The 12th Australasian Conference on Information Security and Privacy (ACISP '07).
(2–4 july 2007, Townsville, Queensland, Australia)
J. Pieprzyk, H. Ghodosi and E. Dawson Ed. Springer-Verlag, LNCS 4586, pages 96–106.

An Application of the Goldwasser-Micali Cryptosystem to Biometric Authentication*

Julien Bringer[1], Hervé Chabanne[1], Malika Izabachène[2], David Pointcheval[2],
Qiang Tang[2] and Sébastien Zimmer[2]

[1] Sagem Défense Sécurité
[2] Departement d'Informatique, École Normale Supérieure
45 Rue d'Ulm, 75230 Paris Cedex 05, France

Abstract This work deals with the security challenges in authentication protocols employing volatile biometric features, where the authentication is indeed a comparison between a fresh biometric template and that enrolled during the enrollment phase. We propose a security model for biometric-based authentication protocols by assuming that the biometric features to be public. Extra attention is paid to the privacy issues related to the sensitive relationship between a biometric feature and the relevant identity. Relying on the Goldwasser-Micali encryption scheme, we introduce a protocol for biometric-based authentication and prove its security in our security model.

Keywords. Authentication, biometrics, privacy.

1 Introduction

Security protocols generally rely on exact knowledge of some data, such as a cryptographic key, however there are particular applications where environment and human participation generate variability. In biometric-based cryptosystems, when a user identifies or authenticates himself using his biometrics, the biometric feature, which is captured by a sensor (e.g. a camera for iris biometrics), will rarely be the same twice. Thus, traditional cryptographic handling such as a hash value is not suitable in this case, since it is not error tolerant. As a result, the identification or authentication must be done in a special way, and moreover precaution is required to protect the sensitivity (or privacy) of biometrics.

We here consider a practical environment where a human user wants to authenticate himself to a database using his biometrics. A typical scenario is that some reference biometric data is stored inside a database, through which the server authenticates the user by checking whether or not a "fresh" biometric template sent by the sensor matches with the reference one. Our main focus is about biometrics such as iris [4], which can be extracted into binary strings. Therefore, an authentication leads to a comparison between two binary vectors. If the Hamming distance is adopted, then a comparison consists of computing the Hamming distance between the reference data and the fresh template and comparing this to a threshold.

To enforce privacy, we wish biometric data after their capture to be hidden in some way so that an adversary is unable to find out who is the real person that is trying to authenticate himself. Note that a live person is uniquely identified by

* Work partially supported by french ANR RNRT project BACH.

his biometrics and we want to hide the relationship between biometrics and the identity (used in an application). To achieve this goal, an application dependent identity is used and biometric matching is made over encrypted data. Moreover, to retrieve data to be compared with from the database, we introduce a new protocol to hide the index of record from the database.

1.1 Related Works

In [8] Juels and Wattenberg start the pioneering work by combining error correction codes with biometrics to construct fuzzy commitment schemes. Later on two important concepts about, i.e., secure sketch and fuzzy extractor, are widely studied. In [9], a number of secure sketch schemes have been proposed. In [6], Dodis *et al.* formalize the concept of fuzzy extractor, and propose to use for symmetric key generation from biometric features. In [2], Boyen *et al.* propose applications to remote biometric authentication using biometric information. Moreover, the work of Linnartz and Tuyls [10] investigates key extraction generated from continuous sources. In these schemes, biometric features are treated to be secret and used to derive general symmetric keys for traditional cryptographic systems.

There are a number of papers which deal with the secure comparison of two binary strings without using error correcting codes. In the protocol proposed by Atallah *et al.* [1], biometric features are measured as bit strings and subsequently masked and permuted during the authentication process. The comparison of two binary vectors modified following the same random transformation leads then to the knowledge of the Hamming distance. The main drawback of their protocol is that the client needs to store a number of secret values and update them during every authentication process, as the security relies mainly on these transformations.

Cryptographic protocols using homomorphic encryption may also allow us to compare directly encrypted data. For instance, Schoenmakers and Tuyls improve Paillier's public encryption protocol and propose to use it for biometric authentication protocols by employing multi-party computation techniques [12].

In summary, most of these protocols, except the work of [11] which uses biometry for Identity-Based Encryption, rely on the assumption that biometric features belonging to live users are private information. However, this assumption is not true in practice. As a user's biometric information, such as fingerprint, may be easily captured in daily life. In this paper, we assume that the biometric information is public, but the relationship between a user's identity and its biometric information is private.

1.2 Our contributions

In this paper we propose a general security model for biometric-based authentication. The model possesses a number of advantages over the existing ones: The first is that we lower the level of trust on the involved individual principals. The second is that extra attention has been paid to the privacy issues related to the sensitive relationship between a biometric feature and the relevant identities. Specifically, this relationship is unknown to the database and the matcher.

We propose a new biometric authentication protocol which is proved secure in our security model. Our protocol follows a special procedure to query the database, which, as in the case of Private Information Retrieval (PIR) protocol [3], allows to retrieve an item without revealing which item is retrieved. The protocol heavily exploits the homomorphic property of Goldwasser-Micali public-key encryption scheme [7], its ability to treat plaintext bit after bit, and the security is based on its semantic security, namely the quadratic residuosity assumption.

1.3 Organization of this Work

The rest of the paper is organized as follows. In Section 2, we describe our security model for (remote) biometric-based authentication. In Section 3, we describe a new protocol for biometric authentication. In Section 4, we give the security analysis of the new protocol in the new security model. In Section 5, we conclude the paper.

2 A New Security Model

For a biometric-based remote authentication system, we assume the system mainly consists of two parts: the client part and the server part. At the client side, we distinguish the following two types of entities:

- A human being U_i, for any $i \geq 1$, who registers his reference biometric template b_i at the server side, and provides fresh biometric information in order to obtain any service from the authentication server.
- A sensor $\mathcal{S}$ which is capable of capturing the user's biometric and extracting it into a binary string, namely a fresh template.

In practice, the template extraction process may involve a number of components, nonetheless, here we assume that the sensor implements all these functionalities. Implicitly, we assume that the sensor can communicate with the server.

At the server side, we distinguish the following three types of entities:

- An authentication server, denoted $\mathcal{AS}$, which deals with the user's service requests and provides the requested service.
- A database $\mathcal{DB}$, which stores users' biometric templates.
- A matcher $\mathcal{M}$, which helps the server to make a decision related to a user's request of authentication.

Fig. 1 below illustrates this model.

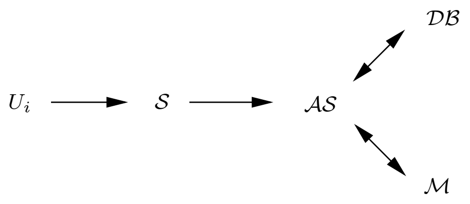

Figure1. Our model

Like most existing biometric-based systems (and many traditional cryptosystems), in our security model, a biometric-based authentication protocol consists of two phases: an enrollment phase and a verification phase.

1. In the enrollment phase, U_i registers its biometric template b_i at the database $\mathcal{DB}$ and its identity information ID_i at the server $\mathcal{AS}$.
2. In the verification phase, U_i issues an authentication request to the server $\mathcal{AS}$ through the sensor $\mathcal{S}$. The server $\mathcal{AS}$ retrieves U_i's biometric information from the database $\mathcal{DB}$ and makes its decision with the help of $\mathcal{M}$.

We assume that a "liveness link" is always available between the sensor $\mathcal{S}$ and the authentication server $\mathcal{AS}$ to ensure $\mathcal{AS}$ that the biometric it receives is from a present living person. The possible methods to achieve this liveness link are beyond the scope of this paper, but one can think about organizational measures or technical anti-spoofing countermeasures as those described in [5]. In addition, classical cryptographic challenge / response may also be used. This liveness link ensures that the server do not receive fake or replayed data. Since the sensor $\mathcal{S}$ is responsible for processing the biometric features, hence, it should be fully trusted and extensively protected in practice. Implicitly, the communications at the server side are also properly protected in the sense of authenticity. We further assume that all principals in the system will not collude and be honest-but-curious, which means they will not deviate from the protocol specification. In practice, certain management measures may be used to guarantee this assumption.

Let $\mathcal{H}$ be the distance function in the underlying metric space, for instance the Hamming space in our case. We regard soundness as a pre-requisite of any useful protocol. Formally, we have the following requirement.

Requirement 1 *The matcher $\mathcal{M}$ can faithfully compute the distance $\mathcal{H}(b_i, b'_i)$, where b_i is the reference biometric template and b'_i is the fresh biometric template sent in the authentication request. Therefore, $\mathcal{M}$ can compare the distance to a given threshold value d and the server $\mathcal{AS}$ can make the right decision.*

Our main concern is the sensitive relationship between U_i's identity and its biometrics. We want to guarantee that any principal except for the sensor $\mathcal{S}$ cannot find any information about the relationship. Formally, we have the following requirement.

Requirement 2 *For any identity ID_{i_0}, two biometric templates b'_{i_0}, b'_{i_1}, where $i_0, i_1 \geq 1$ and b'_{i_0} is the biometric template related to ID_{i_0}, it is infeasible for any of $\mathcal{M}$, $\mathcal{DB}$, and $\mathcal{AS}$ to distinguish between (ID_{i_0}, b'_{i_0}) and (ID_{i_0}, b'_{i_1}).*

We further want to guarantee that the database $\mathcal{DB}$ gets no information about which user is authenticating himself to the server. Formally, we have the following requirement.

Requirement 3 *For any two users U_{i_0} and U_{i_1}, where $i_0, i_1 \geq 1$, if U_{i_β} where $\beta \in \{0,1\}$ makes an authentication attempt, then the database $\mathcal{DB}$ can only guess β with a negligible advantage. Suppose the database $\mathcal{DB}$ makes a guess β', the advantage is $|\Pr[\beta = \beta'] - \frac{1}{2}|$.*

3 A New Biometric-based Authentication Protocol

3.1 Review of the Goldwasser-Micali Scheme

The algorithms $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ of Goldwasser-Micali scheme [7] are defined as follows:

1. The key generation algorithm $\mathcal{K}$ takes a security parameter 1^ℓ as input, and generates two large prime numbers p and q, $n = pq$ and a non-residue x for which the Jacobi symbol is 1. The public key pk is (x, n), and the secret key sk is (p, q).
2. The encryption algorithm $\mathcal{E}$ takes a message $m \in \{0,1\}$ and the public key (x, n) as input, and outputs the ciphertext c, where $c = y^2 x^m \mod n$ and y is randomly chosen from $\mathbb{Z}_n^*$.
3. The decryption algorithm $\mathcal{D}$ takes a ciphertext c and the private key (p, q) as input, and outputs the message m, where $m = 0$ if c is a quadratic residue, $m = 1$ otherwise.

It is well-known (cf. [7]) that, if the quadratic residuosity problem is intractable, then the Goldwasser-Micali scheme is semantically secure. In other words an adversary $\mathcal{A}$ has only a negligible advantage in the following game.

```
$\mathbf{Exp}_{\mathcal{E},\mathcal{A}}^{\text{IND-CPA}}$
    $(sk, pk) \leftarrow \mathcal{K}(1^\ell)$
    $(m_0, m_1) \leftarrow \mathcal{A}(pk)$
    $c \leftarrow \mathcal{E}(m_\beta, pk), \ \beta \leftarrow \{0,1\}$
    $\beta' \leftarrow \mathcal{A}(m_0, m_1, c, pk)$
    return $\beta'$
```

At the end of this game, the attacker's advantage $\mathbf{Adv}_{\mathcal{E},\mathcal{A}}^{\text{IND-CPA}}$ is defined to be

$$\mathbf{Adv}_{\mathcal{E},\mathcal{A}}^{\text{IND-CPA}} = \left|Pr[\mathbf{Exp}_{\mathcal{E},\mathcal{A}}^{\text{IND-CPA}} = 1|\beta = 1] - Pr[\mathbf{Exp}_{\mathcal{E},\mathcal{A}}^{\text{IND-CPA}} = 1|\beta = 0]\right|.$$

Moreover the encryption protocol possesses a nice homomorphic property, for any $m, m' \in \{0, 1\}$ the following equation holds.

$$\mathcal{D}(\mathcal{E}(m, pk) \times \mathcal{E}(m', pk), sk) = m \oplus m'$$

Note that the encryption algorithm encrypts one bit at a time, hence, in order to encrypt a binary string we need to encrypt every bit individually. We thus have the following property.

Lemma 1 ([7]) *Given any $M \geq 1$, the attacker's advantage in the following game is negligible based on the quadratic residuosity assumption.*

$$\mathbf{Exp}_{\mathcal{E},\mathcal{A}'}^{\textit{P-IND-CPA}}$$
$$\left| \begin{array}{cl} (sk, pk) & \leftarrow \mathcal{K}(1^\ell) \\ ((m_{0,1}, \ldots, m_{0,M}), (m_{1,1}, \ldots, m_{1,M})) & \leftarrow \mathcal{A}'(pk) \\ c & \leftarrow (\mathcal{E}(m_{\beta,1}, pk), \ldots, \mathcal{E}(m_{\beta,M}, pk)),\ \beta \leftarrow \{0, 1\} \\ \beta' & \leftarrow \mathcal{A}'((m_{0,1}, \ldots, m_{0,M}), (m_{1,1}, \ldots, m_{1,M}), c, pk) \\ \textit{return } \beta' & \end{array} \right.$$

3.2 Enrollment Phase

In the protocol we treat U_i's biometric template b_i as a binary vector of the dimension M, i.e. $b_i = (b_{i,1}, b_{i,2}, \ldots, b_{i,M})$.

In the enrollment phase, U_i registers (b_i, i) at the database $\mathcal{DB}$, and (ID_i, i) at the authentication server $\mathcal{AS}$, where ID_i is U_i's pseudonym and i is the index of the record b_i in $\mathcal{DB}$. Let N denotes the total number of records in $\mathcal{DB}$.

The matcher $\mathcal{M}$ possesses a key pair (pk, sk) for the Goldwasser-Micali scheme $(\mathcal{K}, \mathcal{E}, \mathcal{D})$, where $pk = (x, n)$ and $sk = (p, q)$.

3.3 Verification Phase

If the user U_i wants to authenticate himself to the authentication server $\mathcal{AS}$, the procedure below is followed:

1. The sensor $\mathcal{S}$ captures the user's biometric data b'_i, and sends $\mathcal{E}(b'_i, pk)$ together with the user's identity ID_i to the authentication server $\mathcal{AS}$, where
$$\mathcal{E}(b'_i, pk) = (\mathcal{E}(b'_{i,1}, pk), \mathcal{E}(b'_{i,2}, pk), \ldots, \mathcal{E}(b'_{i,M}, pk)).$$
Note that a "liveness link" is available between $\mathcal{S}$ and $\mathcal{AS}$ to ensure that data coming from the sensor are indeed fresh and not artificial.
2. The server $\mathcal{AS}$ retrieves the index i using ID_i, and then sends $\mathcal{E}(t_j, pk)$ $(1 \leq j \leq N)$ to the database, where $t_j = 1$ if $j = i$, $t_j = 0$ otherwise.
3. For every $1 \leq k \leq M$, the database $\mathcal{DB}$ computes $\mathcal{E}(b_{i,k}, pk)$, where
$$\mathcal{E}(b_{i,k}, pk) = \prod_{j=1}^{N} \mathcal{E}(t_j, pk)^{b_{j,k}} \mod n,$$
Then it sends these $\mathcal{E}(b_{i,k}, pk)$ $(1 \leq k \leq M)$ to the authentication server $\mathcal{AS}$.

4. The authentication server $\mathcal{AS}$ computes ν_k $(1 \le k \le M)$, where

$$\begin{aligned}\nu_k &= \mathcal{E}(b'_{i,k}, pk)\mathcal{E}(b_{i,k}, pk) \mod n \\ &= \mathcal{E}(b'_{i,k} \oplus b_{i,k}, pk)\end{aligned}$$

   It then makes a random permutation among ν_k $(1 \le k \le M)$ and sends the permuted vector λ_k $(1 \le k \le M)$ to the matcher $\mathcal{M}$.
5. The matcher $\mathcal{M}$ decrypts the λ_k $(1 \le k \le M)$ to check if the Hamming weight of the corresponding plaintext vector is equal to or less than d, and sends the result to $\mathcal{AS}$.
6. The authentication server $\mathcal{AS}$ accepts or rejects the authentication request accordingly.

To sum up, $\mathcal{S}$ stores the public key pk, $\mathcal{AS}$ stores the public key pk and a table of relations (ID_i, i) for $i \in \{1, \ldots, N\}$, $\mathcal{DB}$ contains the enrolled biometric data $b_1, \ldots, b_N$, and $\mathcal{M}$ possesses the secret key sk, then the protocol runs following Fig. 2.

$\mathcal{S}$ Capture b'_i from U_i	$\xrightarrow{\mathcal{E}(b'_i,pk),ID_i}$	$\mathcal{AS}$
$\mathcal{AS}$ Choose $t_j = \delta_{i,j}$	$\xrightarrow{(\mathcal{E}(t_j,pk))_{1\le j\le N}}$	$\mathcal{DB}$
$\mathcal{AS}$	$\xleftarrow{(\mathcal{E}(b_{i,k},pk))_{1\le k\le M}}$	For $1 \le k \le M$, compute $\prod_{j=1}^{N} \mathcal{E}(t_j, pk)^{b_{j,k}} \mod n$ $= \mathcal{E}(b_{i,k}, pk)$
Compute $\nu_k = \mathcal{E}(b'_{i,k}, pk)\mathcal{E}(b_{i,k}, pk) \mod n$ $= \mathcal{E}(b'_{i,k} \oplus b_{i,k}, pk)$ Take a random permutation σ, compute $\lambda_k = \nu_{\sigma(k)}$	$\xrightarrow{\lambda_1,\ldots,\lambda_M}$	$\mathcal{M}$ Check the weight of $(\mathcal{D}(\lambda_1, sk), \ldots, \mathcal{D}(\lambda_M, sk))$
$\mathcal{AS}$	$\xleftarrow{\text{OK / NOK}}$	$\mathcal{DB}$

Figure2. The Authentication protocol

It is easy to verify that the sensor $\mathcal{S}$ performs at most $2M$ modular multiplications, the server performs $2N$ modular multiplications in step 2 (which can be pre-computed) and M modular multiplications in step 4. The database needs to perform $\frac{MN}{2}$ modular multiplications in step 3, if we assume that 0 and 1 are equally distributed in the set $\{b_{j,k}\}_{1\le j\le N, 1\le k\le M}$. The matcher performs M modular exponentiations to check quadratic residuosity modulo p. And the overall communication complexity is linear on the number N of records in the database.

4 Security Analysis of the Protocol

The introduction of the matcher $\mathcal{M}$, which holds the decryption key, effectively limits the access to users' biometric information. The matcher $\mathcal{M}$ can only obtain the Hamming distance between two measurements of any user's biometrics, which actually can be thought of being public information. The server does not store any biometric information, hence, compromise of the server leaks no information to an outside attacker. Moreover, biometrics are almost always handled in an encrypted form.

Indeed the biometric templates are stored in plaintext in the database $\mathcal{DB}$, however, without any relevant identity information. In case that the database is compromised, no sensitive relationship information would be leaked, though we consider encrypting the biometric templates in the database is an interesting future research topic.

In the next section we show that the protocol satisfies the requirements described in Section 2.

4.1 Fulfillment of our Requirements

In step 4 of the protocol, we show that $\nu_k = \mathcal{E}(b'_{i,k} \oplus b_{i,k}, pk)$ for $1 \leq k \leq M$. Obviously, the Hamming distance between b_i and b'_i, $\mathcal{H}(b_i, b'_i)$, is equal to the Hamming weight of the plaintext vector corresponding to $(\nu_1, \ldots, \nu_M)$ and $(\lambda_1, \ldots, \lambda_M)$. Hence, it is straightforward to verify that **Requirement 1** is fulfilled.

We next show that the authentication protocol satisfies **Requirement 2** under the quadratic residuosity assumption.

Theorem 1 *For any identity ID_{i_0} and two biometric templates b'_{i_0}, b'_{i_1}, where $i_0, i_1 \geq 1$ and b'_{i_0} is the biometric template related to ID_{i_0}, any of $\mathcal{M}$, $\mathcal{DB}$, and $\mathcal{AS}$ can only distinguish between (ID_{i_0}, b'_{i_0}) and (ID_{i_0}, b'_{i_1}) with a negligible advantage.*

Proof. It is clear that the matcher $\mathcal{M}$ and the database $\mathcal{DB}$ have advantage 0 in distinguishing between (ID_{i_0}, b'_{i_0}) and (ID_{i_0}, b'_{i_1}), because they have no access to any information about users' identities.

As to the server $\mathcal{AS}$, the proof follows. From (ID_{i_0}, b'_{i_β}) with $\beta \in \{0, 1\}$, if the database $\mathcal{AS}$ can guess β with a non-negligible advantage δ, then we construct an attacker $\mathcal{A}$ for the Goldwasser-Micali scheme (as defined in Lemma 1) which has the advantage δ. The attacker simulates the protocol executions for the server $\mathcal{AS}$.

Suppose $\mathcal{A}$ receives pk from the challenger and gets a challenge $c_d = \mathcal{E}(m_{i_d}, pk)$ for $m_{i_0} \neq m_{i_1}$, where d is a random bit chosen by the challenger. $\mathcal{A}$ simulates the protocol executions by assuming that the matcher $\mathcal{M}$ and the database $\mathcal{DB}$ take pk as the public key. Then $\mathcal{A}$ registers m_{i_0} and m_{i_1} in the database. Note that it is straightforward to verify that the protocol execution for $\mathcal{AS}$ can be faithfully simulated by $\mathcal{A}$, and the knowledge of private key sk is not needed. If the server $\mathcal{AS}$ outputs a guess β', then $\mathcal{A}$ outputs the guess bit $d' = \beta'$ for d. As $\mathcal{A}$ wins if $\mathcal{AS}$ wins, the theorem now follows from Lemma 1. □

Now we prove that the authentication protocol also satisfies **Requirement 3** under the quadratic residuosity assumption.

Theorem 2 *For any two users U_{i_0} and U_{i_1}, where $i_0, i_1 \geq 1$, if U_{i_β} where $\beta \in \{0,1\}$ makes an authentication attempt, then the database $\mathcal{DB}$ can only guess β with a negligible advantage.*

Proof. If the database $\mathcal{DB}$ can guess β with a non-negligible advantage δ, then we construct an attacker $\mathcal{A}$ for the Goldwasser-Micali scheme which has the advantage δ.

Suppose $\mathcal{A}$ receives pk from the challenger and gets a challenge $c_d = \mathcal{E}(m_d, pk)$ for $m_0 = 0, m_1 = 1$, where d is a random bit chosen by the challenger. In addition, $\mathcal{DB}$ takes pk as the matcher's public key. For any $i_0, i_1 \geq 1$ and $i_0 \neq i_1$, $\mathcal{A}$ issues a query with $\mathcal{E}(t_j, pk)$ $(1 \leq j \leq N)$, where $\mathcal{E}(t_{i_1}, pk) = c_d$, $\mathcal{E}(t_{i_0}, pk) = y^2 x c_d$ where y is randomly chosen from $\mathbb{Z}_n^*$, and $t_j = 0$ for all $1 \leq j \leq N, j \neq i_0, j \neq i_1$. If the database $\mathcal{DB}$ outputs a guess β', then $\mathcal{A}$ outputs the guess bit $d' = \beta'$ for d. And it is straightforward to verify that $\mathcal{A}$ wins if $\mathcal{DB}$ wins. □

4.2 Advantages of the protocol

To emphasize the interest of our protocol, we further compare it with one recent protocol of Atallah *et al.* [1] which also allows the comparison between two binary biometric templates.

In the protocol of Atallah *et al.* [1] two entities are involved: a server which stores some information about the reference data b and a client (with a biometric sensor) which sends other information derived from the measured data b'. In the initialization phase, the client stores a random permutation Π_1 of $\{0,1\}^n$ and three random boolean vectors s_1, s_2, r_1. The client then sends $s_1 \oplus \Pi_1(b_1 \oplus r_1), \mathsf{H}(s_1), \mathsf{H}(s_1, \mathsf{H}(s_2))$ to the server for backup, where H is a hash function and b_1 is the user's biometric data. When measuring a new features vector b_2, the client sends $s_1, \Pi_1(b_2 \oplus r_1)$ to the server which could then verify the value of $\mathsf{H}(s_1)$ and compute the Hamming distance of b_1, b_2 to check if it is in an acceptable range. Thereafter, the remaining vectors are used to renew all the information stored at the client and the server sides for a future authentication.

The main drawback of this protocol is that the client needs to store secret values. Once these values are compromised, the attacker would be able to compute a user's biometric template easily by passively eavesdropping on the communication channel. It is also possible to show that an active attacker could impersonate the client to the server. Finally, it is also clear that the user's privacy is not ensured against the server. Therefore, it makes sense for us to explore new protocols that avoid these drawbacks.

Hence, the most important points that make our protocol more appropriate for biometrics authentication protocols are the following. Firstly, no secret information storage is required at the client side. Secondly, the protocol guarantees the privacy of the relationship between the user's identity and its biometric data, and the privacy of the user's biometric information.

5 Conclusion

In this paper, we considered a biometric authentication protocol where confidentiality is required for biometric data solely for privacy reasons. We captured these notions into a security model and introduced a protocol which is proved secure in this security model. It remains an interesting issue to improve its performance. For a better acceptability, we also want to look at an extension of this work where biometric data inside the database are also encrypted.

Acknowledgments

We would like to thank Michel Abdalla for the fruitful discussions.

References

1. M. J. Atallah, K. B. Frikken, M.l T. Goodrich, and R. Tamassia. Secure biometric authentication for weak computational devices. In A. S. Patrick and M. Yung, editors, *Financial Cryptography*, volume 3570 of *Lecture Notes in Computer Science*, pages 357–371. Springer, 2005.
2. X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith. Secure remote authentication using biometric data. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 147–163. Springer, 2005.
3. B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan. Private information retrieval. *J. ACM*, 45(6):965–981, 1998.
4. J. Daugman. How iris recognition works. In *ICIP (1)*, pages 33–36, 2002.
5. J. Daugman. Iris recognition and anti-spoofing countermeasures. In 7*-th International Biometrics Conference*, 2004.
6. Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 523–540. Springer, 2004.
7. S. Goldwasser and S. Micali. Probabilistic encryption and how to play mental poker keeping secret all partial information. In *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing, 5-7 May 1982, San Francisco, California, USA*, pages 365–377. ACM, 1982.
8. A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *ACM Conference on Computer and Communications Security*, pages 28–36, 1999.
9. Q. Li and E. Chang. Robust, short and sensitive authentication tags using secure sketch. In *MM&Sec '06: Proceeding of the 8th workshop on Multimedia and security*, pages 56–61. ACM Press, 2006.
10. Jean-Paul M. G. Linnartz and Pim Tuyls. New shielding functions to enhance privacy and prevent misuse of biometric templates. In Josef Kittler and Mark S. Nixon, editors, *AVBPA*, volume 2688 of *Lecture Notes in Computer Science*, pages 393–402. Springer, 2003.
11. A. Sahai and B. Waters. Fuzzy identity-based encryption. In R. Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*, pages 457–473. Springer, 2005.
12. B. Schoenmakers and P. Tuyls. Efficient binary conversion for Paillier encrypted values. In S. Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 522–537. Springer, 2006.