New Security Definitions for Biometric Authentication with Template Protection: Toward covering more threats against authentication systems

Toshiyuki Isshiki[1], Toshinori Araki[1], Kengo Mori[1], Satoshi Obana[2], Tetsushi Ohki[3], Shizuo Sakamoto[1]

[1] NEC Corporation [2] Hosei University
[3] National Institute of Advanced Industrial Science and Technology (AIST)
[1] {t-issiki@bx, t-araki@ek, ke-mori@bx, s-sakamoto@bu}.jp.nec.com
[2] obana@hosei.ac.jp [3] tetsushi.ohki@aist.go.jp

Abstract: Existing studies on the security of biometric authentication with template protection have considered the adversaries who obtain only protected templates. Since biometric authentication systems transmit data other than the protected templates, we need to consider how to secure biometric authentication systems against adversaries with those data. In this paper, we propose a classification of adversaries in biometric authentication with template protection into the following three types in accordance with their knowledge: (1) protected template data, (2) data transmitted during authentication, and (3) both types of data. We also propose a new security metric *unforgeability*, which provides authentication security against attacks by adversaries impersonating someone else on authentication systems even when they cannot obtain the biometric information of a claimant. We then give security definitions against each type of adversary we classified. We also propose a biometric authentication scheme with template protection that is irreversible against all types of adversaries.

1 Introduction

Biometric authentication provides advantages in terms of better usability in systems with person identification functions by freeing users from having to remember something or carry around a token. However, biometric authentication may have some vulnerabilities, which appear in various system elements including users, environmental conditions, operating conditions, biological data, and biometric equipment. Among these vulnerabilities, template leakage is the most critical. Leaked templates can be abused by adversaries for replay attacks. Even though replay attacks are prevented, biometric information can be forged by using the leaked templates [Ma03, Ca07]. Hence, the biometric templates must be protected for security.

Many techniques such as *cancelable biometrics* [NNJ10] and *biometric cryptosystems* [Tu05] have already been proposed to improve biometric template security. The performance of template protection techniques can be evaluated not only by recognition performance but also from security and privacy aspects. The latter metrics have not been established yet,

and many researchers are still seeking them. Previous works mainly focus on security and privacy against stored templates. Simoens et al. proposed and evaluated irreversibility and unlinkability of biometric cryptosystems [STP09]. Zhou also defined a systematic evaluation framework to assess irreversibility and unlinkability [Zh11]. Recently, Inuma et al. proposed alternative definitions of irreversibility and unlinkability [IO12]. Their work also focuses on security and privacy against stored auxiliary data. Wang et al. proposed revocability and reusability of biometric cryptosystems [Wa11], and Nagar et al. also proposed similar properties for cancelable biometrics and bio-hashing [NNJ10]. ISO/IEC 24745 [ISO11] defined reference architecture for template protection, and Simoens et al. [Si12] proposed criteria and several metrics that comply with ISO/IEC 24745.

The above metrics only consider the adversaries with stored data. However, since biometric authentication systems transmit data other than stored data, we need to consider how to secure the biometric authentication systems against adversaries who obtain those data. It is especially important to consider how to secure biometric authentication systems against adversaries who obtain query data, because query data transmitted during authentication are usually generated from biometric information of claimants. For this purpose, in this paper, we classify adversaries into the following three types in accordance with their knowledge. The first obtains protected templates. These adversaries have been considered in existing studies. The second obtains query data transmitted during authentication. The third obtains both protected templates and query data, whose importance will show in Section 4.1. As discussed above, no security properties against the second and third types of adversaries have been defined yet. We thus present the security definitions against each type of adversary. We also propose a new security metric for biometric authentication with template protection: *unforgeability*. Unforgeability provides authentication security against attacks by adversaries impersonating someone else on authentication systems even when they cannot obtain the biometric information of a claimant. We show that this property is not weaker than the irreversibility in [Si12]. Furthermore, we evaluate the security of the scheme in [Tu05] under our security definitions. We show that though the scheme [Tu05] is irreversible against the first and second types of adversaries respectively, the third type can break its irreversibility. Then, we propose a biometric authentication scheme with template protection that is irreversible against all types of adversaries. We cannot have provided a proof that the proposed scheme is unforgeable in this paper yet. Constructing a scheme possessing the unforgeability is a future work.

2 Biometric Authentication with Template Protection

This section introduces the model, components, and schemes of biometric authentication with template protection.

2.1 Model

In this paper, we deal with the model in which the biometric references and the identity references are stored on a server. In the target model, the query data extracted from the biometric information of a claimant are transferred to the server. The model can be applied to verification (i.e., one-to-one matching) and to identification (one-to-many matching). The merit of such model is that a claimant requires only his/her biometric characteristic for verification/identification and does not need to memorize any secret data nor bring any physical devices such as smart cards. Figure 1 shows our model, which is slightly modified from Model A [ISO11]. More specifically, we introduce session specific data (SSD for short). The motivation of introducing such data is to secure the system against replay attacks. Details are described in Section 3.

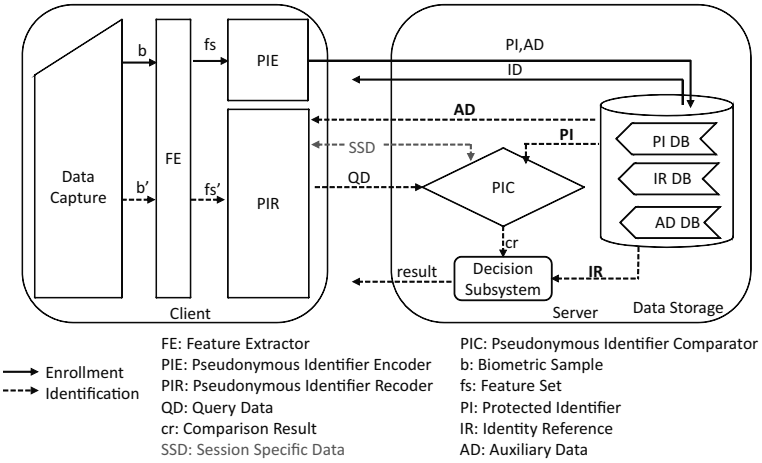

Figure 1: The model referenced in this paper.

2.2 Components

The biometric authentication schemes consist of the following five components.

• **Feature Extraction**: given a biometric sample b as input, extracts a feature set fs of the sample b. We denote $fs \leftarrow \mathsf{FE}(b)$.

• **Pseudonymous Identifier Encoder**: takes a feature set fs as input and generates a *protected template* PT. We denote $PT = (PI, AD) \leftarrow \mathsf{PIE}(fs)$, where PI is a pseudonymous identifier and AD is auxiliary data.

• **Pseudonymous Identifier Recoder**: takes a feature set fs', $\boldsymbol{AD}$, and session specific data SSD as input and generates query data QD, where $\boldsymbol{AD}$ is a set of ADs. We denote $QD \leftarrow \mathsf{PIR}(fs', \boldsymbol{AD}, SSD)$.

• **Pseudonymous Identifier Comparator**: takes query data QD, a set $\boldsymbol{PI}$ of PIs, and session specific data SSD as input and outputs a comparison result cr, where $cr \subset \boldsymbol{PI}$ or $cr = \perp$. We denote $cr \leftarrow \mathsf{PIC}(QD, \boldsymbol{PI}, SSD)$.

• **Decision Subsystem**: takes a comparison result cr and an identity reference set $\boldsymbol{IR}$ as input, and outputs a result $result$, which is a set of accepted IDs or is $reject$. We denote $result \leftarrow \mathsf{DS}(cr, \boldsymbol{IR})$.

2.3 Biometric Authentication Schemes with Template Protection

We use the notation $(y_c, y_s) \leftarrow P[C(x_c), S(x_s)](x)$ to denote that an interactive protocol P between *client* C with private input x_c and *server* S with private input x_s is to run with common input x. At the end of the protocol P, C's output is y_c and S's output is y_s. If a party has no input or output, we use the placeholder "-".

One of the biometric authentication schemes $\langle \mathrm{ENROLL}, \mathrm{IDENTIFY} \rangle$ is for identification with template protection that consists of two protocols: ENROLL and IDENTIFY. We assume that all entities know the system parameters $param$.

The enrollment protocol ENROLL is an interactive protocol between C and S. C which privately takes a biometric sample b_c as input and S which maintains an enrollment storage, collaboratively execute the enrollment protocol. In the enrollment protocol, C extracts the feature set *fs* by the feature extraction component FE with input b_c. Then, C generates the protected template PT by the pseudonymous identifier encoder component PIE with input *fs*. The PT consists of a pseudonymous identifier (PI) and auxiliary data (AD). The PI represents the individual and is used as reference for identification. The AD helps to generate query data QD in the identification phase. C sends $PT = (PI, AD)$ to S and S stores them with original user identity reference ID in the enrollment storage. We denote this protocol by $(\text{-}, ID) \leftarrow \mathrm{ENROLL}[C(b_c), S(\text{-})](param)$.

The identification protocol IDENTIFY is an interactive protocol for identifying an individual from the enrollment storage after sharing the session specific data SSD between C and S if needed. C with a biometric sample b_c' and S with a set $\boldsymbol{PI}$ of PIs and a set $\boldsymbol{IR}$ of IDs as private inputs collaboratively execute an identification protocol with common input $\boldsymbol{AD}$ and SSD. $\boldsymbol{AD}$ is a set of auxiliary data AD generated in ENROLL. In the identification protocol, C extracts his/her feature set fs' by the feature extraction component FE with a biometric sample b_c' as private input. C then generates query data QD by the pseudonymous identifier recoder component PIR with input fs', $\boldsymbol{AD}$, and the session specific data SSD and sends the query data QD to S. S executes the pseudonymous identifier comparator component PIC with input $\boldsymbol{PI}$, SSD, and QD, and obtains a comparison result cr. cr is a subset of $\mathbf{PI}$ or is $\perp$. S executes the decision subsystem DS with input cr and $\boldsymbol{IR}$ and obtains a result $result$. $result$ is a subset of $\boldsymbol{IR}$ or is $reject$. We denote this protocol by $(\text{-}, result) \leftarrow \mathrm{IDENTIFY}[C(b_c'), S(\boldsymbol{PI}, \boldsymbol{IR})](param, \boldsymbol{AD}, SSD)$.

Note that we can introduce a one-to-one verification scheme by adding ID to the common input in the identification protocol and replacing $\boldsymbol{AD}$ with AD corresponding to ID. S executes PIC with input PI_{ID}, QD, and SSD, where PI_{ID} is the protected template of ID.

We denote the verification protocol by (-, $result$) $\leftarrow$ VERIFY$[C(b'_c), S(\boldsymbol{PI}, \boldsymbol{IR})](param, AD, ID, SSD)$.

3 New Security Definitions

3.1 Proposed Classification of Adversaries

Existing studies on the security of biometric authentication with template protection have considered only adversaries who obtain PIs. However, since biometric authentication systems transmit data other than stored data, we need to consider how to secure biometric authentication systems against adversaries who obtain those data. Thus, we classify adversaries into the following three types in accordance with their knowledge, based on PIs as well as QDs.

- **The adversary with PIs** obtain leaked protected templates.
- **The adversary with QDs** obtain QDs transmitted in authentication.
- **The adversary with PIs and QDs** obtain leaked protected templates and transmitted QDs.

The first type of adversary is captured in the metrics by Simoens et al. [Si12]. However no properties have been defined against the second and the third. The second covers the replay attack in which the adversary who observed the QD sent by a genuine user tries to impersonate him/her by re-transmitting the QD as is. To prevent such attacks, QD should not be independent of session (i.e., QD accepted in a session should not be accepted in the other sessions). To make QD session dependent, we introduce SSD into our model, where (possibly random) SSD can be used to compute QD. If we employ a secure channel, we may obtain a scheme secure against the adversaries with QDs. However, the secure channel technique is not sufficient against the adversaries with compromised servers, classified into the third.

3.2 Proposed Security Definitions

Irreversibility [Si12] focuses on the adversaries who want to obtain biometric information of a claimant. However, from the view-point of authentication systems, they may be considered to be broken even if the adversaries successfully impersonate someone else. Therefore, we define a new metric, *unforgeability*, which captures such adversaries, in the same manner as the security of digital signatures.

In the following, we propose three types of unforgeability in accordance with Section 3.1. Note that we only discuss the one-to-many identification scheme, but it is easy to expand our discussion to the one-to-one verification scheme in a similar way.

3.2.1 Unforgeability against Attacks with PIs

Unforgeability against attacks with PIs is defined via the following game involving an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$. Let n be the number of enrolled claimants.

Setup. $\mathcal{C}$ generates system parameters $param$ and randomly n samples $\{b_1, b_2, \ldots, b_n\}$. Then, for $i = 1, 2, \ldots, n$, $\mathcal{C}$ executes $fs_i \leftarrow \mathsf{FE}(b_i)$ and $(PI_i, AD_i) \leftarrow \mathsf{PIE}(fs_i)$. We assume that for $i = 1, 2, \ldots n$, both PI_i and AD_i are indexed as ID_i and denote the set of ID_i by $\boldsymbol{IR}$. $\mathcal{C}$ sends $param$ and $\boldsymbol{AD}$ to $\mathcal{A}$.

Phase 1. $\mathcal{A}$ is permitted to make queries to $\mathcal{O}_{\mathrm{Corrupt}}$, $\mathcal{O}_{\mathrm{Enroll}}$, $\mathcal{O}_{\mathrm{PI}}$, and $\mathcal{O}_{\mathrm{PIC}}$:

- $\mathcal{O}_{\mathrm{Corrupt}}$ takes ID_i and returns a biometric sample b_i. This oracle captures corruption of a claimant. By querying to this oracle, the adversary can corrupt at most $n - 1$ claimants.
- $\mathcal{O}_{\mathrm{Enroll}}$ takes $(\widehat{PI}, \widehat{AD})$. $\mathcal{O}_{\mathrm{Enroll}}$ updates $\boldsymbol{PI}$ to $\boldsymbol{PI} \cup \{\widehat{PI}\}$, $\boldsymbol{AD}$ to $\boldsymbol{AD} \cup \{\widehat{AD}\}$, and $\boldsymbol{IR}$ to $\boldsymbol{IR} \cup \{\widehat{ID}\}$. Then $\mathcal{O}_{\mathrm{Enroll}}$ returns an index $\widehat{ID}$ of $\widehat{PI}$. This oracle captures addition of a new claimant.
- $\mathcal{O}_{\mathrm{PI}}$ takes ID_i and returns a protected identifier PI_i.
- $\mathcal{O}_{\mathrm{PIC}}$ takes QD and SSD and returns a result $result$. If cr such that $cr \leftarrow \mathsf{PIC}(QD, \boldsymbol{PI}, SSD)$ is not $\perp$, then $result$ is a subset of $\boldsymbol{IR}$ corresponding to cr. Otherwise, $result$ is $\perp$. By querying this oracle, the adversary can check whether QD, which is made by himself is acceptable or not.

Challenge. $\mathcal{C}$ generates and sends session specific data SSD^* [1] to $\mathcal{A}$.

Phase 2. $\mathcal{A}$ is permitted to make queries to $\mathcal{O}_{\mathrm{Corrupt}}$, $\mathcal{O}_{\mathrm{Enroll}}$, $\mathcal{O}_{\mathrm{PI}}$, and $\mathcal{O}_{\mathrm{PIC}}$ as same as Phase 1. Finally, $\mathcal{A}$ generates QD^* and outputs it with SSD^*.

We define a set of identities not queried to $\mathcal{O}_{\mathrm{Corrupt}}$ and not returned by $\mathcal{O}_{\mathrm{Enroll}}$ by U_{NC}. Then we define the unforgeability against the attacks with PIs as follows.

Definition 1 (Unforgeability against Attacks with PIs) *Let $cr^* \leftarrow \mathsf{PIC}(QD^*, \boldsymbol{PI}, SSD^*)$. In the above game, the adversary $\mathcal{A}$ wins if there exists $ID_i \in result \cap U_{\mathrm{NC}}$, where $result \leftarrow \mathsf{DS}(\boldsymbol{IR}, cr^*)$. A biometric authentication system is said to be unforgeable against the attacks with PIs if for an arbitrary polynomial time adversary $\mathcal{A}$ in the security parameter, the probability $\Pr[\mathcal{A}\ wins]$ is negligible.*

We show that if a biometric authentication system satisfies unforgeability, then the system also satisfies authorized-leakage irreversibility [Si12]. Let A be an adversary who can break authorized-leakage irreversibility. That is, A can compute fs, which matches the unprotected template in the unprotected system from PI. Then, by invoking A, an adversary B can obtain fs and generate new QD by using fs. This means we can construct an adversary B who can break unforgeability by invoking A. Therefore, unforgeability

[1] Hereafter, $SSD^* = \emptyset$ for the system without SSD.

is not weaker than authorized-leakage irreversilibity. Similarly, we can also prove that unforgeability is not weaker than pseudo-authorized-leakage irreversibility [Si12].

We assume that if a sample is accepted in the unprotected system, then it is also accepted in the protected system. Clearly, if a biometric authentication system meets the metric of authorized-leakage irreversibility, then this system also meets the metric of full-leakage irreversiblity [Si12]. Therefore, unforgeability is not weaker than any type of irreversibility.

3.2.2 Unforgeability against Attacks with QDs

Unforgeability against attacks with QDs is defined via the following game involving an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$.

Setup. Same as the setup phase described in Section 3.2.1.

Phase 1. $\mathcal{A}$ is permitted to make queries to $\mathcal{O}_{\mathrm{Corrupt}}$, $\mathcal{O}_{\mathrm{Enroll}}$, $\mathcal{O}_{\mathrm{QD}}$, and $\mathcal{O}_{\mathrm{PIC}}$, where $\mathcal{O}_{\mathrm{Corrupt}}$, $\mathcal{O}_{\mathrm{Enroll}}$, and $\mathcal{O}_{\mathrm{PIC}}$ are described in Section 3.2.1. $\mathcal{O}_{\mathrm{QD}}$ takes ID and returns query data QD such that for some SSD, $PI_{ID} \in cr$, where $cr \leftarrow \mathsf{PIC}(\boldsymbol{PI}, QD, SSD)$.

Challenge. $\mathcal{C}$ generates and sends session specific data SSD^* to $\mathcal{A}$.

Phase 2. $\mathcal{A}$ is permitted to make queries to $\mathcal{O}_{\mathrm{Corrupt}}$, $\mathcal{O}_{\mathrm{Enroll}}$, $\mathcal{O}_{\mathrm{QD}}$, and $\mathcal{O}_{\mathrm{PIC}}$. Finally, $\mathcal{A}$ generates QD^* and outputs it with SSD^*.

We define the unforgeability against attacks with QDs as follows.

Definition 2 (Unforgeability against Attacks with QDs) *Let $cr^* \leftarrow \mathsf{PIC}(\boldsymbol{PI}, QD^*, SSD^*)$. U_{NC} is defined in Section 3.2.1. In the above game, the adversary $\mathcal{A}$ wins if there exists $ID_i \in result \cap U_{\mathrm{NC}}$, where $result \leftarrow \mathsf{DS}(\boldsymbol{IR}, cr^*)$. A biometric authentication system is said to be unforgeable against attacks with QDs if for an arbitrary polynomial time adversary $\mathcal{A}$ in the security parameter, the probability $\Pr[\mathcal{A}\ wins]$ is negligible.*

We can also define three types of irreversibility (i.e., full-leakage, authorized-leakage, and pseudo-authorized-leakage irreversibility) against attacks with QDs in the similar manner as Simoens et al. [Si12]. We can show that unforgeability is not weaker than any of irreversibility in the same manner as Section 3.2.1.

3.2.3 Unforgeability against Attacks with PIs and QDs

Unforgeability against attacks with PIs and QDs is defined via the following game involving an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$.

Setup. Same as the setup phase described in Section 3.2.1.

Phase 1. $\mathcal{A}$ is permitted to make queries to $\mathcal{O}_{\mathrm{Corrupt}}$, $\mathcal{O}_{\mathrm{Enroll}}$, $\mathcal{O}_{\mathrm{PI}}$, $\mathcal{O}_{\mathrm{QD}}$, $\mathcal{O}_{\mathrm{SSD}}$, and $\mathcal{O}_{\mathrm{PIC}}$. $\mathcal{O}_{\mathrm{Corrupt}}$, $\mathcal{O}_{\mathrm{Enroll}}$, $\mathcal{O}_{\mathrm{PI}}$, $\mathcal{O}_{\mathrm{QD}}$, and $\mathcal{O}_{\mathrm{PIC}}$ are described in Section 3.2.1 and 3.2.2. $\mathcal{O}_{\mathrm{SSD}}$ takes QD as input and outputs SSD, which is used for generating QD. By querying this oracle, the adversary can obtain SSD, which is secretly shared between the (non-corrupted) client and the server.

Challenge. $\mathcal{C}$ generates session specific data SSD^* and sends it to $\mathcal{A}$.

Phase 2. $\mathcal{A}$ is permitted to make queries to $\mathcal{O}_{\mathrm{Corrupt}}$, $\mathcal{O}_{\mathrm{Enroll}}$, $\mathcal{O}_{\mathrm{PI}}$, $\mathcal{O}_{\mathrm{QD}}$, $\mathcal{O}_{\mathrm{SSD}}$, and $\mathcal{O}_{\mathrm{PIC}}$. Finally, $\mathcal{A}$ generates QD^* and outputs it with SSD^*.

We define the unforgeability against attacks with PIs and QDs as follows.

Definition 3 (Unforgeability against Attacks with PIs and QDs) *Let* $cr^* \leftarrow \mathsf{PIC}(\boldsymbol{PI}, QD^*, SSD^*)$. U_{NC} *is as defined in Section 3.2.1. In the above game, the adversary* $\mathcal{A}$ *wins if there exists* $ID_i \in result \cap U_{\mathrm{NC}}$, *where* $result \leftarrow \mathsf{DS}(\boldsymbol{IR}, cr^*)$. *A biometric authentication system is said to be unforgeable against the attacks with* PIs *and* QDs *if for an arbitrary polynomial time adversary* $\mathcal{A}$ *in the security parameter, the probability* $\Pr[\mathcal{A}\ wins]$ *is negligible.*

We can also define three types of irreversibility (i.e., full-leakage, authorized-leakage, and pseudo-authorized-leakage irreversibility) against attacks with PIs and QDs by the similar manner as [Si12]. We can show that unforgeability is not weaker than any type of irreversibility in the same manner as in Section 3.2.1.

4 Proposed Scheme

In this section, we evaluate the security of the scheme in [Tu05] under classification of adversaries. We show that their scheme is not irreversible against the adversaries with PIs and QDs. Then we propose a scheme that is irreversible against all types of adversaries.

We introduce some notations to describe the proposed scheme. Let $x = (x_{n-1}x_{n-2}\cdots x_0)_2$ and $y = (y_{n-1}y_{n-2}\cdots y_0)_2$ be two n bit integers. Then, we denote the hamming distance between x and y by $d_H(x, y) = \sum_{i=0}^{n-1} |x_i - y_i|$.

A linear binary error correcting code ECC with parameters (K, s, d) consists of two algorithms: ENCODE and DECODE. ENCODE takes s-bit data x as input and outputs its K bits codeword CW, written as $CW \leftarrow \mathsf{ENCODE}(x)$. DECODE takes a K bits codeword CW' as input and outputs s-bit x' or $\perp$, written as $\{x', \perp\} \leftarrow \mathsf{DECODE}(CW')$. If $d_H(CW, CW') \leq d$, then $x' = x$. We write $CW \leftarrow \mathsf{ENCODE}(x)$ and $x' \leftarrow \mathsf{DECODE}(CW')$, respectively. A BCH code is employed, which is a linear binary ECC. Note that a BCH code possesses the following property: for all information symbols x_1 and x_2, $\mathsf{ENCODE}(x_1) \oplus \mathsf{ENCODE}(x_2) = \mathsf{ENCODE}(x_1 \oplus x_2)$.

We denote an inner product operation of two vectors $\mathbf{v}$ and $\mathbf{w}$ by $\langle \mathbf{v}, \mathbf{w} \rangle$ where the length of the result $\langle \mathbf{v}, \mathbf{w} \rangle$ is s. Note that by treating $\mathbf{v}$ and $\mathbf{w}$ as vectors over GF(2^s), we can compute an inner product with n-bit output efficiently. Also note that it is easy to see that for every $\mathbf{v}$, $\mathbf{w}_1$, and $\mathbf{w}_2$, $\langle \mathbf{v}, \mathbf{w}_1 \rangle \oplus \langle \mathbf{v}, \mathbf{w}_2 \rangle = \langle \mathbf{v}, \mathbf{w}_1 \oplus \mathbf{w}_2 \rangle$.

4.1 Security Evaluation of Scheme in [Tu05] under Classification of Adversaries

We evaluate the security of the scheme in [Tu05] with secure channel connection between the client and the server, because their scheme requires only a biometric characteristic for identification and is suitable for our model. A straightforward adoption of their scheme to our model is as follows.

ENROLL: The client C with a biometric sample b generates a feature set z by $\mathsf{FE}(b)$. Then, C chooses a random $r \in \{0,1\}^s$ and computes $W_1 = \mathsf{ENCODE}(r) \oplus z$ and $W_2 = H(r)$. C sends (W_1, W_2) to the server S as his PI. Note that we assume W_1 is held in the server S, while no explicit designation in [Tu05].
IDENTIFY: C with a biometric sample b' generates a feature set z' by $\mathsf{FE}(b')$. Then, C sends z' to S as QD. S checks $H(\mathsf{DECODE}(W_1 \oplus z')) = W_2$. If this equation holds, S outputs $accept$. Otherwise, S outputs $reject$.

If H is a cryptographic hash function, the adversary cannot compute r from W_2. Therefore, the above scheme is irreversible against attacks with PIs. Since the secure channel connection is used, the above scheme can possess irreversibility against attacks with QDs.

However, even if the secure channel is used, S can obtain the query data z'. Since z' is the feature set, the above scheme does not possess irreversibility against attacks with PIs and QDs.

4.2 Construction of Proposed Scheme

We improve the scheme in [Tu05] in several points for achieving irreversibility against attacks with PIs and QDs. Our basic idea of construction is as follows: (1) We replace the hash function H of the scheme in [Tu05] with an xor homomorphic function h such that $h(s) \oplus h(s')$ = $h(s \oplus s')$. (2) In ENROLL protocol, the same as the scheme in [Tu05], the proposed scheme protects a feature set z by xor-ing with a codeword $c = \mathsf{ENCODE}(r)$, for some random number r. (3) In IDENTIFY protocol, the proposed scheme protects a transmitted feature set z' by xor-ing with a codeword $c' = \mathsf{ENCODE}(r')$, for another random number r'. Then, we can verify whether z and z' are close enough by checking if $h(\mathsf{DECODE}((z \oplus c) \oplus (z' \oplus c')))$ is equal to $h(r) \oplus h(r')$, without recovering z, z', r, r'.

We describe the proposed scheme. Let $2K$ be the length of a feature set, p an order of a group G, and ℓ the security parameter. The proposed scheme uses an ECC with parameters (K, s, d) and two cryptographic hash functions H_1 and H_2 such that $H_1 : \{0,1\}^* \times \mathbb{F}_p \to \{0,1\}^s$ and $H_2 : \{0,1\}^* \to \{0,1\}^\ell$. We assume that the client C and the server S have the common parameter $c \in \{0,1\}^K$ and g that is a generator of G. We also assume that FE possesses the following properties [2]: (1) If the biometric samples b and b' are generated by the same claimant, then $d_H(z_1, z_1') \leq d$ and $d_H(z_2, z_2') \leq d$, where $z_1 \| z_2 \leftarrow \mathsf{FE}(b)$ and $z_1' \| z_2' \leftarrow \mathsf{FE}(b')$. (2) If the biometric samples b and b' are generated by the different claimants, then $d_H(z_1, z_1') > d$ and $d_H(z_2, z_2') > d$.

[2] This assumption on the feature extractor is stronger than that in [Tu05]. However, the feature extractor we need can be easily constructed by using the feature extractor in [Tu05] twice.

Note that there is no AD in the proposed scheme. Therefore, PIE does not output AD and no ADs are input in PIR or PIC.

ENROLL: The client C with a biometric sample b wants to be registered in the biometric system of the server S. We set $param = (H_1, H_2, K, s, d, c)$. Then, the enrollment protocol $\mathsf{ENROLL}[C(b), S(-)](param)$ is as follows:

1. C generates a feature set $z_1 \| z_2 \in \{0,1\}^{2K}$ by $\mathsf{FE}(b)$.
2. C runs the algorithm $\mathsf{PIE}(z_1 \| z_2)$ as follows:
   (a) C chooses $r_1 \in \{0,1\}^s$ uniformly at random and computes $\mathsf{ENCODE}(r_1)$. C also computes $V_1 = \mathsf{ENCODE}(r_1) \oplus z_1$.
   (b) C chooses $r_2 \in \mathbb{Z}_p$ uniformly at random and computes $V_2 = H_2(\langle c, \mathsf{ENCODE}(r_1)\rangle \oplus H_1(V_1, g^{r_2}))$. C also computes $V_3 = \mathsf{ENCODE}(H_1(V_1, g^{r_2})) \oplus z_2$.
   (c) C sets and sends $PI = (V_1, V_2, V_3)$ to S as a protected template.
3. S stores them with original identity reference ID in its storage and outputs ID.

IDENTIFY: We assume that S has the set of protected template $\boldsymbol{PI} = \{PI_1, PI_2, \ldots, PI_N\}$, where N is the number of enrolled users and each $PI_i = (V_{1,i}, V_{2,i}, V_{3,i})$.

Before the identification phase starts, C and S collaboratively generate a shared secret g_{sc} by using a key exchange protocol such as the Diffie-Hellman key exchange protocol [DH76]. We set $SSD = g_{sc}$.

Then, the identification protocol $\mathsf{IDENTIFY}[C(b'), S(\boldsymbol{PI})](param, -, g_{sc})$ is as follows:

1. C who has a biometric sample b' generates feature sets $z'_1 \| z'_2 \in \{0,1\}^{2K}$ by $\mathsf{FE}(b')$.
2. C executes the algorithm $\mathsf{PIR}(z'_1 \| z'_2, -, g_{sc})$ as follows:
   (a) C chooses $r_3, r_4 \in \{0,1\}^s$ uniformly at random and computes $r_5 = r_3 \oplus r_4$.
   (b) C computes $W_1 = \mathsf{ENCODE}(r_5) \oplus z'_1$, $W_2 = \langle c, \mathsf{ENCODE}(r_3)\rangle \oplus H_1(W_1, g_{sc})$, and $W_3 = \mathsf{ENCODE}(\langle c, \mathsf{ENCODE}(r_4)\rangle) \oplus z'_2$.
   (c) C sets and sends $QD = (W_1, W_2, W_3)$ to S.
3. S runs the algorithm $\mathsf{PIC}(QD, \boldsymbol{PI}, g_{sc})$ as follows:
   (a) S computes $H_1(W_1, g_{sc})$ and sets $i = 1$. For $PI_i = (V_{1,i}, V_{2,i}, V_{3,i})$, S computes as follows:
      i. S computes $W_1 \oplus V_{1,i}$ and $W_3 \oplus V_{3,i}$.
      ii. S generates $WV_{1,i} = \mathsf{DECODE}(W_1 \oplus V_{1,i})$ and $WV_{3,i} = \mathsf{DECODE}(W_3 \oplus V_{3,i})$. If $WV_{1,i}$ or $WV_{3,i}$ is $\perp$, then sets $i = i + 1$ and goes to the step 3(a)i. Otherwise, goes to the next step.
      iii. S computes $WV_{2,i} = H_2\left(H_1(W_1, g_{sc}) \oplus \langle c, \mathsf{ENCODE}(WV_{1,i})\rangle \oplus WV_{3,i} \oplus W_2\right.$
      iv. If $V_{2,i} = WV_{2,i}$, then outputs $res = accept$ and corresponding ID_i. Otherwise, if $i = N$, then outputs $res = \perp$. If not, sets $i = i + 1$ and goes to the step 3(a)i.

(b) S outputs res. If $res = accept$ then S also outputs ID_i.

Correctness. Let $PI = (V_1, V_2, V_3)$ be a protected template from the biometric samples b and $QD = (W_1, W_2, W_3)$ be verification data from b' in the proposed system. We can prove that if b and b' are generated by the same client, then the following equation holds.

$$V_2 = H_2\left(H_1(W_1, g_{sc}) \oplus \langle c, \mathsf{ENCODE}(WV_1)\rangle \oplus WV_3 \oplus W_2\right), \tag{1}$$

where $WV_1 = \mathsf{DECODE}(W_1 \oplus V_1)$ and $WV_3 = \mathsf{DECODE}(W_3 \oplus V_3)$. From the assumption on the algorithm FE, $d_H(z_1, z_1') \leq d$ and $d_H(z_2, z_2') \leq d$ where $z_1 \| z_2 \leftarrow \mathsf{FE}(b)$ and $z_1' \| z_2' \leftarrow \mathsf{FE}(b')$. Therefore, $WV_1 = r_1 \oplus r_5$ and $WV_3 = H_1(V_1, g^{r_2}) \oplus \langle c, \mathsf{ENCODE}(r_4)\rangle$. Then, the right-hand side of the equation (1) is

$$\begin{aligned}
&H_2(H_1(W_1, g_{sc}) \oplus \langle c, \mathsf{ENCODE}(r_1 \oplus r_5)\rangle \oplus H_1(V_1, g^{r_2}) \oplus \langle c, \mathsf{ENCODE}(r_4)\rangle \\
&\qquad \oplus \langle c, \mathsf{ENCODE}(r_3)\rangle \oplus H_1(W_1, g_{sc})) \\
&= H_2(\langle c, \mathsf{ENCODE}(r_1 \oplus r_3 \oplus r_4 \oplus r_5)\rangle \oplus H_1(V_1, g^{r_2})) \\
&= V_2.
\end{aligned}$$

4.3 Intuition behind the Proposed Scheme

The security of the proposed scheme has not been proven to be connected to any infeasible problems. Here, we intuitively explain the security of the scheme instead.

Let the hash functions H_1 and H_2 be collision resistant. We assume that it is difficult to compute g^{ab} from g^a and g^b over G. [3] Then, the security of the proposed scheme can be discussed as follows.

To compute $z_1 \| z_2$ from PI, an adversary needs to compute r_1 and $H_1(V_1, g^{r_2})$ from PI. However, since H_2 is a cryptographic hash function, the adversary cannot compute r_1 and $H_1(V_1, g^{r_2})$. Therefore, we can say that the proposed scheme is irreversible against the attacks with PIs.

Similarly, we can say that the proposed scheme is irreversible against the attacks with QDs. To compute $z_1' \| z_2'$ from QD, an adversary needs to compute r_4 and r_5. Then the adversary cannot compute g_{sc} and r_4 and r_5 are uniformly random.

The compromised servers obtain not only PIs and QDs, but also $SSD(= g_{sc})$. From some $QD = (W_1, W_2, W_3)$ and g_{sc}, the adversary obtains the following equation:

$$\mathsf{ENCODE}(\langle c, W_1\rangle) \oplus \mathsf{ENCODE}(W_2) \oplus W_3 \oplus \mathsf{ENCODE}(H_1(W_1, g_{sc})) = \mathsf{ENCODE}(\langle c, z_1'\rangle) \oplus z_2'.$$

The above equation can be seen as a system of at most K equations with $2K$ unknowns (since both z_1' and z_2' are K bits). Therefore, we can say that the proposed scheme is irreversible against the attacks with the compromised servers.

[3] This is a standard assumption in cryptography: *Computational Diffie-Hellman (CDH)* assumption.

5 Conclusion

We proposed a classification of adversaries on biometric authentication with template protection into three types in accordance with their knowledge: (1) PIs, (2) QDs, and (3) PIs and QDs. We also presented the security definitions against each type of adversary and showed that the scheme of Tuyls et al. [Tu05] is not irreversible against the third. Then we proposed a biometric authentication scheme that is irreversible against all types of adversaries.

Constructing a scheme possessing unforgeability is our future work.

References

[Ca07] Cappelli, R., Lumini, A., Maio, D., and Maltoni, D. Fingerprint Image Reconstruction from Standard Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1489–1503, 2007.

[DH76] Diffie, W. and Hellman, M. E. New Directions in Cryptography. *IEEE Transactions on Information Theory*, IT-22 No. 6:644–654, 1976.

[IO12] Inuma M. and Otsuka A. Relations among Security Metrics for Template Protection Algorithms, 2012. arXiv:1212.4195, `http://arxiv.org/abs/1212.4195`

[ISO11] ISO/IEC 24745 Information Technology - Security Techniques - Biometric Information Protection, 2011.

[Ma03] Maltoni, D., Maio, D., Jain, A. K., and Prabhakar, S. *Handbook of Fingerprint Recognition*. Springer-Verlag NewYork, 2003.

[NNJ10] Nagar, A., Nandakumar, K., and Jain, A. K. Biometric Template Transformation: a security analysis. *SPIE Media Forensics and Security Media Forensics and Security II*, pages 99–100, 2010.

[STP09] Simoens. K., Tuyls, P., and Preneel, B. Privacy Weaknesses in Biometric Sketches. In *Proceedings of 2009 IEEE Symposium on Security and Privacy*, pages 188–203, 2009.

[Si12] Simoens, K., Yang, B., Beato, F., Busch, C., Newton, E., and Preneel, B. Criteria Towards Metrics for Benchmarking Template Protection Algorithms. In *Proceedings of 2012 5th IAPR International Conference on Biometrics*, pages 498–505, 2012.

[Tu05] Tuyls, P., Akkermans, A., Kevenaar, T., Schrijen, G. J., Bazen, A., and Veldhuis, R. Practical Biometric Authentication with Template Protection. In *Proceedings of 2005 5th International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 436–446, 2005.

[Wa11] Wang, W., Rane, S., Draper, C. S., and Ishwar, P. An Information-Theoretic Analysis of Revocability and Reusability in Secure Biometrics. In *Proceedings of 2011 Information Theory and Applications Workshop(ITA2011)*, pages 1–10, 2011.

[Zh11] Zhou, X. *Privacy and Security Assessment of Biometric Template Protection*. PhD thesis, Technische Universitt Darmstadt, 2011.