# Privacy Preserving Multi-Factor Authentication with Biometrics

Abhilasha Bhargav-Spantzel
CERIAS
Purdue University
West Lafayette, IN
bhargav@cs.purdue.edu

Anna Squicciarini
Purdue University
West Lafayette, IN
squiccia@cs.purdue.edu

Elisa Bertino
Purdue University
West Lafayette, IN
bertino@cs.purdue.edu

## ABSTRACT

An emerging approach to the problem of reducing the identity theft is represented by the adoption of biometric authentication systems. Such systems however present however several challenges, related to privacy, reliability, security of the biometric data. Inter-operability is also required among the devices used for the authentication. Moreover, very often biometric authentication in itself is not sufficient as a conclusive proof of identity and has to be complemented with multiple other proofs of identity like passwords, SSN, or other user identifiers. Multi-factor authentication mechanisms are thus required to enforce strong authentication based on the biometric and identifiers of other nature.

In this paper we provide a two-phase authentication mechanism for federated identity management systems. The first phase consists of a two-factor biometric authentication based on zero knowledge proofs. We employ techniques from vector-space model to generate cryptographic biometric keys. These keys are kept secret, thus preserving the confidentiality of the biometric data, and at the same time exploit the advantages of a biometric authentication. The second authentication combines several authentication factors in conjunction with the biometric to provide a strong authentication. A key advantage of our approach is that any unanticipated combination of factors can be used. Such authentication system leverages the information of the user that are available from the federated identity management system.

## Categories and Subject Descriptors

K.6.5 [**Computing Milieux**]: Management of Computing Information Systems—*Security and protection*

## General Terms

Design, Security

## Keywords

Identity Theft Prevention, Privacy, Biometrics, Authentication


*DIM'06,* November 3, 2006, Alexandria, Virginia, USA.


## 1. INTRODUCTION

The problem of identity theft, that is, the act of impersonating others' identities by presenting stolen identifiers or proofs of identities, has been receiving increasing attention because of its high financial and social costs. Recent federated digital identity management systems if on one side have improved the management of identity information and user convenience, on the other side do not provide specific solutions to address identity theft. One approach to such problem is the adoption of biometric *identification* and *authentication* systems. These systems are automated methods for recognizing an individual based on some physical characteristics, such as fingerprints, voice, or facial features.

Biometric identification and authentication are differentiated as follows. Biometric identification occurs when an individual provides a sample biometric, sometimes without any additional knowledge, and the system must compare that sample with every stored record to identify a match. This is known as a one-to-many match, and is executed without any corroborating data. By contrast, biometric authentication occurs when an individual presents a biometric sample, and some additional identifying data, such as a photograph or password, which is then compared with the stored sample for that individual. Biometric authentication provides some inherent advantages as compared to other non-biometric identifiers since biometrics correspond to a direct evidence of the personal identity versus possession of secrets which can be potentially stolen. Moreover, most of the times biometric enrollment is executed in-person and in controlled environments making it very reliable for future use.

**Challenges in Biometric Authentication.** Biometric authentication poses however several non-trivial security challenges because of the inherent features of the biometric data itself. Addressing these challenges is crucial for the large scale adoption of biometric authentication and its integration with other authentication techniques and with access control systems.

Biometric matching is probabilistic in nature, which implies that two samples of the same individual are never exactly the same. If the two samples are encrypted for security reasons, they need to be decrypted before they can be matched. This raises the issue of key management to enable decryption, and also represents a point of vulnerability in the process. Moreover, it is very hard to revoke and change biometrics in case biometric data are compromised. At the time of enrollment or verification the individuals biometric is read as a template, that is, is a binary file created using distinctive information from a biometric sample, which is then stored in a database or on a token. These templates are often vendor-specific and therefore the interoperable use of such templates in a distributed system is very difficult if at all possible.

Biometric authentication from a remote location also represents a difficult issue because of the risk of spoofing attacks. The credibility of the output from a biometric matching process depends entirely on the integrity of the sample provided, and whether it was provided by the true owner of the biometric. Older generation biometric capture devices were vulnerable to spoofing attacks, and there is extensive work currently in the area of biometric capture devices to able to withstand different spoofing attacks.

Biometric authentication can be implemented through systems performing the matching either on the *server* or on the *client side*. Depending on whether the matching of the biometric template is executed - at the server or at the client - different security problems arise. In the former case the main issues are related with the large scale and distributed management of biometric templates. The creation of a database of a particular biometric at the server should itself be secure and possibly decentralized. Also, such database would be highly dependent on a particular software or hardware and thus could not be interoperable. Such a system is also CPU-intensive because of the matching operations.

Additionally, storing biometric information in repositories along with other personally identifiable information raises several security and privacy risks [1]. These databases are vulnerable to attacks by insiders or external adversaries and may be searched or used outside of their intended purposes. It is important to note that if the stored biometric identifiers of an individual are compromised, there will be severe consequences for the individual because of the lack of revocation mechanisms for biometrics.

Due to the security and privacy problems of server side matching, there have been several efforts in biometric authentication technology using client side matching [15, 16]. Such an approach is convenient as it is relatively simple and cheap to build biometric authentication systems supporting biometric storage at the client end able to support local matching. Nevertheless, systems of such type are not secure if the client device is not trusted; therefore additional cryptographic support is needed.

Several efforts have been undertaken to strengthen client side authentication. Previous approaches [3, 12] have been developed based on Chaum and Pederson wallet-with-observer paradigm [5]. An interesting approach recently proposed focuses on key extraction from biometrics which entails the problem of "approximate equality" in biometric comparisons. Several approaches have been proposed for overcoming this difficulty, including the use of error-correcting codes [9], fuzzy commitments and fuzzy vaults [15, 16] and fuzzy extractors [10]. However, several of these schemes may be vulnerable to replay attacks, non-repudiation and to cryptanalysis.

Client side authentication systems also led to research in key generation mechanisms using biometrics [26, 10, 18]. Key generation is executed by first extracting the biometric features from the biometric data based on the feature extraction module of the biometric authentication system. Then, the biometric features are sent to the system specific key-generation module to generate a key, that we refer to as *bio-key*. The challenge in such research direction is to devise algorithms for reliable key generation. Such key generation algorithms must be able to generate the same key despite the noise in biometric readings. Moreover, the semantics of the usage of such a key should still retain the property of *"what you are"* versus *"what you have"*.

**Desiderata.** Based on the previous discussion we identify several crucial properties of a suitable biometric authentication system. The system must:

- preserve the privacy of biometric data;
- be convenient to use and interoperable with different authentication servers thus providing scalability.
- be able to perform client-side matching without requiring tamperproof or trusted hardware;
- support revocation of the biometric identifiers;
- be resilient to the compromise of the biometric template itself;
- be resilient to replay attacks so that the replay of the biometric signal or the key generated based on the biometric cannot result in successful authentication;
- provide security for any cryptographic token associated with the biometric and efficiently manage keys;
- provide non-repudiation and accountability.

**Our approach.** In this paper we cast the problem of biometric authentication in the specific context of federated identity systems [11, 13, 23], which typically rely on attributes and properties of the member users to enforce authentication. Our main objective is to achieve a privacy preserving methodology, in which use of credentials and biometric is completed without loss or exposure of additional data. The first problem we have to deal with in our effort toward a methodology for biometric authentication in federated systems is related with interoperability. If on one hand federated digital identity systems need to support data heterogeneity, on the other hand, biometric vendors typically generate proprietary templates which are not interoperable among each other. Obviously, this represents a major limitation to the large scale deployment of a federated system. In order to address this problem in a privacy preserving fashion we develop a cryptographic key generation algorithm for use at the client side, which can thenceforth be used interoperably with other clients in the federation. Precisely, in this paper we use the mechanisms from vector space modelling [24] to generate cryptographic bio-keys.

To further preserve privacy we provide authentication protocols based on well known techniques called zero knowledge proof of knowledge (ZKPK's for brevity) [2, 4]. ZKPK's allow a user to have a private secret, and prove its possession without releasing it. As such, bio-keys are never released but are instead used to generate a *proof* of the ownership of the biometric. This proof is sufficient for the purposes of authentication as it would correspond to the biometric enrolled in the system. The use of ZKPK proof enables us to have a *two-factor* authentication by using information theoretically secure Pedersens commitments [19]. Such commitments also elegantly handle revocation of the generated bio-keys.

We also show how the two-factors can be combined with other identity information available in the federation to provide *multi-factor authentication*. Providing proof of the biometric identifier itself is not sufficient as the proof of knowledge of other sensitive identifiers like social security number (SSN) or the credit card number (CCN) can be required to complete the authentication procedure. The following example introduces a scenario which we will use to illustrate the authentication phases.

EXAMPLE 1. *Consider a federation including a Bank $CityBank$, and a Tax Authority $TaxAuthr$. $CityBank$ is the local bank for the user $Alice$ and contains all financial information concerning $Alice$. She also enrolls other information with the bank which can be potentially used at the time of authentication. $CityBank$ is essentially Alice's local identity provider.*

*Alice wants to fill her tax on line with $TaxAuthr$. However $TaxAuthr$ requires its on-line users to authenticate using two-factors biometric authentication to access such service. Further, if Alice wants to do money transactions, then $TaxAuthr$ requires her to perform multi-factor authentication by providing proof of ownership of a registered 1) biometric 2) CCN and 3) verified SSN. Thus depending on the service requested, Alice would need either need to perform two-factor or multi-factor authentication.*

The main contributions of our work can be summarized as follows. We describe a novel method for key-generation using vector-space modeling. This is a generic methodology to generate bio-key where the vectors used could correspond to any defined combination of one or more biometrics. As compared to existing bio-key generation work, we differ in how the bio-keys are actually used. The actual key is never revealed as it is possibly could leak further information of the individual. We therefore use it for zero knowledge proof of knowledge which directly provides for unlinkability and replay avoidance. We provide a two-phase authentication mechanism for federated identity management systems. The first phase consists of a two-factor authentication with biometric data, and the second of a multi-factor authentication with other user attributes. We show how we can use our protocols to secure biometric data itself thus preventing its fraudulent use that would result in identity theft and other security breeches. In addition, we show how privacy preserving multi-factor authentication can be enforced in federated identity management systems with the ability to use biometric data just like the other identifying attributes of the user. Our approach is privacy preserving in that all the authentication steps are done with limited disclosure of data that cannot be used for any other purpose other than the authentication decision itself. The paper is organized as follows. In Section 2 we provide basic background information regarding biometric authentication followed by Section 3 where we provide a brief review of authentication in federated identity management systems. Then in Section 4 we provide our key protocols required for biometric authentication. In particular in Section 4.1 we elaborate on our approach towards biometric key generation, followed by Section 4.2 where we show how it can be used in zero-knowledge proof of knowledge. In Section 5 we show how we can provide strong authentication using multi-factor. Finally, we provide a detailed analysis of our approach in Section 6 and then conclude.

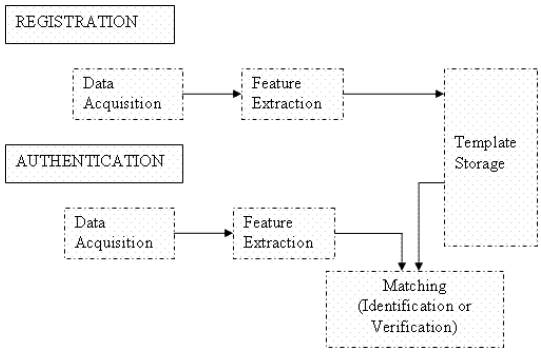


**Figure 1: A Generic Biometric System.**

## 2. BIOMETRIC AUTHENTICATION BACKGROUND

Biometrics adds a new type of authentication which, unlike from conventional approaches, is not based on what an individual knows or possesses, but on some characteristics of the individual itself. We elaborate on the main concepts related to biometric in this section.

A typical biometric system model consists of *a capture device, a feature extraction unit, a comparison algorithm*, and *a storage device*. The capture device captures the raw sample provided by the individual. The feature extraction unit processes the raw sample to extract the relevant information, also called *features*, which can then be used in the comparison process. The comparison algorithm will match two processed biometric samples, and give as output a similarity score. The storage device will store the *templates* created during the enrollment process. A template is data, which represents the biometric measurement of an individual, used by a biometric system directly or indirectly for comparison against other biometric samples.

A biometric system typically supports two sub-processes: registration (also called enrollment), and authentication (see Figure 1).

**Enrollment**: It is the process of capturing the features from a biometric sample provided by an individual and converting it into a template. The effectiveness of enrollment strictly depends on the quality of the data submitted along with the biometric. Thus, the enrollment process has also to ensure that the verification documents (like passports and drivers licenses) are trustworthy so that a fake or false identity is not linked to a biometric. Additionally, no duplicate records have to be stored in the database for the same identity. Such enrollment mechanism is a key aspect of biometric authentication making it very reliable. Enrollment is the first interaction of the user with the biometric system, and misuses of such operation can affect the quality of sample being provided by the user, which in turn affects the overall performance of the system. An uncomfortable first experience could affect later interactions of the user with the system, thereby affecting the overall system performance.

Once the process of registration is successfully completed, the individual can use the biometric system for authentication.

**Authentication** Biometric authentication is performed when the individual presents his/her biometric sample along with some other identifier which uniquely ties a template with that individual. The matching process is performed against only that template. This is a one-to-one matching process.

## 3. AUTHENTICATION IN FEDERATED IDENTITY MANAGEMENT SYSTEM

Digital identity can be defined as the digital representation of the information known about a specific individual or organization. As such it encompasses, not only login names, but many additional information, referred to as *identity attributes* or *identifiers*, about users. Managing identity attributes raises a number of challenges, due to conflicting requirements. On the one hand, identifiers need to be shared to speed up and facilitate authentication of users and access control. On the other hand, identity attributes need to be protected as they may convey sensitive information about an individual and can be targets of attacks like *identity theft*. In cyberspace preventing identity theft is especially hard because digital information can be copied hence stolen unnoticed. Identity of an individual can be represented through different types and combinations of identifiers. Therefore when reasoning about authentication it is important to consider a combination of multiple identifiers. Such an authentication mechanism is called *multi-factor authentication* and is a prevalent mechanism to mitigate the threat of identity theft.

| | | Alice@Registrar1 | PARAMS |
|---|---|---|---|
| **Strong IdTag** | **Commit-ment [M]** | **assurance** | **WeakID (list)** |
| **CCN** | 329839797987<br>223493827983 | good | Value: Alice, Mars; tag: fname, lname; assure: B, B |
| **SSN** | 398723987479<br>232738294991 | undecided | Value: Alice, 12442; tag: fname, zip; assure: A, B |
| **FINGER PRINT** | 729874666210<br>047937477211 | good | Value: Cap-bio, 80; tag: sensor, threshold; assure: A, A |

**Figure 2: Identity Record Example**

An emerging approach to address issues related to identity management is based on the notion of *federations* [11, 13, 23]. The goal of federations is to provide users with protected environments to federate identities by the proper management of identity attributes. Federations provide a controlled method by which federation members can provide more integrated and complete services to a qualified group of individuals within certain sets of business transactions. By controlling the scope of access to participating sites, by enabling secure, cross-domain transmission of users personal information, federations can make more difficult the perpetration of identity frauds, as well as their frequency, and the potential impact of these frauds.

Federations are usually composed by two main entities: identity providers (IdPs), managing identities of individuals, and service providers (SPs), offering services to registered individuals. In a typical federated identity management system the individual registers with his/her local IdP and is assigned a username and password. Registration is usually based on an in-person verification at some registration office. Based on this information a registered individual can submit additional attributes and its corresponding attribute release policies, which are stored at the local IdP. The IdP is then contacted whenever the user interacts with any other SP in the federation when additional user information is needed. The IdP is in charge of sending the SP the submitted user attributes in accordance to the attribute release policies. In [2] a third type of entity was introduced referred to as *Registrar* which essentially captures the notion of proofs of identity for static user attributes like SSN and CCN. These proofs are based on ZKPK protocols and Section 4 describes a modified version of the ZKPK which is specific for biometrics. Like the biometric system, the proposed IdM system also supports two main phases, namely enrollment and authentication. At the time of enrollment the zero knowledge commitments of the strong identifiers[1] are recorded with additional meta information about the state of the identifier. This record, referred to *identity record*, is stored under the single sign-on identifier of the individual. An example of an identity record (IdR) is shown in Figure 2. Typically the identifiers recorded in an identity record correspond to *what you have*, like Credit Card Number (CCN) and Social Security Number (SSN). We show how we can use identifiers corresponding to *who you are* in such an identity record, as illustrated in the last row of Figure 2 with the tag "finger print".

Our goal is to use biometrics in combination with the other committed identifiers in the identity record at the time of authentication. As such, at the time of authentication, multiple commitments

[1]Strong identifiers are those that uniquely identify an individual. This is also known has personally identifiable information.

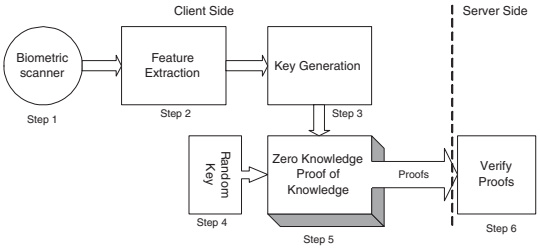


**Figure 3: Flow representing biometric authentication using zero knowledge proofs**

of strong identifiers and biometric commitments can be combined in an ad-hoc fashion, and verified by using exactly the same approach developed for the case on non-biometric data. This mechanism would prove the knowledge and possession of such proven identifiers. In the next section we illustrate how biometric readings can be used so that the extracted information can be used across the federation, thus achieving interoperability.

# 4. BIOMETRIC AUTHENTICATION

The main steps in the proposed biometric authentication are shown in Figure 3. Here the biometric is read at step 1, followed by feature extraction and bio-key generation at step 3. It is non trivial to use a freshly recorded biometric reading as the secret key, used in step 5 in the ZKPK. This is because every time particular biometric is read, the resulting template may be substantially different. Therefore reducing a biometric reading to a unique secret key poses several challenges [26, 18]. Using well known error correcting codes [9] on the fingerprint templates itself is not straightforward because the digital encoding of the two templates vary substantially if only the bit pattern is considered. Moreover, key generation differs from template matching since while re-generating the key the original fingerprint template is not available. We therefore investigate a technique based on fingerprint classification to define a characteristic vector for a fingerprint which is subsequently used by vector space modeling (VSM) [24] methodologies to retrieve a unique key.

## 4.1 Key Generation using Biometric Templates

Fingerprint classification can be taken as a special case of pattern classification techniques which aim at reducing the computational overhead of pattern matching. If biometric patterns can be categorized, then given a pattern it may be possible to match the input pattern only against the stored templates in the same category of the input pattern. This is sometimes referred to as "binning". It has been shown in [14] that fingerprints can be classified according to three different dimensions: by the shapes and contours of individual patterns, by the finger positions of the pattern types, and by relative size, determined by counting the ridges in loops and by tracing the ridges in whorls. The resulting information is coded according to a concise representation, referred to as *characteristic vector*.

Each element of the characteristic vector is weighted with respect to the importance of that characteristic. For example, a common approach to classify a fingerprint is based on the pattern type which can be precisely one of five classes, namely, whorl, right loop, left loop, arch, and tented arch. Since this classification is based not on exact minutiae points but instead on certain invariant characteristics of a fingerprint depending on the properties of the fingerprint as a whole, two such sets of characteristics of a partic-

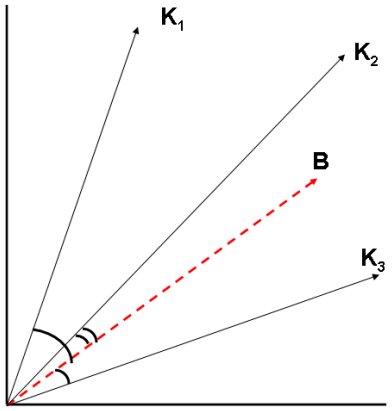


**Figure 4: Vector Representation**

ular fingerprint would be almost the same. We assign a weight to such invariant characteristics which is higher than the fine granular information provided by the fingerprint template, that typically has higher possibility to be erroneous. Multiple characteristics of different types would be required so as to uniquely generate a key.

Once the values corresponding to the elements of the characteristic vector of the input fingerprint are retrieved, we then use Salton's Vector Space Model [20] for retrieving all information required to evaluate the key. A valid pre-defined bio-key space and the input fingerprint are represented in a high-dimensional space where each dimension corresponds to a term in the characteristic vector of the biometric. Both the valid key space and the fingerprint input for key generation are vectors as illustrated in Figure 4.

If the key space is denoted as $\vec{K}$, then the $i^{th}$ vector in $\vec{K}$ is represented as $\vec{k_i} = \langle w_{i,1}, .., w_{i,m} \rangle$ where $w_{i,j}$ is the weight corresponding to the $j^{th}$ dimension, $1 \leq j \leq m$, of this vector. Similarly the biometric reading of the input biometric is denoted as $\vec{b} = \langle w_{b,1}, .., w_{b,m} \rangle$. The weights directly help in computing the similarity of any two vectors. The similarity is essentially the *cosine measure* of the angle between two such vectors. For two vectors $\vec{k}$ and $\vec{b}$ the cosine similarity is given by:

$$\cos\theta = \frac{\vec{k} \times \vec{b}}{|\vec{k}||\vec{b}|} = \frac{\sum_j w_{i,j} \times w_{b,j}}{\sqrt{\sum_j w_{i,j}^2}\sqrt{\sum_j w_{b,j}^2}}$$

Here $\vec{k} \times \vec{b}$ is the vector product of $\vec{k}$ and $\vec{b}$, calculated by multiplying corresponding weights together. The cosine measure also calculates the angle between the vectors in a high-dimensional virtual space. Different heuristics can be used to set the weight of each element in the vector [21]. The higher the weight the greater is the impact on the cosine. The main idea is to give more weight to those characteristics of the biometric which are more constant and unique for an individual. To weigh the unique characteristics, the term weighting used in [25], called inverse document frequency ($idf$ for brevity), can be used. Here the term rarity is a measure of its importance. It is calculated as $idf_j = log(B/bf_j)$. Here $B$ is the number of biometric samples in a collection and $bf$ us the frequency of the particular biometric feature. Thus $idf$ measures the importance of a feature based on that feature rarity. Often $idf$ is normalized to force values to fall in a particular range. Once the vectors $\vec{k_i}$ and $\vec{b}$ are generated, we are able to determine the closest match of $\vec{b}$ in the keyspace $\vec{K}$. The resulting $\vec{k_{match}}$ is then used as the biometric key for authentication.

**Protocol 1** generate-biometric-key

**Require:** Valid key space definition $\vec{K}$ in a vector space model. User $U$ with biometric $x$ provided to sensor $S$.
**Ensure:** $S$ is trusted with biometric data of $U$.
1: read_biometric $(x) \leftarrow t$ : $U$ sends its biometric data $x$ to $S$ and $t$ is the result of the feature extraction.
2: init_char_vector $(t) \leftarrow \vec{b^0}$ : This function outputs $\vec{b^0} = \langle w^0_{b,1}..w^0_{b,m} \rangle$ such that the weights for each dimension of the vector is set to 0.
3: weigh_char_vector $(\vec{b^0}) \leftarrow \vec{b}$ : This function outputs $\vec{b} = \langle w_{b,1}..w_{b,m} \rangle$ with appropriate weights for each dimension.
4: get_closest_key $(\vec{b}, \vec{K}) \leftarrow \vec{k_i}$ : This function outputs $\vec{k_i} = \langle w_{i,1}, .., w_{i,m} \rangle$ which is most similar to $\vec{b}$.
5: generate-crypto-key $(\vec{k_i}) \leftarrow m$ : The key vector $\vec{k_i}$ is transformed to an integer $m$ used as the cryptographic secret key.
6: **return** $m$

## 4.2 Using Biometric Keys for Zero Knowledge Proof

Once the bio-key is generated it can then be used in to perform a zero knowledge proof of knowledge (ZKPK) to authenticate the individual to the authenticating server. The ZKPK module illustrated in step 5 in Figure 3 is described in this section.

**Preliminary ZKP Concepts.** ZKP systems are interactive systems in which two parties, the prover and verifier, interact. The prover claims that a statement is true and the verifier wants to be convinced that this is true. At the end of the interaction, the verifier is either convinced that the statement is true, or alternatively, discovers that the statement is not true. ZKP's have extensively been used for identification purposes [2, 4, 6]. Using the ZKP for biometric data is not straightforward because the biometric template cannot be replicated exactly like the cryptographic keys. We therefore present a semantically secure ZKP based on the final key generated according to the approach described in Section 4.1. ZKP's are based on secrets that are hidden in tokens provided to the authenticating party at the time of enrollment. Such tokens are called *commitments*. Commitments are cryptographic tokens that enables the user being bound to a secret the possession of which can be verified at a later stage without revealing the secret itself.

We now present two relevant protocols: a protocol for generating the bio-key to use at authentication time and the commitment protocol.

**Protocol Description.** The main steps of the key generation protocol (see Protocol 1) can be summarized as follows. Note that the same key generation protocol is used both at the time of enrollment and authentication. Let $U$ be the individual to be authenticated. First the biometric template $x$ of $U$ is read by sensor $S$. Then (step 2) the client invokes function init_char_vector which extracts the desired features from the biometric template read. This is the initial characteristic vector which is weighed according to system defined heuristics in the function weigh_char_vector. At this point the input vector $\vec{b}$ is ready to be matched with the key space $\vec{K}$ to find the closest matching key vector $\vec{k_i}$. This key space is stored at the client device. This is executed by function get_closest_key which depends on the VSM technique. Once the key vector is identified, function generate-crypto-key reads this key vector to obtain the final secret used as a cryptographic key in the ZKP. This function may use an expansion function to sample the bio-key from a well spread or uniform distribution. The generated keys size is dictated by the group $Z_{2^{B+k}}$ where $k$ and $B$ are security parameters of the federation system. Additional secrets can also be incorporated in

**Protocol 2** ZKP with Biometric Commitments

**Require:** $U$, Registrar $Reg$ and Federations Verifiers $V$ agree on a group $\mathcal{G}$, a large integer $Func(k)$ and $2^B \rangle ord(\mathcal{G})$, $T$ is a public constant chosen arbitrarily large, $k$ and $B$ are security parameters.

**Ensure:** Private knowledge of $U$ is $m, r$.

{**Enrollment or Commitment Phase:**}

1: $Reg$ chooses $h \in \mathcal{G}$ which has a $Func(k)$ rough order, a random secret $s \in Z_{2^{B+k}}$. It sets $g \leftarrow h^s$ and sends the public key $K = (g; h)$ and proves that $g \in \langle h \rangle$.
2: $U$ chooses random $r \leftarrow Z_{2^{B+k}}$.
3: $U$ assigns $m :=$ generate-biometric-key .
4: $U$ sends its public commitment $C_K(m, r) := g^m * h^r$ to $Reg$ and stores $\delta$. ($\delta$ can optionally be stored at $Reg$)

{**Authentication or Proving Phase:**}

5: $U$ picks random $y \in [0..T * 2^k], s \in [0..T * 2^{B+2k}]$ at random and sends $d = g^y * h^s$ to $V$.
6: $V$ sends random challenge $e \in [0..Func(k)]$ to $U$.
7: $U$ assigns $m' :=$ generate-biometric-key $(x')$.
{$m'$ should be equal to $m$}
8: $U$ sends $u = y + em', v = s + er$ to $V$.
9: $V$: accepts if $g^u * h^v = d * c^e$
10: **return**

the last function if desired.

Functions in Protocol 1, collectively referred to as generate-crypto-key are in the Damgard Fugisaki Integer Commitment Scheme [8] as shown in Protocol 2. In the following we highlight the main steps which enable the use of biometric commitments. At the time of enrollment instead of sending the actual biometric $x$, to the authentication entity ($Reg$ in this case), $U$ at step 3 calculates the biometric commitment and generates the secret being committed as $m :=$generate-crypto-key($x$). In a typical integer commitment scheme $m$ would be the value of the sensitive data which is being committed. Consequently, at the time of verification $U$ will have to prove the knowledge of this $m$ and the random number $r$ which it generated at step 2 of Protocol 2. $U$ has to store $r$ as specified in a typical ZKP system, but $m$ should be generated at the time of authentication. More specifically, at step 7, $U$ can generate $m'$ using the same key generation function as enrollment. If the difference between $x$ and $x'$ is tolerable then committed secret $m$ will be equal to the retrieved secret $m'$. If $U$ has $m$ and $r$ it follows that the proof at step 9 will succeed. The ZKP can be efficiently computed as shown in [4]. The challenge response can be made non-interactive using fiat shamir heuristic to enhance the efficiency. As compared to other models where the bio-key could be based on symmetric encryption, would require the key to be known to the verifier, which we prevent in our trust model. Thus we have shown how biometric data can be successfully used in a ZKPK of authentication.

# 5. MULTI-FACTOR AUTHENTICATION

Several biometric mechanisms have been recently considered and proposed, such as DNA sequencing, retina scans and fingerprints. However, most commercially viable physical biometric set in the foreseeable future are fingerprints [22]. A multi-factor authentication becomes essential if fingerprint is the biometric chosen, because today's cheap fingerprint scanners are not reliable enough to be used alone. The fingerprint biometric may not be more secure than PINs. The final readings of fingerprints retrieved from a sensor are typically static values and their false acceptance rates imply, e.g., 1/100,000 security (i.e., perhaps 17 bits) [17]. Therefore fingerprint may be more affective as one of the factors in a multi-factor authentication. In Section 4 we showed how we can preserve the privacy of a biometric used for authentication. In this section we show how we can achieve privacy preserving multi-factor authentication providing relatively stronger authentication as compared to systems that are solely dependent on either *what you know* identifiers or *who you are* identifiers. This is because identifiers can be stolen or shared, and there is a threat to copy biometrics too. However, if a collection of these different types of identifiers are challenged in an ad-hoc, unpredictable manner, then such threats are mitigated sufficiently.

## 5.1 Two-Factor Authentication using ZKP

In Protocol 2 there are two specific secrets which have to be known to the user, namely $m$ and $r$[2]. $m$ corresponds to the number generated by Protocol 1 of the fingerprint of the prover. Thus, $m$ covers the *who you are* criterion of the authentication. The second secret $r$ is the random number generated by the prover when the initial enrollment is done. This is depicted in step 2 of Protocol 2. Without the knowledge of $r$ the proofs $u$ and $v$ cannot generated at step 8. $r$ is chosen such that its bit length is at least twice that of the order of $h$ and the final commitment is statistically close to uniform distribution in $\langle h \rangle$, for any value of $m$. This is essential for the hiding property of the ZKP. Hence, we conclude that the random $r$ serves as a second factor essential for a successful ZKPK used for authentication. $r$ thus corresponds to the *what you have* aspect of authentication. Note that even if one of the secrets, either $m$ or $r$ but not both, is compromised, the adversary cannot generate the correct proofs. We therefore assert that Protocol 2 provides a secure two-factor authentication.

## 5.2 Multi-Factor Authentication using Federation Registrar

Additional privacy preserving multi-factor authentication is possible by leveraging the registrar of the identity management system mentioned in Section 3. Typical enrollment of attributes omits step 3 of Protocol 2. In this manner an *identity record* corresponding to the proofs of identity is created. An example of such identity record is shown in Figure 2. Here, additional information regarding each committed identifier is also recorded which provides additional security properties as illustrated in [2]. Depending on the policy of the authenticator, multiple commitments of the identity record can be used for authentication. We omit the details of such proofs as they have been provided in our previous work [2].

Referring to Example 1, Alice can enroll with her identity provider *CityBank*. Now at the time when using non-financial services from $TaxAuthr$, with the help of Protocols 1 and 2, proof of the fingerprint itself provides the required two-factor authentication. To satisfy the authentication requirements at the time of financial transactions with $TaxAuthr$, Alice refers to the identity record stored at her local registrar (see Figure 2). She combines the proof of knowledge of the identifiers with tags CCN, SSN and Fingerprint and uses the same proof of knowledge as in Protocol 2 for each of the identifiers. Efficient combinations of such proofs have been explored in [4].

# 6. ANALYSIS

We now analyze the security, privacy and architectural aspects of the proposed authentication protocols. In particular we assess the level of protection provided by our mechanism against identity theft in the presence of malicious parties.

[2]All symbols and variables used in this section correspond to the ones presented in Protocol 2

## 6.1 Security Analysis

Before presenting the security properties of our system, we illustrate the key assumptions which our solution builds on.

ASSUMPTION 1. *The characteristic vector used in Protocol 1 is sound.*

The VSM technique adopted for the key generation in Protocol 1 requires that correct and sufficient features are recorded by the function init_char_vector. The output vector is then weighed by the function weigh_char_vector resulting in the final characteristic vector $\vec{b}$. This is however a realistic assumption as shown by the examples of possible characteristics provided in Section 4.1.

ASSUMPTION 2. *The characteristic vector used in Protocol 1 is sufficiently expressive to uniquely identify an individual with a high probability.*

This assumption relates to the expressiveness of the characteristic vector so that it can capture the uniqueness of a given biometric. We require that this is done with a sufficiently high probability, although we show in the security analysis that in the presence of collision Protocol 1 is still resistant to collision.

ASSUMPTION 3. *The initial enrollment of the biometric is secure.*

This assumption is especially true considering the current day biometric enrollment where the individual is required to come in person and the enrollment is performed in a controlled environment by the designated authorities. If the enrollment is done in an insecure fashion, it would lead to serious repercussions, especially in systems where the enrolled biometric is the only factor checked during authentication.

Based on the above assumption the following security properties hold.

THEOREM 1. **Soundness***: Let U be an individual, and B be the biometric associated with it. If U has enrolled using Protocol 2, then it can execute the authentication phase successfully.*

PROOF. At the time of enrollment $U$ generates a bio-key $m$ and also chooses a random $r$. $r$ is the only value stored with $U$. Then at authentication time, because of Assumption 1, the bio-key $m' = m$ can be regenerated. As evident from step 8 of Protocol 2 the final proof can be constructed correctly based on the retrieved values $m$ and $r$. Therefore Protocol 2 is sound. □

We now show an interesting result on the unforgeability of our protocols. In our context this notion is different to the conventional notion of key unforgeability, as for us unforgeability refers to the incapacity of any attacker to forge the protocol to be authenticated with someone else's biometric.

THEOREM 2. **Unforgeability** *(Two-factor): Let U be an individual with biometric B and associated random secret r, and A be an adversary with biometric B'. If $r$ is not known to A, then Protocol 2 satisfies unforgeability.*

PROOF. Let the keys generated at step 3 of Protocol 2 be $m_U$ for $U$ and $m_A$ for $A$. Under Assumption 2, with a high probability $m_U$ is not equal to $m_A$. If the enrollment has been executed correctly, then $m_U$ is the biometric enrolled. Therefore, the proof generated would be incorrect and step 9 of Protocol 2 will return false.

If however, $m_A$ happens to be equal to $m_U$ to complete the proof in step 8 $A$ would have to guess the random value $r$. Since this random value is chosen from $Z_{2^{B+k}}$ it is infeasible for an adversary to guess it. This condition holds true provided the secrecy condition on $r$ (as given in the theorem). Thus the two-factor authentication is unforgeable and the thesis holds. □

THEOREM 3. **Unforgeability** *(Multi-factor): Let U be an individual with biometric B included with other identifiers in an Identity Record IdR, and A be an adversary. Let a subset $I$ of identifier commitments in IdR including the commitment of B be used for authentication. If at least one of the secrets associated with $I$ is not known by A, then A cannot execute the authentication phase of Protocol 2 successfully.*

PROOF. Let $I$ be set defined as $\{i_1, .., i_n\}$ where $i_k, 1 \leq k \leq n$, is the biometric commitment. Note each identifier is also associated with random secrets $\{r_1, .., r_n\}$ generated at the time of the enrollment. Multi-factor authentication in essence executes Protocol 2 separately. If anyone of the $2n$ identifiers in $I$ and random secrets together is not known to the adversary, then the proof of at least one of the ZKPK will fail. This would result in failing the authentication process. Note that the length and the content of $I$ is chosen in an ad-hoc fashion at the time of authentication, thus making the challenge fresh and not pre-determined. This further makes the probability of forging minimal. □

THEOREM 4. **Revocation and Re-enrollment of the revoked biometric***: Let U be a user with biometric B enrolled in a federated system such that the bio-key generated with B is $m$. The authentication system supports revocation of the enrolled biometric and also re-enrollment of the same biometric as a different biometric commitment.*

PROOF. Revocation of $U$'s biometric commitment can be done trivially by including such commitment in a revocation list which is checked before authentication. Moreover, the IdR associated with $U$ can be updated to ensure that the status of this biometric commitment is updated. For further details corresponding to revocation of identifiers in IdR please see [2]. Once a commitment is revoked, the two secrets $m$ and $r$ corresponding to this commitment cannot be used together. Therefore for a legitimate re-enrollment of a biometric B, a new pair $m'$ and $r'$ has to be used such that either $m' \neq m$ or $r' \neq r$, or both. Since the random $r'$ will be chosen randomly, the re-enrollment of B with the same bio-key is possible. However, if required a new $m'$ can be generated in the generate-crypto-key function in Protocol 1, if such a procedure incorporates an additional random input at the time of generation of the bio-key. Therefore revocation and re-enrollment are possible in the authentication system using Protocols 1, 2. □

Note that the above discussion also implies that a user can enroll multiple times with the same biometric identifier and these enrollments cannot be linked based on the enrolled values.

**Identity Theft Protection**. Biometric identifiers correspond to the physical characteristics of a person and are as such harder to steal as compared to other identifiers which are normally stored in external devices. Furthermore, the enrollment procedure for biometrics is typically very strong thus leading to higher assurance on the enrolled biometric commitment. In our system, biometric keys are generated on the fly and are not stored either at the client or the server. The biometric templates in fact represent intrinsic information about the user, therefore theft of the template leads to identity theft.

It would be important that the freshness and liveness of a biometric be ensured by the biometric scanner (step 1 of Figure 3). If, however, an adversary manages a duplicate latex fingerprint to be able to generate the bio-key, then the identity theft attempt would be prevented as the attacker would still need to have the random secret to generate the proofs as required by the authentication Protocol 2. Even if the client device and ZKPK modules are compromised so as to maliciously store previous legitimate proofs, still the adversary cannot execute Protocol 2 successfully. This is because of the nature of the ZKPK itself, that requires a fresh random challenge and to reconstruct the proof each time.[3] Furthermore, from Theorem 3 we get strong authentication, which is in fact the predominant solution to mitigate the threat of identity theft.

## 6.2 Privacy

The problem with biometric data is that it inherently may contain other information not required for the purposes of authentication. Our approach of key generation based on the biometric maintains the advantages of the biometric authentication and at the same time prevents any leakage of additional personal information. Thus the data collected at the time of authentication cannot be used for any other purpose other than the authentication decision itself.

The bio-key generated is also secured based on the ZKPK proofs. Thus the bio-key cannot be reverse engineered to guess the characteristic vector used to generate that key. This further preserves privacy of the biometric. Moreover, as illustrated in the previous section, the unforgeability and revocability also help in preserving the privacy and misuse.

## 6.3 Architectural Issues

In our scheme, the actual biometrics template is never stored anywhere, thus providing storage efficiency and preventing the need of databases storing biometric templates. Accordingly, the database threats with respect to external and internal attackers, and tampering with stored templates are prevented.

A federated identity management system has to handle heterogeneity of the various clients and the servers. Using the methodology presented, the clients can generate the keys using any proprietary biometric scanner or software, and the server can still authenticate based on the same ZKPK. In fact, the level of interoperability is even higher, in that adding biometric authentication does not require additional verifications at the server end. As such, the server can verify the proofs of a biometric just like other identifiers stored in the users IdR. This also helps addressing deployment and scalability concerns.

The computational overhead at the client is also minimal since the vector space methods are efficient. The ZKPK proofs can be efficiently implemented and aggregation techniques have been proposed to further compute the multi-factor proofs in a concise manner.

# 7. RELATED WORK

Several efforts have been undertaken to strengthen client side authentication. Previous approaches like [3, 12] build on Chaum and Pederson wallet-with-observer paradigm. Here a tamper-proof device available at the clients end is required where the comparison is made. In this model a smart card provided to the user acts as the wallet, and the wallet runs a local process which is called the observer. The wallet is assumed to be tamper-resistant and powerful enough to carry out relatively expensive cryptographic computations. Security of the wallet in possession of the user is the key. Indeed, if the client side device is compromised then replay attacks or fraudulent authentication results may be sent to the server. Furthermore this model is not scalable nor interoperable. It can potentially be used by a party other than the owner which is an undesired property for biometric authentication.

Several efforts in cryptography based on biometrics are fuzzy commitments and fuzzy vaults [15, 16] and fuzzy extractors [10] have been proposed. However, several of these schemes may be vulnerable to replay attacks, non-repudiation and the vulnerability of the resultant keys generated to cryptanalysis. Many of the schemes mentioned depend on the fuzzy commitment scheme which was developed by Juels and Wattenberg [15] who have proposed an improvement and generalization of the approach by Davida et. al. [9] where a synthesis of error correcting codes with cryptographic techniques was proposed.

Unlike traditional commitment schemes, the fuzzy commitment scheme was designed mainly to be resilient to the small corruptions of the committed value, which is also called the *witness*. This means that if prover $P$ originally committed value $x$, then $P$ can potentially open the original commitment or decommit successfully with a value $x'$ very close to $x$. One of the ways to express closeness of two values is Hamming distance. This cryptographic primitive of fuzzy commitments was proposed to be achieved with the help of error correcting codes (ECC). ECC enable transmission of a message *m* intact over a noisy communication channel. An approach based on ECC is not trivial to implement because of the complexity and intensive calculations. Therefore we investigate an approach based on recording specific characteristics of a biometric, instead of dealing with the biometric template as a simple bit pattern. We believe such a mechanism is highly practical. We not only use the bio-key as it is, but we also combine it with ZKPK to assure the privacy of the bio-key itself.

Key Generation based on biometric aggregation has been investigated in [7]. Here several invariant features of different types of biometric are used to derive a bio-key that is used to encrypt a plain text message with a header information. The decryption is based on a new generated bio-key which may be not exactly the same as the initial key. Different permutations of the newly computed bio-key are used to decrypt the header of the encrypted message after which the rest of the message is inferred. This approach was shown efficient and addressed the non-repudiation problems. However, to be robust this scheme needs several biometrics. We provide a more fine granular approach which can generate different keys for closely looking biometrics. This greatly depends on the characteristic vector as highlighted in Section 4.1. We further use information theoretic ZKPK that provide additional privacy and security properties as shown in the analysis in Section 6.

# 8. CONCLUSION AND FUTURE WORK

In this paper we have developed a privacy preserving biometric authentication methodology. Our approach has several security and privacy advantages for the authentication mechanism and the biometric itself. We provide a new application of vector-space model to generate efficiently cryptographic biometric keys. We preserve privacy and unconditional security of the biometric key by employing information theoretically secure ZKPK. Our notion of privacy relates to the amount of data disclosed and the controlled usage of it, for the specific authentication purposes it was designed for. Moreover, we show how the biometric authentication can be combined with other identifiers used in a federated IdM system for authentication, thus resulting in a multi-factor authentication.

We plan to extend this work in several directions especially with

[3]This is true under the assumption that the bio-key is not stored in the device.

respect to experimentation. First we plan to work on simulations and evaluations of fingerprints, and voice prints biometrics to develop examples of the characteristic vectors. We will investigate how this characteristic vector differs in the various vendor specific biometric scanners. The second direction is to use this characteristic vector to generate a key space in VSM which is sufficiently expressive to generate different keys from closely related biometrics. Finally, we would like to investigate combining various biometrics in the VSM model as investigated in [7]

## 9. ACKNOWLEDGEMENT

The work reported in this paper has been sponsored by NSF under the ITR Project 0428554 "The Design and Use of Digital Identities". We thank Prof. Samuel Wagstaff and Shimon Modi for their useful suggestions.

## 10. REFERENCES

[1] The battle against phishing: Dynamic security skins. In *SOUPS '05: Proceedings of the 2005 symposium on Usable privacy and security*, pages 77–88, New York, NY, USA, 2005. ACM Press.

[2] A. Bhargav-Spantzel, A. Squicciarini, and E. Bertino. Establishing and protecting digital identity in federation systems. *Journal of Computer Security*, 13(3):269–300, 2006.

[3] G. Bleumer. Biometric yet privacy protecting person authentication. *Lecture Notes in Computer Science*, 1525:99–110, 1998.

[4] J. Camenisch and A. Lysyanskaya. Efficient non-transferable anonymous multi-show credential system with optional anonymity revocation. In B. Pfitzmann, editor, *Advances in Cryptology — EUROCRYPT 2001*, volume 2045, pages 93–118. Springer Verlag, 2001.

[5] D. Chaum and T. P. Pedersen. Wallet databases with observers. In *CRYPTO '92: Proceedings of the 12th Annual International Cryptology Conference on Advances in Cryptology*, pages 89–105, London, UK, 1993. Springer-Verlag.

[6] D. Chaum and H. van Antwerpen. Undeniable signatures. In *Advances in Cryptology — CRYPTO '89*, volume 435, pages 212–216. Springer-Verlag, 1990.

[7] C. R. Costanzo. Biometric cryptography: Key generation using feature and parametric aggregation. Online Technical Report, 2004.

[8] I. Damgård and E. Fujisaki. An integer commitment scheme based on groups with hidden order. In *Advances in Cryptology — ASIACRYPT 2002*, volume 2501. Springer, 2002.

[9] G. Davida, Y. Frankel, and B. Matt. The relation of error correction and cryptography to an offine biometric based identication scheme, 1999.

[10] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *Eurocrypt 2004*, 2006.

[11] Identity-Management. Liberty alliance project. `http://www.projectliberty.org`.

[12] R. Impagliazzo and S. M. More. Anonymous credentials with biometrically-enforced non-transferability. In *WPES '03: Proceedings of the 2003 ACM workshop on Privacy in the electronic society*, pages 60–71, New York, NY, USA, 2003. ACM Press.

[13] Internet2. Shibboleth. `http://shibboleth.internet2.edu`.

[14] A. Jain, S. Prabhakar, L. Hong, and S. Pankanti. Filterbank-based fingerprint matching, 2000.

[15] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *ACM Conference on Computer and Communications Security*, pages 28–36, 1999.

[16] A. Juels and M. Wattenberg. A fuzzy vault scheme. In *Proceedings of IEEE International Symposium on Information Theory, 2002.*, 2002.

[17] C. Mills. Biometrics: Back to security basics, rsa security, 2002.

[18] F. Monrose, M. Reiter, Q. Li, and S. Wetzel. Using voice to generate cryptographic keys. 2001.

[19] T. P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In J. Feigenbaum, editor, *Advances in Cryptology – CRYPTO '91*, volume 576, pages 129–140. Springer Verlag, 1992.

[20] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, 1975.

[21] H. soo Kim, I. Choi, and M. Kim. Refining term weights of documents using term dependencies. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 552–553, New York, NY, USA, 2004. ACM Press.

[22] U. Uludag, S. Pankanti, S. Prabhakar, and A. Jain. Biometric cryptosystems: Issues and challenges, 2004.

[23] I. M. R. VeriSign. Web Services Federation Language (WS-Federation). version 1.0. July 8 2003. `http://www-128.ibm.com/developerworks/library/specification/ws-fed/`.

[24] Z. W. Wang, S. K. M. Wong, and Y. Y. Yao. An analysis of vector space models based on computational geometry. In *SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 152–160, New York, NY, USA, 1992. ACM Press.

[25] H. Wu and G. Salton. A comparison of search term weighting: term relevance vs. inverse document frequency. In *SIGIR '81: Proceedings of the 4th annual international ACM SIGIR conference on Information storage and retrieval*, pages 30–39, New York, NY, USA, 1981. ACM Press.

[26] W. Zhang, Y.-J. Chang, and T. Chen. Optimal thresholding for key generation based on biometrics. In *ICIP*, pages 3451–3454, 2004.