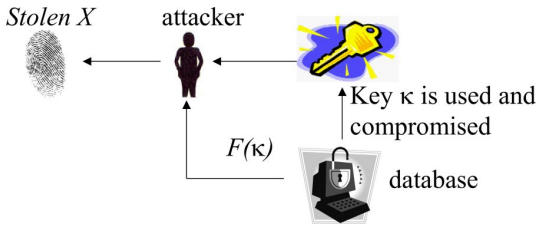


**Fig. 3**. The SKI attack. If the attacker has knowledge of the secret $\kappa$, the template $X$ can be recovered.

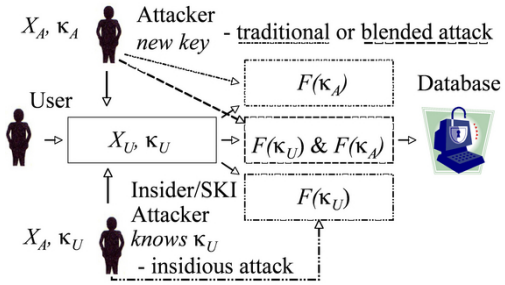


**Fig. 4**. Traditional and blended substitution attacks

authentication or verification system not explicitly integrated with or controlled by the biometric security system (i.e. the operating system), especially by an insider. Even for external attackers, traditional system attacks have exploited unprotected data transmission, unencrypted memory and virtual memory, or have utilized Trojan horse programs to intercept data. PETs, in their most basic form, do not place any requirements on the nature of the secret. If $\kappa$ is known to one or more persons, it could be obtained by an attacker via social engineering - an attack that biometric systems are usually resilient to. Failing to consider asymmetric modes of attack invites trouble; determined attackers will not take the direct approach in their own security evaluations.

### 2.3. Blended Substitution Attacks

In a substitution attack, an attacker alters the contents of a biometric record, with or without knowing something about the record or biometric data. For a general PETs system, Figure 4 shows multiple ways this could occur. The user enrolls their biometric data $X_U$ and a secret $\kappa_U$. Later, the attacker injects another set of biometric data $X_A$ and another secret $\kappa_A$ (or $\kappa_U$ if known) inplace of the user's template. The attacker's data may be directly injected before encoding, or pre-encoded and inserted into the message before (or after) it has been accepted into the database. The database holds only the attacker's data. Digitally signed templates are a partial solution.

In the new *blended substitution*, the user and attacker's data are combined in a single template. If they blend using secret $\kappa_U$ we call it an *insidious blending* as there is no way to detect it is being used. A blended template allows either the user or the attacker to authenticate against the same record. In the traditional substitution case the attacker can authenticate but simultaneously produces a denial of service to the original user, which increases the chance of detection. In the new blended substitution the attacker can use the records simultaneously with the user.

With straight biometrics, e.g. minutiae, blended substitution is not practical since a match with 50% of minutiae not matching would likely be rejected. As we shall see, with biometric PETs, the privacy filter also prevents detection of the blending. Even more alarming, an insidious blended attack can be used as a back-door to biometric authentication systems - by malicious parties or legitimate insiders (recall the Clipper initiative), even with signed templates.

## 3. BIOMETRIC FUZZY VAULTS

In order to solve the problem of hiding a key and unlocking it with biometric or other approximately matching data, Juels and Sudan [5] introduce a construct called a *fuzzy vault*. Inspired by previous work [6] on fuzzy commitment, whereby error-correcting codes were first introduced for approximate matching, fuzzy vaults account for two deficiencies in the fuzzy commitment scheme: intolerance of substantial symbol reordering, and security over non-uniform distributions.

Briefly explained, Alice places a secret $\kappa$ in a fuzzy vault and locks it using a set $A$ of elements from some public universe $U$. To unlock the vault, and retrieve $\kappa$, Bob must present a set $B$ that substantially overlaps with $A$. Fuzzy vaults are order invariant, meaning $A$ and $B$ may be arranged in any order. To protect $\kappa$, it is represented as a polynomial $p$, specifically encoded in the coefficients. A set of points $R$ is constructed from $A$ and $p(A)$. In addition to these points, chaff points $C$ are randomly generated and inserted into $R$. [5] solves the subset matching problem with Reed-Solomon coding. To decode $\kappa$, if Bob's $B$ approximately matches $A$, he can isolate enough points in $R$ that lie on $p$ so that applying the error correcting code he can reconstruct $p$, and hence $\kappa$.

As an example, assume Alice has chosen the polynomial $p(x) = -5x^2 + 2x - 1$. The coefficients (-5, 2, -1) encode the secret $\kappa$. With an unordered set of data $A = \{2, 1, 4, -3\}$, Alice will obtain the polynomial projection: $\{(A, p(A))\} = \{(2, -17), (1, -4), (4, -73), (-3, 38)\}$. $N$ chaff points $C = \{(-1, 14), (-5, 22), ...\}$ that do not lie on $p$ are randomly generated. These chaff points are added to the polynomial projection, and all of the points are shuffled randomly. If Bob can accurately isolate at least 3 points that lie on $p$, he will be able to reconstruct $p$, and recover $\kappa$ encoded as (-5, 2, -1). Without at least 3 points, Bob will not be able to recover $\kappa$.

In [7], fuzzy vaults are applied to secure fingerprint templates. To generate the sets $A$ and $B$, the $x$ and $y$ coordinates of fingerprint minutiae points are used. $\kappa$ is secured via the fingerprint minutiae, and can only be recovered if the minutiae set $A \approx B$. It is important to note that in this scheme, the vault is separate from any back-end application (the use of $\kappa$ is decoupled from the vault) - it is only responsible for securing $\kappa$ with the fingerprint data. The implementation of [7] creates $\kappa$ as a 128-bit random stream, which is analogous to an AES symmetric key. Instead of the error correcting code of [5], a 16-bit CRC is used for set comparison. Both the CRC and $\kappa$ bits are concatenated into a 144-bit representation, that is encoded into an 8th degree polynomial.

Enhancements to fingerprint-based fuzzy vaults are made in [8] to use minutiae location and orientation. This allows more chaff points to be included during encoding, and allowing faster identification of chaff points during decoding. Further, a minutiae matcher is used during decoding, to im-

prove the decoding process. Selecting more reliable minutiae is done by the use of local quality measures.

The basic idea of fingerprint fuzzy vaults to include *helper data* to aid in alignment was considered in [8] and [9]. Alignment is a serious problem for fuzzy vault fingerprint systems. A submitted image might be affected by translation, rotation, and non-linear distortion, causing a false-rejection during comparison. Helper data is used as input to alignment algorithms, but must not leak any information about the minutiae. [8] and [9] align the query fingerprint with respect to the template via an Iterative Closest Point algorithm.

In [10] a similar problem/approach for biometric authentication is developed with some general results when the distribution of the original data is known and the system does not leak data other than the single template + helper. In [11], they analyzed that model and present some practical results. The paper, however, lacks sufficient details to actually analyze the security of the helper functions, or the distributions. The attacks presented are expected to apply to these as well.

### 3.1. Previous Attacks Against Biometric Fuzzy Vaults

Chang et al. [3] have investigated chaff identification, that is in accordance with the second observation of non-randomness in fuzzy vault schemes noted in [2]. The main idea of the attack is that chaff points generated later in the process tend to have a smaller *free area*. The notion of free area refers to the amount of neighboring points in $R = (X \cup C)$. A point with a smaller free area has more neighboring points than a point with a larger free area. The authors [3] are unable to prove this analytically, but do attempt to establish it empirically.

In [4], Boyen takes the definitions of a fuzzy extractor as introduced by [12], which is theoretically an improvement over [5] to leak less information. Using Dodis's definition, Boyen constructs a counterexample fuzzy extractor which is provably insecure if the same noisy secret is reused a few times. We categorize that work as an attack on the definitions put forth in [12] rather than on an actual approach. They propose two security models that specifically address the case of fuzzy secret reuse, respectively from an outsider and an insider perspective, in what they call a chosen perturbation attack. No actual biometric fuzzy vault approaches are discussed/attacked in [4].

### 3.2. New Attacks against Biometric Fuzzy Vaults

Biometric fuzzy vaults are particularly vulnerable to ARM attacks, and we present a few simple examples. Consider a BFV scheme [7] [8] with $m$ sets of minutiae, $(x[i], y[i], t[i], p[i])$, where $0 \leq i \leq m$, and where the real minutiae sets are mixed with $n$ lines of random chaff, $(u, v, w, z)$. During enrollment, we would assume the database would bind different keys via the helper data/polynomials coefficients $p[i]$, such that given a matching print they extract the key with either using error correcting codes or a separate CRC to test subsets and handle a few errors. Without a matching print, the key and original data are reasonably well protected since guessing the subset of real data, without a matching print requires significant effort, even considering Chang et al. [3].

However, given two or more such BFV instances generated from the same print, but with different keys and different random chaff, the minutiae are likely recoverable by matching the two templates. Even with the chaff, each of minutiae+chaff needs to be matchable against the live minutiae. It is possible that some chaff will randomly match. However, the error handling mechanism (CRC or ECC) is inherently designed to address that problem. So while one such template may be securely protected, given access to two, they can be easily matched and directly attacked.

We are unaware of any previous work describing attacks on fuzzy vaults given knowledge of the released key, i.e. a SKI attack, though some might consider it obvious. If an attacker is able to recover the secret $\kappa$ through means other than an attack against the template, it becomes trivial to recover the biometric data within the currently known biometric fuzzy vaults. From $\kappa$, the polynomial $p$ is directly reconstructed. Once $p$ is known, $R$ may be directly enumerated to separate the biometric data, in the form $(A, p(A))$, from the chaff which by design is not on the polynomial. Acidential chaff inclusion is handeled by the correction mechanism.

A blended substitution attack against BFVs is forthright. Most of the vault is chaff, so the attacker can overwrite chaff lines with a minutiae set $(x_A[i], y_A[i], t_A[i], p_A[i])$, encoding $X_A$ and $\kappa_A$. The resulting template will contain data for both the legitimate user and the attacker. They could either release a different key (but still be associated with the record - the backdoor approach), or with knowledge of the original key can also do an insidious blending and inject their data with the true polynomial. Given the high volume of chaff used, e.g. [7] has 200 chaff lines to 18 real minutiae, there is capacity to have many blended substitutions within one biometric fuzzy vault.

## 4. BIOMETRIC ENCRYPTION

Biometric Encryption (BE) [13], is a particular approach but there are, however, many related techniques which will be discussed below. Like BFVs, BE attempts to provide a way to retrieve a key using a biometric. The fundamental principle underlying BE is correlation, with security provided by using a random-phase correlation "pad". They provide a formal reduction to one-time pads. At the core of the BE is a phase-only template which is the point-wise product $H_{s_{i,j}}(u) = e^{-i\phi_{A_i}(u)} \cdot e^{-i\phi_{R_j}(u)}$, where $R_j$ is the random pad used in enrollment $j$ of user $i$, and $H_{s_{i,j}}(u)$ is the stored template for user $i$ at enrollment $j$.

Bringing this into the realm of biometric identification and authentication, to satisfy the aforementioned requirements, correlation is the process used to link and retrieve secret keys. Instead of returning a single scalar value indicating similarity, BE returns a 128-bit key. The function $H_{s_{i,j}}$ is designed to

produce a more complex output pattern than a standard correlation. During verification an input is converted into the Fourier domain, point-wise multiplied by the stored mask to produce a result $c(u)$. A "link table" is defined to determine output the key, by listing multiple locations from $c(u)$ that when thresholded, produce either a 0 or 1. Multiple (e.g. 5) entries are combined for each output bit with a majority vote.

For enrollment, BE builds a token composed of $H(u)$, a cryptographic key $k_0$ linked with $c_0(x)$, and an identification code $id_0$ generated by encrypting $k_0$ and generating a one-way hash. For verification, the token input image is used to produce $c_1(x)$, which, via the link table, generates a key $k_1$. If this key when used in hashing the beginning of H(y) matches $id_0$, the key is released.

In [14] a related approach was presented for face-based protection, except it used a MACE filter rather than standard correlation, and it did not include the link-table component to release the key. The security again depends on the reduction to the one-time pad. Also related is [15] and other works by that group, that proposed similar threshold-based + random pad methods.

### 4.1. Previous Attacks Against Biometric Encryption

Adler [2] uses a "hill-climbing" attack to successfully compromise a BE scheme [16] applied to facial recognition. Hill-climbing involves the iterative modification of a test image till it matches an enrolled image. Adler uses the consistency of a generated "link-table" as his metric, and used it, even with quantization, to produce a successful attack. Even if the results were not visually a great match to the input, the results provide for at worst a masquerade attack, if not a full compromise. How could hill-climbing work against something with "perfect security"? One answer is that the link-table constraints, not part of a standard one-time pad, leak information. Modifications in each iteration are made based on information leaked during the comparison.

Interestingly, Adler makes the claim that the hill-climbing attack is also possible against fuzzy vaults, despite the proof of security presented in [5]. The first observation is that the proof of security assumes the data held in the fuzzy vault are random. When applied to biometrics, this assumed property is violated since biometric data is inherently structured. The second observation is that chaff placement is careful, in order to space chaff points far away from legitimate points. This placement scheme strays from the randomness assumption. No demonstration of a successful attack, however, is presented.

In [17] Teoh et. al. show that when the tokenized random numbers are obtained by an attacker, the performance and protection of [15], becomes unacceptable.

### 4.2. New Attacks Against Biometric Encryption

If the attacker knows the key, a SKI-enhanced hill-climbing attack becomes possible. Adler's attack used "consistency" as a measure during the climb. When the attacker knows the expected result, they know the link-table constraints on the data and hence can construct more accurate approximations of the original biometric data.

Though it has claimed security, we show BE is susceptible to multiple types of substitution attacks, though not quite as simply as the straightforward manner applied to fuzzy vaults. We have found four possible attack routes. The first presumes that an attacker has knowledge of the key $k_0$ and a traditional substitution attack. Given the key the attacker simply generates an enrollment for themselves with their filter function $H_A(u)$, link table and ID. They can simply "add" this as a new enrollment and the output identification code remains the same, as $k_0$ has not changed. Thus it is a subsitution without denial of service and therefore less detectable.

The blended substitution attack requires more knowledge. It is easiest if attacker has knowledge of the user's biometric and the key (e.g. from SKI attack). In this case, the attacker can craft a filter function $H_{AU}(u)$ that is able to generate a correlation function $c_{AU}(u)$ based on the user's and the attacker's biometric data and a link table that chooses consistent bits for both the user and the attacker to output the key. The many levels of freedom in the filter and link-table design process makes this straightforward, though it is likely the added constraints will increase the change of false-rejection. If the attacker has access to the biometric but not the key, a third approach is to generate a new enrollment for the combined $H_{AU}(u)$ and choose a new key and generate the proper hash $id_0$ data. Since BE does not require any type of mutual authentication with external storage of the key this blended substitution with a new key would fail on data protected with the original key, but for any new usage the blended substitution would be accepted for either attacker or user. If an insider were able to do this during or soon after enrollment it would be virtually undetectable.

Other blended substitution attacks combine the use of hill-climbing [2] to estimate a masquerade image $M$ and key, which are then used as stand-ins for the known biometric/key in the above attacks.

With respect to ARM attacks, BE is clearly subject to correlation of the templates. While they have different random pads all user records have the same phase information for the user, so by using complex conjugates we can consider testing two templates of the same user yielding $e^{-i\phi_{A_u}} \cdot e^{-i\phi_{R_j}} e^{i\phi_{A_u}} \cdot e^{i\phi_{R_k}} = e^{-i\phi_{R_j}} \cdot e^{i\phi_{R_k}}$ while for templates of different users we get $e^{-i\phi_{A_u}} \cdot e^{-i\phi_{R_j}} e^{i\phi_{A_i}} \cdot e^{i\phi_{R_k}}$ which does not reduce and has lower magnitude since the user and imposter have different phase information.

With respect to an ARM-based reconstruction, let us presume an advisory gains access to $H_{s_{i,j}}(u)$ for $j = 1..N$. Looking at the definitions, this is simply multiple samples of the user's phase data, $\phi_{A_u}$, corrupted by random phase noise, i.e. a standard signal estimation problem given an accurate noise model. Despite the claims in [13] about phase-only data

improving security, they offered no proofs. It turns out that once the phase data is estimated, the techniques of [18] show how to provide an estimation of the full signal from its phase-only part.

## 5. CONCLUSION

In this paper, we have introduced three new classes of attacks against biometric fuzzy vaults (BFVs) and biometric encryption (BE): record multiplicity (ARM), surreptitious key-inversion (SKI) attack, and blended substitution attacks. BFVs are easily compromised by all three attacks. Biometric encryption is impacted by SKI via improved hill climbing and is compromised by the ARM and substitution attacks - but with more effort. Regardless, the security limitations are strong enough for us to conclude that BFVs and BE are not suitable for securing biometric systems or for protecting privacy.

Based on what we have learned by studying the weaknesses of biometrics and PETs architectures, and through our own work [19] designing PETs , we have defined the following requirements for secure biometric PETs architectures:

1. No combination of data from multiple enrollments by the same individual should be able to be combined to recover the biometric template data or to generate a spoof.
2. If any non-biometric data that is used to encode/decode (e.g. link table), or is released by the system (e.g. key), is known, the biometric template must not be recoverable nor should it allow hill-climbing or spoof generation.
3. It should not be possible for two users to authenticate against the same token with significantly higher frequency than the system's documented False Accept Rate.
4. No undetected substitution of records should be possible.
5. Any data transmitted outside the system, except during enrollment, should not be suitable to link the underlying user over space/time/companies.

It is our hope that these new requirements will be adopted by researchers working in this space to improve the security and privacy of biometric PETs.

## 6. REFERENCES


[1] A. Ross, J. Shah, and A. Jain, "From Template to Image: Reconstructing Fingerprints from Minutiae Points," *IEEE Transactions of Pattern Analysis and Machine Translation*, vol. 29, pp. 544–560, 2007.

[2] A. Adler, "Vulnerabilities in Biometric Encryption Systems," in *AVBPA 2005, Audio- and Video-Based Biometric Person Authentication*, 2005, pp. 1100–1109.

[3] W. Chang, R. Shen, and F. W. Teo, "Finding the Original Point Set Hidden Among Chaff," in *ASIACCS '06: Proc ACM Symp Information, Computer And Communications Security*. 2006, pp. 182–188, ACM.

[4] X. Boyen, "Reusable Cryptographic Fuzzy Extractors," in *ACM Conf. on Computer and Communications Security*, 2004, pp. 82–91.

[5] A. Juels and M. Sudan, "A Fuzzy Vault Scheme," in *Proc. IEEE Int. Symp. on Information Theory*, 2002, p. 408.

[6] A. Juels and M. Wattenberg, "A Fuzzy Commitment Scheme," in *Sixth ACM Conference on Computer and Communications Security*, 1999, pp. 28–36.

[7] U. Uludag, S. Pankanti, and A. Jain, "Fuzzy Vault for Fingerprints," in *AVBPA 2005, Audio- and Video-Based Biometric Person Authentication*, 2005, pp. 310–319.

[8] U. Uludag and A. Jain, "Securing Fingerprint Template: Fuzzy Vault with Helper Data," in *Proc. IEEE Workshop on Privacy Research In Vision (PRIV)*, 2006.

[9] K. Nandakumar and A. Jain, "Fingerprint-based Fuzzy Vault: Implementation and Performance," *Tech Report MSU-CSE-06-31, Dept of Comp. Sci., Michigan St. Univ.*, 2006.

[10] P. Tuyls and J. Goseling, "Capacity and Examples of Template-Protecting Biometric Authentication Systems," in *Biometric Authentication Systems. Biometric Authentication Workshop*, 2004.

[11] P. Tuyls, A. Akkermans, T. Kevenaar, G. Schrijen, A. Bazen, and R. Veldhuis, "Practical Biometric Authentication with Template Protection," in *AVBPA 2003*, 2005, pp. 436–446.

[12] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy Extractors: How to Generate Strong Keys From Biometrics and Other Noisy Data.," in *Proc. Advances in Cryptology Eurocrypt*, 2004.

[13] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, and V. Kumar, "Biometric Encryption," in *Chapter 22 of the ICSA Guide to Cryptography*, R.K. Nichols, Ed. 1999, McGraw-Hill.

[14] M. Savvides, B. V. Kumar, and P. Khosla, "Cancelable Biometric Filters for Face Recognition," in *Proc. Int. Conf. on Pattern Recognition*, 2004, pp. 922–925.

[15] T. Connie, A.B. Teoh, M. Goh, and D.C. Ngo, "Palmhashing: A Novel Approach for Cancelable Biometrics," *Info. Proc. Let.*, vol. 93, pp. 614–634, 2005.

[16] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, and B. Vijaya, "Biometric Encryption Using Image Processing," in *Proc of the SPIE Int. Soc. Opt. Eng.*, 1998, vol. 3314.

[17] A. B. Teoh and D. C. Ngo, "Cancellable Biometerics Featuring with Tokenized Random Number," *Pat. Rec. Let.*, pp. 1454–1460, 2005.

[18] H. Pozidis, "Signal Reconstruction From Phase Only Information And Application To Blind System Estimation," in *ICASSP*. 1997, IEEE.

[19] T. Boult, W. Scheirer, and R. Woodworth, "Secure Revocable Finger Biotokens," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.