# The Cryptographic Layer of Biometric Authentication

Keng-Yu Chen[1] and Serge Vaudenay[1]

EPFL, Lausanne, Switzerland

**Abstract.** In this paper, we focus on the cryptographic layer for biometric authentication. The layer is added on the top of an authentication scheme for security and privacy reasons. We first formalize a biometric authentication scheme and propose security models for two security properties of interest: *unforgeability* and *indistinguishability*. Unforgeability refers to an adversary's ability to impersonate a user, while indistinguishability evaluates an adversary's knowledge of users' biometrics, related to privacy preservation. We then introduce generic constructions using a digital signature scheme and a public-key encryption scheme to achieve the two security properties, respectively. To overcome the limitations of these generic constructions, we further analyze existing instantiations of biometric authentication built on the cryptographic primitives: function-hiding inner product functional encryption. Our results demonstrate conditions for the biometrics and the cryptographic instantiations under which these schemes achieve security within our security model.

## 1 Introduction

Biometric authentication offers an error-tolerant and user-friendly approach to identity verification. Unlike traditional methods such as passwords or tokens, biometrics provide a natural means of authentication: they are inherently tied to an individual and do not require users to remember secrets or carry additional devices. Despite its convenience, biometric authentication introduces unique challenges. Verification requires comparing the similarity of enrolled and probed data rather than testing for exact equality, which rules out straightforward methods such as hashing templates for comparison. Additionally, unlike user-defined passwords, biometrics reveal sensitive personal information and cannot be changed if compromised, raising significant privacy concerns. Furthermore, the probabilistic nature of biometric matching can lead to a false acceptance rate that is not negligible. These issues make the design and security analysis of biometric authentication schemes challenging and highlight the importance of a rigorous study in this domain.

**Previous Work.** Previous works have demonstrated several potential cryptographic primitives that can be utilized to instantiate a biometric authentication scheme, such as function-hiding inner product functional encryption (fh-IPFE)

[14,16,8,6,12], homomorphic encryption [13,24,20], fuzzy extractor [4,17], oblivious transfer [5], relational hash [18], etc. Some of them provide non-interactive protocols in the sense that only the clients transmit enrollment and probe messages to the server before the server decides the authentication results. On the other hand, an interactive protocol allows the server to send hints or challenges to the clients during the authentication process.

**Contribution.** In this paper, we present the following contributions:

- **General Framework.** We introduce a new general framework for analyzing non-interactive biometric authentication schemes. The framework formalizes a biometric authentication scheme by splitting it into two layers: the *biometric layer* and the *cryptographic layer*. The biometric layer accounts for collecting biometric data from users, comparing the closeness of enrolled and probed biometric templates, and deciding the authentication result. The cryptographic layer, on the other hand, protects user privacy and strengthens the security of the scheme.
  Existing works either devote little attention to the biometric layer or assume simplified biometric distributions. While such assumptions facilitate analysis, real-world biometric distributions are intractable and vary across different types of biometrics, feature extraction algorithms, and devices running these algorithms. Our framework accommodates more general distributions and identifies necessary conditions on the biometric and cryptographic layers to achieve different security properties.
- **Unforgeability and indistinguishability.** We formalize two security games to model two security notions that we consider relevant to biometric authentication: the unforgeability (UF) game and the indistinguishability (IND) game. The UF game models an adversary's ability to impersonate a legitimate user by submitting a (possibly invalid) probe message that results in successful authentication, which is similar to the unforgeability notion in [18] and the malicious adversary in [12]. However, we consider several options for the adversary to add more flexibility to our security model. In the real world, an authentication scheme comprises several components, such as the feature extraction device, the system executing the cryptographic algorithms, the biometric database, and the server deciding the authentication result. Our flexible modeling options allow us to capture an adversary's ability to impersonate a user in different scenarios.
  The IND game evaluates an adversary's ability to identify a user's biometrics. While prior works [18,16,8,12] consider similar notions by considering an adversary who has enrollment and probe messages, our model provides the adversary with oracles to users' biometrics and asks it to determine which one is used in the authentication process. This formulation accounts for biometric distributions and supports multiple adversarial capabilities, similar to the UF game.
- **Generic transformations.** We present generic transformations that upgrade an existing biometric authentication scheme to satisfy UF or IND

security. The transformation for the UF security employs a digital signature scheme, while the transformation for the IND security employs a public-key encryption scheme. Both transformations are applicable to any scheme that satisfies some baseline conditions and incur only minimal overheads compared to the original construction.
- **Analysis of existing instantiations.** We analyze the UF and IND security of existing instantiations of biometric authentication schemes from previous works. In particular, we study function-hiding inner-product functional encryption (fh-IPFE) [14,16,8,6,12] as an instantiation of the cryptographic layer. We show that the schemes instantiated by fh-IPFE provide stronger security guarantees than the generic transformations alone, and that applying our transformations to them further strengthens its security at little cost compared to its original complexity.

**Limitation.** Our framework focuses on non-interactive biometric authentication schemes. In general, an interactive protocol may reduce the computational overheads and support more flexible authentication policies. For example, a protocol using fully homomorphic encryption allows the server to compute complex matching functions, while a fh-IPFE-based protocol only supports inner product computations. We leave the study of interactive protocols as future work.

**Structure of the Paper.** In Section 2, we formally define a biometric authentication scheme, including the biometric layer and cryptographic layer. In Section 3, we introduce the unforgeability (UF) game and the indistinguishability (IND) game. We also provide generic transformations that upgrade an existing biometric authentication scheme to satisfy UF or IND security. In Section 4, we recall an instantiation using fh-IPFE and provide analyses of the UF and IND security of it.

**Notation.** In this paper, we use $\lambda$ as the security parameter. $[m]$ denotes the set of integers $\{1, 2, \cdots, m\}$. $\mathbb{Z}_q$ is the finite field modulo a prime number $q$. Given a vector $\mathbf{x} \in \mathbb{Z}_q^n$, we write $x_i$ for its $i$-th entry. For two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_q^n$, $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{n} x_i \cdot y_i \bmod q$ is the canonical inner product of $\mathbf{x}$ and $\mathbf{y}$. $\mathsf{poly}(\lambda)$ ($\mathsf{negl}(\lambda)$) is the class of polynomial (negligible) functions in $\lambda$, and we also use $\mathsf{poly}(\lambda)$ ($\mathsf{negl}(\lambda)$) to represent an arbitrary polynomial (negligible) function. Unless otherwise specified, all algorithms run in PPT. Finally, we write sampling a value $r$ from a distribution $\mathcal{D}$ as $r \leftarrow\!\!\$\ \mathcal{D}$. If $S$ is a finite set, then $r \leftarrow\!\!\$\ S$ means sampling $r$ uniformly from $S$.

## 2 Formalization

We consider a biometric authentication scheme composed of a biometric layer and a cryptographic layer. The biometric layer is responsible for extracting biometric templates from users and comparing them, while the cryptographic layer

is added on top of the biometric layer to protect the security and privacy of users.

### 2.1 Biometric Layer

We first define how we simulate biometric distributions of users. Assume the existence of a family $\mathbb{B}$ of biometric distributions. More formally, each element in $\mathbb{B}$ is a continuous distribution $\mathcal{B}$ with a distribution ensemble $\{\mathcal{B}_\lambda\}_\lambda$, where $\mathcal{B}_\lambda$ for each $\lambda$ is a discrete distribution that approximates $\mathcal{B}$ with an approximation parameter depending on $\lambda$. Furthermore, we have the following oracles $\mathsf{BioSamp}, \mathsf{BioDel}$, and $\mathsf{getTemp}$ for all algorithms to interact with $\mathbb{B}$.

- $\mathsf{BioSamp}$: This is an oracle that generates a random distribution $\mathcal{B}$ in $\mathbb{B}$. By this we mean providing a reference to a distribution $\mathcal{B}$ which can be used in the following oracles. For simplicity, we write $\mathcal{B} \leftarrow \mathsf{BioSamp}$ as $\mathcal{B} \leftarrow\!\!\$\ \mathbb{B}$.
- $\mathsf{BioDel}(\cdot)$: On input $\mathcal{B}$ a biometric distribution in $\mathbb{B}$, this oracle deletes $\mathcal{B}$ from $\mathbb{B}$. Consequently, no further access to $\mathsf{BioSamp}$ can derive $\mathcal{B}$. For simplicity, we write $\mathsf{BioDel}(\mathcal{B})$ as $\mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}$.
- $\mathsf{getTemp}(1^\lambda, \cdot)$: On input $\mathcal{B}$ a biometric distribution in $\mathbb{B}$, this oracle samples a concrete biometric template that depends on the security parameter $\lambda$ from $\mathcal{B}$. For simplicity, we write $\mathbf{b} \leftarrow \mathsf{getTemp}(1^\lambda, \mathcal{B})$ as $\mathbf{b} \leftarrow\!\!\$\ \mathcal{B}$. We further denote $\mathsf{O}_\mathcal{B}$ the oracle which answers by $\mathsf{getTemp}(1^\lambda, \mathcal{B})$ to any query.

In this work, we assume that all adversaries have access to the three oracles $\mathsf{BioSamp}, \mathsf{BioDel}$, and $\mathsf{getTemp}$ to characterize the knowledge of adversaries about the biometric distributions.

**Definition 1 (Biometric Layer).** *The biometric layer of a biometric authentication scheme associated with a family $\mathbb{B}$ of biometric distributions is composed of the following algorithms.*

- $\mathsf{getEnroll}^{\mathsf{O}_\mathcal{B}}(1^\lambda) \to \mathbf{b}$*: Given oracle $\mathsf{O}_\mathcal{B}$ for some distribution $\mathcal{B}$ in $\mathbb{B}$, it outputs a biometric template $\mathbf{b}$ for enrollment.*[1]
- $\mathsf{getProbe}^{\mathsf{O}_\mathcal{B}}(1^\lambda) \to \mathbf{b}'$*: Given oracle $\mathsf{O}_\mathcal{B}$ for some distribution $\mathcal{B}$ in $\mathbb{B}$, it outputs a biometric template $\mathbf{b}'$ for probe.*[2]
- $\mathsf{BioComp}(\mathbf{b}, \mathbf{b}') \to s$*: Given a biometric template $\mathbf{b}$ from* $\mathsf{getEnroll}$ *and another template $\mathbf{b}'$ from* $\mathsf{getProbe}$*, it outputs a score $s$.*
- $\mathsf{Vrfy}(s) \to r \in \{0, 1\}$*: It is a deterministic algorithm that reads the comparison score $s$ and determines whether this is a successful authentication ($r = 1$) or not ($r = 0$).*

**Definition 2 (True Positive Rate and False Positive Rate).** *Given a biometric layer associated with a family $\mathbb{B}$ of distributions, we define the following probabilities.*

[1] In practice, $\mathsf{getEnroll}$ can collect several biometric samples from a user's biometric distribution $\mathcal{B}$ to create a more accurate template.

[2] In practice, $\mathsf{getProbe}$ often directly outputs the answer from $\mathsf{O}_\mathcal{B}$.

$$\mathsf{TP} := \Pr\left[\begin{array}{l}\mathcal{B} \leftarrow\!\$\ \mathbb{B} \\ \mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda) \\ \mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda)\end{array} : \mathsf{Vrfy}(\mathsf{BioComp}(\mathbf{b}, \mathbf{b}')) = 1\right]$$

$$\mathsf{FP} := \Pr\left[\begin{array}{l}\mathcal{B} \leftarrow\!\$\ \mathbb{B}, \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B} \\ \mathcal{B}' \leftarrow\!\$\ \mathbb{B} \\ \mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda) \\ \mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}'}}(1^\lambda)\end{array} : \mathsf{Vrfy}(\mathsf{BioComp}(\mathbf{b}, \mathbf{b}')) = 1\right]$$

The true positive rate $\mathsf{TP}$ indicates the probability that a legitimate user is successfully authenticated, while the false positive rate $\mathsf{FP}$ indicates the probability that an illegitimate user is incorrectly accepted as legitimate. Ideally, $\mathsf{TP} = 1$ and $\mathsf{FP}$ is negligible. However, due to the inherent variability of biometrics, the false negative rate $1 - \mathsf{TP}$ may be greater than 0, and $\mathsf{FP}$ may not be negligible. Our security model and subsequent analysis account for these limitations.

The fundamental goal of an authentication scheme is to distinguish legitimate users from illegitimate ones. We formalize this requirement as follows.

**Definition 3 (Discriminative Scheme).** *An authentication scheme is discriminative if there exists a positive polynomial* $\mathsf{poly}(\lambda)$ *such that*

$$\mathsf{TP} - \mathsf{FP} > \frac{1}{\mathsf{poly}(\lambda)}.$$

Unless stated otherwise, all schemes considered in this paper are discriminative.

We introduce two examples of biometric layers.

*Euclidean Distance.* In the face recognition system in [21], the authors apply a deep convolution network to embed human faces into a metric space of Euclidean distance. In the context of our biometric layer, $\mathsf{getEnroll}$ and $\mathsf{getProbe}$ both output a vector in $\{0, 1, \ldots, m\}^n$ for some integers $m$ and $n$. Two templates $\mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda)$ and $\mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda)$ are considered to belong to the same person if their Euclidean distance is smaller than a threshold $\tau$; that is, $\|\mathbf{b} - \mathbf{b}'\| \leq \tau$.

*Hamming Distance.* In the iris recognition system in [11,19], human iris images are transformed into binary vectors. In the context of our biometric layer, $\mathsf{getEnroll}$ and $\mathsf{getProbe}$ both output a vector in $\{0, 1\}^n$ for some integer $n$. Two templates $\mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda)$ and $\mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda)$ are considered to belong to the same person if their Hamming distance is smaller than a threshold $\tau$; that is, $\mathsf{HD}(\mathbf{b}, \mathbf{b}') := \#\{i : b_i \neq b'_i\} \leq \tau$.

### 2.2 Cryptographic Layer

**Definition 4 (Cryptographic Layer).** *The cryptographic layer of a biometric authentication scheme associated with a family $\mathbb{B}$ of biometric distributions is composed of the following algorithms.*

- $\mathsf{Setup}(1^\lambda) \to (\mathsf{esk}, \mathsf{psk}, \mathsf{csk})$*: It outputs the enrollment secret key* $\mathsf{esk}$*, probe secret key* $\mathsf{psk}$*, and comparison secret key* $\mathsf{csk}$*.*
- $\mathsf{Enroll}(\mathsf{esk}, \mathbf{b}) \to \mathsf{em}$*: On input a biometric template* $\mathbf{b}$*, it outputs the enrollment message* $\mathsf{em}$*.*
- $\mathsf{Probe}(\mathsf{psk}, \mathbf{b}') \to \mathsf{pm}$*: On input a biometric template* $\mathbf{b}'$*, it outputs the probe message* $\mathsf{pm}$ *.*
- $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm}) \to s$*: It compares the enrollment message* $\mathsf{em}$ *and probe message* $\mathsf{pm}$ *and outputs a score* $s$*.*

**Correctness***: A cryptographic layer is correct if for any biometric distributions $\mathcal{B}$ and $\mathcal{B}'$ in $\mathbb{B}$, let* $(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$*,* $\mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda)$*,* $\mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}'}}(1^\lambda)$*,* $\mathsf{em} \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$*,* $\mathsf{pm} \leftarrow \mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$*. Then*

$$\Pr\left[\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm}) = \mathsf{BioComp}(\mathbf{b}, \mathbf{b}')\right] = 1.$$

In a real-world biometric system, these algorithms may be run by different parties such as a biometric scanner, a user's secure hardware, a trusted authority that issues keys, and the server. In Appendix A, we provide several case studies of how our model can be applied in practice.

As a trivial example, consider the following cryptographic layer for any biometric authentication scheme.

**Definition 5 (Trivial Scheme).** *Given any biometric authentication scheme, there exists a trivial cryptographic layer.*

- $\mathsf{Setup}(1^\lambda)$*:* $\mathsf{esk} = \mathsf{psk} = \mathsf{csk}$ *are all empty strings.*
- $\mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$*: Output* $\mathsf{em} \leftarrow \mathbf{b}$*.*
- $\mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$*: Output* $\mathsf{pm} \leftarrow \mathbf{b}'$*.*
- $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm}) = \mathsf{BioComp}(\mathbf{b}, \mathbf{b}')$*.*

Obviously, this trivial scheme does not add any security on the biometric scheme, or more formally, UF and IND security as defined in Section 3. However, we show how to strengthen this trivial scheme in Sections 3.1 and 3.2. Besides, we also provide an instantiation of a biometric authentication scheme with a cryptographic layer using fh-IPFE in Section 4.2.

## 3 Security Games

In this section, we discuss two security notions of a biometric authentication scheme: *unforgeability* and *indistinguishability*.

### 3.1 Unforgeability

To describe the unforgeability of an authentication scheme, we model the ability of an adversary who tries to impersonate a user. The adversary $\mathcal{A}$ is given auxiliary information $\mathsf{option}$ that depends on our threat model and tries to find a valid probe message $\tilde{\mathbf{z}}$. The whole game $\mathsf{UF}_{\Pi,\mathbb{B},\mathsf{option}}$ is defined in Fig. 1.

$\mathsf{UF}_{\Pi,\mathbb{B},\mathsf{option}}(\mathcal{A})$

$\mathcal{B} \leftarrow\!\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}$
$(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$
$\mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$
$\mathsf{em} \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$
$\tilde{\mathbf{z}} \leftarrow \mathcal{A}(\mathsf{option})$
**if** $\tilde{\mathbf{z}}$ is equal to any output of $\mathsf{O}_{\mathsf{pm}}$ or $\mathsf{O}_{\mathsf{Probe}}$ **then**
  **return** 0
**endif**
$s \leftarrow \mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \tilde{\mathbf{z}})$
**return** $\mathsf{Vrfy}(s)$

Fig. 1: The $\mathsf{UF}_{\Pi,\mathbb{B},\mathsf{option}}$ game for an authentication scheme $\Pi$ over a distribution family $\mathbb{B}$ and $\mathsf{option}$.

The auxiliary information $\mathsf{option}$ can be nothing or include $\mathsf{esk}, \mathsf{psk}, \mathsf{csk}, \mathsf{em}$ or the following oracles:

- $\mathsf{O}_\mathcal{B}$. It outputs a biometric template $\mathbf{b} \leftarrow\!\!\$\ \mathcal{B}$.
- $\mathsf{O}_{\mathsf{pm}}$: It first samples a biometric sample $\mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$ and outputs $\mathsf{pm} \leftarrow \mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$.
- $\mathsf{O}_{\mathsf{Enroll}}(\cdot)$: On input $\mathbf{b}'$ in the range of $\mathsf{getEnroll}$, it outputs the enrollment message $\mathsf{Enroll}(\mathsf{esk}, \mathbf{b}')$.
- $\mathsf{O}_{\mathsf{Probe}}(\cdot)$: On input $\mathbf{b}'$ in the range of $\mathsf{getProbe}$, it outputs the probe message $\mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$.

Note that we require the input to $\mathsf{O}_{\mathsf{Enroll}}$ and $\mathsf{O}_{\mathsf{Probe}}$ to be in the range of their corresponding functions. This not only seperates the power of having the keys $\mathsf{esk}$ and $\mathsf{psk}$ and having the oracles $\mathsf{O}_{\mathsf{Enroll}}$ and $\mathsf{O}_{\mathsf{Probe}}$, but also models the situation where an adversary can access to only a device that runs cryptographic algorithms and checks the validity of the given input. The adversary may still be able to query artificial $\mathbf{b}'$ that are not sampled from $\mathsf{getEnroll}$ or $\mathsf{getProbe}$, but they must be at least in the corresponding range.

**Definition 6 (Unforgeability).** *An authentication scheme $\Pi$ associated with a family $\mathbb{B}$ of distributions is called* $\mathsf{option}$*-unforgeable (*$\mathsf{option}$*-UF) if for any*

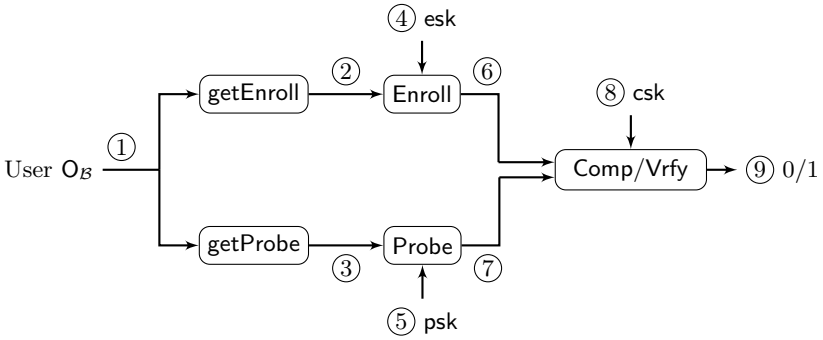



Fig. 2: Oracles in the UF game. $\mathsf{O}_{\mathcal{B}}$: extract ① ; $\mathsf{O}_{\mathsf{pm}}$: extract ⑦ ; $\mathsf{O}_{\mathsf{Enroll}}$: inject ② to extract ⑥ ; $\mathsf{O}_{\mathsf{Probe}}$: inject ③ to extract ⑦. In general, we can consider more flexible choices of option and oracles.

*PPT adversary $\mathcal{A}$, the advantage of $\mathcal{A}$ in the $\mathsf{UF}_{\Pi,\mathbb{B},\mathsf{option}}$ game is*

$$\mathbf{Adv}^{\mathsf{UF}}_{\Pi,\mathbb{B},\mathcal{A},\mathsf{option}} := \Pr[\mathsf{UF}_{\Pi,\mathbb{B},\mathsf{option}}(\mathcal{A}) \to 1] = \mathsf{negl}(\lambda).$$

For the rest of this work, if the scheme $\Pi$, the family $\mathbb{B}$ of distributions, and the auxiliary information option are clear from context, we omit the subscript and write the game as $\mathsf{UF}(\mathcal{A})$. This abbreviation also holds for all other games.

*Choice of* option. We note that if option includes psk, a UF game adversary can compute and return $\mathsf{Probe}(\mathsf{psk}, \mathbf{b}^*)$ for any chosen template $\mathbf{b}^*$. Consequently, to achieve UF security in this setting, we must assume that the biometric layer satisfies the following assumption:

**Assumption 1.** *Assume that the biometric layer is such that for any adversary $\mathcal{A}$,*

$$\Pr\left[\begin{array}{l}\mathcal{B} \leftarrow\!\$\ \mathbb{B}, \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B} \\ \mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda) : \mathsf{Vrfy}(\mathsf{BioComp}(\mathbf{b}, \mathbf{b}^*)) = 1 \\ \mathbf{b}^* \leftarrow \mathcal{A}\end{array}\right] = \mathsf{negl}(\lambda),$$

Recall that all adversaries are assumed to have access to the three oracles defined in Section 2.1 for interacting with $\mathbb{B}$. An adversary in Assumption 1 may sample $\mathcal{B}^* \leftarrow\!\$\ \mathbb{B}$ and return $\mathbf{b}^* \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}^*}}(1^\lambda)$, achieving a winning probability of FP. Alternatively, an adversary can also randomly select $\mathcal{B}^*$ in the original $\mathbb{B}$ set (assuming this can be done without the help of the BioSamp oracle) and win with probability $\frac{\mathsf{TP}}{|\mathbb{B}|} + \mathsf{FP}(1 - \frac{1}{|\mathbb{B}|})$. Therefore, Assumption 1 implies that both FP and $\frac{1}{|\mathbb{B}|}$ are negligible. Furthermore, if psk and $\mathsf{O}_{\mathcal{B}}$ are both given, the adversary can even win with a probability TP. In discriminative schemes, TP is not negligible for sure. Similar vulnerabilities arise if option includes the oracle $\mathsf{O}_{\mathsf{Probe}}$. To prevent these trivial attacks, we add the following requirements:

- $\mathsf{option}$ cannot include $\mathsf{psk}$ unless Assumption 1 holds.
- $\mathsf{option}$ cannot include both $\mathsf{psk}$ and $\mathsf{O}_{\mathcal{B}}$.
- The adversary is not allowed to return the answer returned by $\mathsf{O}_{\mathsf{pm}}$ or $\mathsf{O}_{\mathsf{Probe}}$.

The first two conditions are necessary. As for the last one, a forbidden use of returning the result of $\mathsf{O}_{\mathsf{pm}}$ is necessary but not for $\mathsf{O}_{\mathsf{Probe}}$. If we had allowed using the result of $\mathsf{O}_{\mathsf{Probe}}$, we would need again Assumption 1. To accomodate biometric layers with non-negligible $\mathsf{FP}$, we prohibit the adversary from returning the answer from $\mathsf{O}_{\mathsf{Probe}}$.

*UF Security with Digital Signature.* We note that we can achieve UF security by a similar approach to [12] with a digital signature scheme.

**Theorem 1.** *Let* $\mathsf{option} = \{\mathsf{esk}, \mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathcal{B}}, \mathsf{O}_{\mathsf{Probe}}\}$ *and* $\mathsf{Sig} = (\mathsf{Sig.KGen}, \mathsf{Sig.Sign}, \mathsf{Sig.Vrfy})$ *be an sEUF-CMA secure digital signature scheme. For any authentication scheme* $\Pi$*, the scheme* $\Pi'$ *in Fig. 3 is* $\mathsf{option}$*-UF secure.*

- $\mathsf{Setup}'(1^\lambda)$: Run $(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and $(\mathsf{sk}_{\mathsf{Sig}}, \mathsf{pk}_{\mathsf{Sig}}) \leftarrow \mathsf{Sig.KGen}(1^\lambda)$. Output $\mathsf{esk}' \leftarrow \mathsf{esk}$, $\mathsf{psk}' \leftarrow (\mathsf{psk}, \mathsf{sk}_{\mathsf{Sig}})$, $\mathsf{csk}' \leftarrow (\mathsf{csk}, \mathsf{pk}_{\mathsf{Sig}})$.
- $\mathsf{Enroll}'$: The same as $\mathsf{Enroll}$.
- $\mathsf{Probe}'(\mathsf{psk}', \mathbf{b}')$: Run $\mathsf{pm} \leftarrow \mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$ and $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_{\mathsf{Sig}}, \mathsf{pm})$. Output $\mathsf{pm}' \leftarrow (\mathsf{pm}, \sigma)$.
- $\mathsf{Comp}'(\mathsf{csk}', \mathsf{em}, \mathsf{pm}')$: If $\mathsf{Sig.Vrfy}(\mathsf{pk}_{\mathsf{Sig}}, \mathsf{pm}, \sigma) = 1$, output $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm})$; otherwise, output $\perp$.

Fig. 3: $\Pi'$ obtained by strengthening an authentication scheme $\Pi$ with a digital signature scheme $\mathsf{Sig}$.

An $\mathsf{UF}_{\mathsf{option}}$ adversary has to forge a signature $\sigma$ to win the game, so the scheme is $\mathsf{option}$-UF secure for any $\mathsf{option}$ that does not include $\mathsf{psk}$. Note that Theorem 1 also holds when $\Pi$ is the trivial authentication scheme in Definition 5. So, the real challenge is to reach $\mathsf{psk}$-UF security, which we will in Theorem 5 and 7.

### 3.2 Indistinguishability

In the game of indistinguishability, we model the ability of an adversary who tries to identify the user, which describes the privacy leakage of the scheme. The adversary $\mathcal{A}$ is given oracles to two biometric distributions $\mathcal{B}^{(0)}$ and $\mathcal{B}^{(1)}$ and $\mathsf{option}$ that depends on our threat model. It tries to guess from either $\mathcal{B}^{(0)}$ or $\mathcal{B}^{(1)}$ the enrollment or probe messages are generated. The whole game $\mathsf{IND}_{\Pi,\mathbb{B},\mathsf{option}}$ is defined in Fig. 4.

The auxiliary information $\mathsf{option}$ can be nothing or include $\mathsf{esk}, \mathsf{psk}, \mathsf{csk}, \mathsf{em}$ or the following oracles:

$\mathsf{IND}_{\Pi,\mathbb{B},\mathsf{option}}(\mathcal{A})$

1 : $b \leftarrow\!\$\ \{0,1\}$
2 : $\mathcal{B}^{(0)} \leftarrow\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}^{(0)}$
3 : $\mathcal{B}^{(1)} \leftarrow\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}^{(1)}$
4 : $(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$
5 : $\mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}^{(b)}}}(1^\lambda)$
6 : $\mathsf{em} \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$
7 : $\tilde{b} \leftarrow \mathcal{A}^{\mathsf{O}_{\mathcal{B}^{(0)}}, \mathsf{O}_{\mathcal{B}^{(1)}}}(\mathsf{option})$
8 : **return** $\mathbb{1}_{\tilde{b}=b}$

Fig. 4: The $\mathsf{IND}_{\Pi,\mathbb{B},\mathsf{option}}$ game for an authentication scheme $\Pi$ over a distribution family $\mathbb{B}$ and option.

- $\mathsf{O}_{\mathsf{pm}}$: It first samples a biometric sample $\mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}^{(b)}}}(1^\lambda)$ and outputs $\mathsf{pm} \leftarrow \mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$. We write $\mathsf{O}_{\mathsf{pm}}^{(t)}$ to denote the same oracle that is restricted to $t$ queries.
- $\mathsf{O}_{\mathsf{Enroll}}(\mathsf{esk}, \cdot)$: On input $\mathbf{b}'$, it outputs the enrollment message $\mathsf{Enroll}(\mathsf{esk}, \mathbf{b}')$.
- $\mathsf{O}_{\mathsf{Probe}}(\mathsf{psk}, \cdot)$: On input $\mathbf{b}'$, it outputs the probe message $\mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$.

Note that at least one of em and $\mathsf{O}_{\mathsf{pm}}$ should be given; otherwise, IND security is trivial.

**Definition 7 (Indistinguishability).** *An authentication scheme $\Pi$ associated with a family $\mathbb{B}$ of distributions is called* option*-indistinguishable (*option*-IND) if for any PPT adversary $\mathcal{A}$, the advantage of $\mathcal{A}$ in the $\mathsf{IND}_{\Pi,\mathbb{B},\mathsf{option}}$ game is*

$$\mathbf{Adv}^{\mathsf{IND}}_{\Pi,\mathbb{B},\mathcal{A},\mathsf{option}} := \left| \Pr[\mathsf{IND}_{\Pi,\mathbb{B},\mathsf{option}}(\mathcal{A}) \to 1] - \frac{1}{2} \right| = \mathsf{negl}(\lambda).$$

*Choice of* option*.* We note that some option lead to trivial attacks.

**Theorem 2.** *A discriminative authentication scheme $\Pi$ is not* option*-IND secure for*

- $\mathsf{option} = \{\mathsf{csk}, \mathsf{em}, \ \textit{either}\ \mathsf{psk}\ \textit{or}\ \mathsf{O}_{\mathsf{Probe}}\}$
- $\mathsf{option} = \{\mathsf{csk}, \mathsf{O}_{\mathsf{pm}}, \ \textit{either}\ \mathsf{esk}\ \textit{or}\ \mathsf{O}_{\mathsf{Enroll}}\}$

*Proof.* Let option be the first case that includes em. The second case when $\mathsf{O}_{\mathsf{pm}}$ is given can be analyzed analogously. Consider the adversary $\mathcal{A}$ in the $\mathsf{IND}_{\mathsf{option}}$ game in Algorithm 1. When the challenge bit $b = 0$, the probability that $\mathcal{A}$ wins is TP. When the challenge bit $b = 1$, the probability that $\mathcal{A}$ wins is $1 - \mathsf{FP}$. Now,

$$\mathbf{Adv}^{\mathsf{IND}}_{\Pi,\mathbb{B},\mathcal{A},\mathsf{option}} = \left| \Pr[\mathsf{IND}_{\Pi,\mathbb{B},\mathsf{option}}(\mathcal{A}) \to 1] - \frac{1}{2} \right| = \left| \frac{1}{2}(\mathsf{TP} + 1 - \mathsf{FP}) - \frac{1}{2} \right| > \frac{1}{\mathsf{poly}(\lambda)}.$$

The last inequality comes from the assumption of a discriminative scheme. □

**Algorithm 1** $\mathcal{A}^{\mathsf{O}_{\mathcal{B}^{(0)}},\mathsf{O}_{\mathcal{B}^{(1)}}}(\mathsf{psk},\mathsf{csk},\mathsf{em})$ or $\mathcal{A}^{\mathsf{O}_{\mathcal{B}^{(0)}},\mathsf{O}_{\mathcal{B}^{(1)}},\mathsf{O}_{\mathsf{Probe}}}(\mathsf{csk},\mathsf{em})$

$\mathbf{b}^{(0)} \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}^{(0)}}}(1^\lambda)$
$\mathsf{pm}^{(0)} \leftarrow \mathsf{Probe}(\mathsf{psk},\mathbf{b}^{(0)})$ ▷ or $\mathsf{pm}^{(0)} \leftarrow \mathsf{O}_{\mathsf{Probe}}(\mathbf{b}^{(0)})$
$r \leftarrow \mathsf{Vrfy}(\mathsf{Comp}(\mathsf{csk},\mathsf{em},\mathsf{pm}^{(0)}))$
**return** $1-r$

*IND Security with Public-Key Encryption.* We note that we can achieve IND security with a public-key encryption scheme. Firstly, we recall the *multi-ciphertext IND-CPA* security [2] for a public-key encryption scheme.

**Definition 8 (Multi-ciphertext IND-CPA).** *A public-key encryption scheme* $\mathsf{PKE} = (\mathsf{PKE.KGen},\mathsf{PKE.Enc},\mathsf{PKE.Dec})$ *is multi-ciphertext IND-CPA secure if for any adversary* $\mathcal{A}$*, the advantage of* $\mathcal{A}$ *in the game* $\mathsf{MC\text{-}IND\text{-}CPA}_{\mathsf{PKE}}$ *in Fig. 5 is*

$$\mathbf{Adv}^{\mathsf{MC\text{-}IND\text{-}CPA}}_{\mathsf{PKE},\mathcal{A}} := \left|\Pr[\mathsf{MC\text{-}IND\text{-}CPA}_{\mathsf{PKE}}(\mathcal{A}) \to 1] - \frac{1}{2}\right| = \mathsf{negl}(\lambda).$$

| $\mathsf{MC\text{-}IND\text{-}CPA}_{\mathsf{PKE}}(\mathcal{A})$ | $\mathsf{O}_{\mathsf{MC\text{-}Enc}}(\mathsf{m}_0,\mathsf{m}_1)$ |
|---|---|
| $b \leftarrow\!\!\$\ \{0,1\}$ | $\mathsf{ct} \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}_{\mathsf{PKE}},\mathsf{m}_b)$ |
| $(\mathsf{sk}_{\mathsf{PKE}},\mathsf{pk}_{\mathsf{PKE}}) \leftarrow \mathsf{PKE.KGen}(1^\lambda)$ | **return** ct |
| $\tilde{b} \leftarrow \mathcal{A}^{\mathsf{O}_{\mathsf{MC\text{-}Enc}}}(\mathsf{pk}_{\mathsf{PKE}})$ | |
| **return** $\mathbb{1}_{\tilde{b}=b}$ | |

Fig. 5: The $\mathsf{MC\text{-}IND\text{-}CPA}_{\mathsf{PKE}}$ game for a public-key encryption scheme PKE.

Note that the multi-ciphertext IND-CPA security can reduce to the standard IND-CPA security with a polynomial blowup in the advantage via a hybrid argument.

**Theorem 3.** *Let* $\mathsf{option} = \{\mathsf{esk},\mathsf{psk},\mathsf{em},\mathsf{O}_{\mathsf{pm}}\}$ *and* $\mathsf{PKE} = (\mathsf{PKE.KGen},\mathsf{PKE.Enc},\mathsf{PKE.Dec})$ *be a multi-ciphertext IND-CPA secure public-key encryption scheme. For any authentication scheme* $\Pi$*, the scheme* $\Pi'$ *in Fig. 6 is* option*-IND secure.*

*Proof.* Given an adversary $\mathcal{A}$ in the $\mathsf{IND}_{\mathsf{option}}$ game, consider the reduction adversary $\mathcal{R}$ in Algorithm 2 which plays the $\mathsf{MC\text{-}IND\text{-}CPA}_{\mathsf{PKE}}$ game by running $\mathcal{A}$. $\mathcal{R}$ simulates $\mathsf{O}_{\mathsf{pm}}$ by the following steps.

1. Sample $\mathbf{b'}^{(0)} \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}^{(0)}}}(1^\lambda)$ and $\mathbf{b'}^{(1)} \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}^{(1)}}}(1^\lambda)$.
2. Run $\mathsf{pm}^{(0)} \leftarrow \mathsf{Probe}(\mathsf{psk},\mathbf{b'}^{(0)})$ and $\mathsf{pm}^{(1)} \leftarrow \mathsf{Probe}(\mathsf{psk},\mathbf{b'}^{(1)})$
3. Output $\mathsf{O}_{\mathsf{MC\text{-}Enc}}(1\|\mathsf{pm}^{(0)},1\|\mathsf{pm}^{(1)})$.

- $\mathsf{Setup}'(1^\lambda)$: Run $(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and $(\mathsf{sk}_{\mathsf{PKE}}, \mathsf{pk}_{\mathsf{PKE}}) \leftarrow \mathsf{PKE.KGen}(1^\lambda)$. Output $\mathsf{esk}' \leftarrow (\mathsf{esk}, \mathsf{pk}_{\mathsf{PKE}})$, $\mathsf{psk}' \leftarrow (\mathsf{psk}, \mathsf{pk}_{\mathsf{PKE}})$, $\mathsf{csk}' \leftarrow (\mathsf{csk}, \mathsf{sk}_{\mathsf{PKE}})$.
- $\mathsf{Enroll}'(\mathsf{esk}', \mathbf{b})$: Run $\mathsf{em} \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$ and $\mathsf{ct}_{\mathbf{x}} \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}_{\mathsf{PKE}}, 0\|\mathsf{em})$. Output $\mathsf{em}' \leftarrow \mathsf{ct}_{\mathbf{x}}$.
- $\mathsf{Probe}'(\mathsf{psk}', \mathbf{b}')$: Run $\mathsf{pm} \leftarrow \mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$ and $\mathsf{ct}_{\mathbf{y}} \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}_{\mathsf{PKE}}, 1\|\mathsf{pm})$. Output $\mathsf{pm}' \leftarrow \mathsf{ct}_{\mathbf{y}}$.
- $\mathsf{Comp}'(\mathsf{csk}', \mathsf{em}', \mathsf{pm}')$: First decrypt $b_e\|\mathsf{em} \leftarrow \mathsf{PKE.Dec}(\mathsf{sk}_{\mathsf{PKE}}, \mathsf{em}')$ and $b_p\|\mathsf{pm} \leftarrow \mathsf{PKE.Dec}(\mathsf{sk}_{\mathsf{PKE}}, \mathsf{pm}')$. If $(b_e, b_p) = (0, 1)$, output $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm})$; otherwise, output $\perp$.

Fig. 6: $\Pi'$ obtained by strengthening an authentication scheme $\Pi$ with a public-key encryption scheme PKE.

**Algorithm 2** $\mathcal{R}^{\mathsf{O}_{\mathsf{MC\text{-}Enc}}}(\mathsf{pk}_{\mathsf{PKE}})$

$\mathcal{B}^{(0)} \leftarrow\!\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}^{(0)}$
$\mathcal{B}^{(1)} \leftarrow\!\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}^{(1)}$
$(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$
$\mathbf{b}^{(0)} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}^{(0)}}}(1^\lambda), \mathsf{em}^{(0)} \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b}^{(0)})$
$\mathbf{b}^{(1)} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}^{(1)}}}(1^\lambda), \mathsf{em}^{(1)} \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b}^{(1)})$
$\mathsf{em}' \leftarrow \mathsf{O}_{\mathsf{MC\text{-}Enc}}(0\|\mathsf{em}^{(0)}, 0\|\mathsf{em}^{(1)})$
$\mathsf{esk}' \leftarrow (\mathsf{esk}, \mathsf{pk}_{\mathsf{PKE}}), \mathsf{psk}' \leftarrow (\mathsf{psk}, \mathsf{pk}_{\mathsf{PKE}})$
$\tilde{b} \leftarrow \mathcal{A}^{\mathsf{O}_{\mathcal{B}^{(0)}}, \mathsf{O}_{\mathcal{B}^{(1)}}, \mathsf{O}_{\mathsf{pm}}}(\mathsf{esk}', \mathsf{psk}', \mathsf{em}')$
**return** $\tilde{b}$

Since $\mathcal{R}$ simulates an $\mathsf{IND}_{\mathsf{option}}$ game for $\mathcal{A}$, the advantage of $\mathcal{A}$ is

$$\mathbf{Adv}^{\mathsf{IND}}_{\Pi,\mathbb{B},\mathcal{A},\mathsf{option}} = \mathbf{Adv}^{\mathsf{MC\text{-}IND\text{-}CPA}}_{\mathsf{PKE},\mathcal{R}} = \mathsf{negl}(\lambda).$$

□

Theorem 3 shows that we can achieve option-IND security for any option that does not include csk by a public-key encryption scheme. We also find in Appendix B that for certain biometric layers, which are considered in prior works such as [4,18], we can achieve csk-IND security easily. Therefore, we will mainly focus on the csk-IND security for a more general biometric layer. We show in Section 4.3 that an fh-IPFE establishes csk-IND security for a much broader class of biometric layers.

## 4 Security Analysis: fh-IPFE-based Instantiation

### 4.1 Function-Hiding Inner Product Functional Encryption

**Definition 9 (Function-Hiding Inner Product Functional Encryption (adapted from [14])).** *A function-hiding inner product functional encryption*

*(fh-IPFE) scheme* $\mathsf{FE}$ *over a message space* $\mathbb{Z}_q^n$ *is composed of PPT algorithms* $\mathsf{FE.Setup}$*,* $\mathsf{FE.KGen}$*,* $\mathsf{FE.Enc}$*, and* $\mathsf{FE.Dec}$*:*

- $\mathsf{FE.Setup}(1^\lambda) \to (\mathsf{pp}, \mathsf{msk})$*: It outputs the public parameter* $\mathsf{pp}$ *and the master secret key* $\mathsf{msk}$*.*
- $\mathsf{FE.KGen}(\mathsf{msk}, \mathbf{x}) \to \mathsf{sk}_\mathbf{x}$*: On input a vector* $\mathbf{x} \in \mathbb{Z}_q^n$*, it generates the functional decryption key* $\mathsf{sk}_\mathbf{x}$*.*
- $\mathsf{FE.Enc}(\mathsf{msk}, \mathbf{y}) \to \mathsf{ct}_\mathbf{y}$*: On input a vector* $\mathbf{y} \in \mathbb{Z}_q^n$*, it generates ciphertext* $\mathsf{ct}_\mathbf{y}$*.*
- $\mathsf{FE.Dec}(\mathsf{pp}, \mathsf{sk}_\mathbf{x}, \mathsf{ct}_\mathbf{y}) \to z$*: On input the public parameter* $\mathsf{pp}$*, a functional decryption key* $\mathsf{sk}_\mathbf{x}$*, and a ciphertext* $\mathsf{ct}_\mathbf{y}$*, it outputs either a value* $z \in \mathbb{Z}_q$ *or an error* $\perp$*.*

**Correctness** *An fh-IPFE scheme* $\mathsf{FE}$ *over* $\mathbb{Z}_q^n$ *is correct if, for all non-zero vectors* $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_q^n \setminus \{\mathbf{0}\}$*, the following condition holds. Let* $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{FE.Setup}(1^\lambda)$*,* $\mathsf{sk}_\mathbf{x} \leftarrow \mathsf{FE.KGen}(\mathsf{msk}, \mathbf{x})$*, and* $\mathsf{ct}_\mathbf{y} \leftarrow \mathsf{FE.Enc}(\mathsf{msk}, \mathbf{y})$*, then*

$$\Pr[\mathsf{FE.Dec}(\mathsf{pp}, \mathsf{sk}_\mathbf{x}, \mathsf{ct}_\mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle] = 1.$$

In Appendix C, we recall an fh-IPFE construction in [14] as an example. Given an fh-IPFE scheme $\mathsf{FE}$, we define the $\mathsf{fh\text{-}IND}$ game [14] in Fig. 7.

| $\mathsf{fh\text{-}IND}_{\mathsf{FE}}(\mathcal{A})$ | $\mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}(\mathbf{x}^{(0)} \neq \mathbf{0}, \mathbf{x}^{(1)} \neq \mathbf{0})$ |
|---|---|
| $b \leftarrow\!\!\$ \{0,1\}$ | $\mathsf{sk}_\mathbf{x} \leftarrow \mathsf{FE.KGen}(\mathsf{msk}, \mathbf{x}^{(b)})$ |
| $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{FE.Setup}(1^\lambda)$ | **return** $\mathsf{sk}_\mathbf{x}$ |
| $\tilde{b} \leftarrow \mathcal{A}^{\mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}, \mathsf{O}^{\mathsf{ind}}_{\mathsf{Enc}}}(\mathsf{pp})$ | $\mathsf{O}^{\mathsf{ind}}_{\mathsf{Enc}}(\mathbf{y}^{(0)} \neq \mathbf{0}, \mathbf{y}^{(1)} \neq \mathbf{0})$ |
| **return** $\mathbb{1}_{\tilde{b}=b}$ | $\mathsf{ct}_\mathbf{y} \leftarrow \mathsf{FE.Enc}(\mathsf{msk}, \mathbf{y}^{(b)})$ |
| | **return** $\mathsf{ct}_\mathbf{y}$ |

Fig. 7: The $\mathsf{fh\text{-}IND}_{\mathsf{FE}}$ game for an fh-IPFE scheme $\mathsf{FE}$.

To avoid trivial attacks, we consider *admissible adversaries.*

**Definition 10 (Admissible Adversary).** *Let* $\mathcal{A}$ *be an adversary in an* $\mathsf{fh\text{-}IND}$ *game, and let* $(\mathbf{x}_1^{(0)}, \mathbf{x}_1^{(1)}), \cdots, (\mathbf{x}_{Q_K}^{(0)}, \mathbf{x}_{Q_K}^{(1)})$ *be its queries to* $\mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}$ *and* $(\mathbf{y}_1^{(0)}, \mathbf{y}_1^{(1)})$*,* $\cdots, (\mathbf{y}_{Q_E}^{(0)}, \mathbf{y}_{Q_E}^{(1)})$ *be its queries to* $\mathsf{O}^{\mathsf{ind}}_{\mathsf{Enc}}$*. We say* $\mathcal{A}$ *is admissible if* $\forall i \in [Q_K], \forall j \in [Q_E]$*, we have*

$$\left\langle \mathbf{x}_i^{(0)}, \mathbf{y}_j^{(0)} \right\rangle = \left\langle \mathbf{x}_i^{(1)}, \mathbf{y}_j^{(1)} \right\rangle.$$

**Definition 11 (fh-IND Security).** *An fh-IPFE scheme* $\mathsf{FE}$ *is called fh-IND secure if for any admissible adversary* $\mathcal{A}$*, the advantage of* $\mathcal{A}$ *in the* $\mathsf{fh\text{-}IND}_{\mathsf{FE}/}$ *game is*

$$\mathbf{Adv}^{\mathsf{fh\text{-}IND}}_{\mathsf{FE},\mathcal{A}} := \left| \Pr[\mathsf{fh\text{-}IND}_{\mathsf{FE}}(\mathcal{A}) \to 1] - \frac{1}{2} \right| = \mathsf{negl}(\lambda).$$

We note that fh-IND security is a standard notion for an fh-IPFE, and constructions in [3,10,23,14] are proven fh-IND secure.

### 4.2 Instantiation using an fh-IPFE Scheme

Let $\mathsf{FE} = (\mathsf{FE.Setup}, \mathsf{FE.KGen}, \mathsf{FE.Enc}, \mathsf{FE.Dec})$ be an fh-IPFE scheme. Following [14,12], we can instantiate as follows a biometric authentication scheme $\Pi_E$ using $\mathsf{FE}$ with the Euclidean distance metric and scheme $\Pi_H$ with the Hamming distance metric.

**Instantiation $\Pi_E$** Let $\mathsf{getEnroll}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$ and $\mathsf{getProbe}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$ both output vectors in $\{0, 1, \cdots, m\}^k$ for all biometric distributions $\mathcal{B} \in \mathbb{B}$. For a predefined real number $\tau \geq 0$, let $\mathsf{BioComp}(\mathbf{b}, \mathbf{b}') \to \|\mathbf{b} - \mathbf{b}'\|^2$ and $\mathsf{Vrfy}(s) \to 1$ if $s \leq \tau^2$ and 0 otherwise. Let $\mathsf{FE}$ be an fh-IPFE scheme over $\mathbb{Z}_q^{k+2}$, where $q > m^2 \cdot k$, the maximum possible Euclidean distance. The scheme $\Pi_E$ is given in Fig. 8.

- $\mathsf{Setup}(1^\lambda)$: Run $\mathsf{FE.Setup}(1^\lambda) \to (\mathsf{pp}, \mathsf{msk})$ and output $\mathsf{esk} \leftarrow \mathsf{msk}$, $\mathsf{psk} \leftarrow \mathsf{msk}$, and $\mathsf{csk} \leftarrow \mathsf{pp}$.
- $\mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$: On input a template vector $\mathbf{b} = (b_1, b_2, \cdots, b_k)$, first encode it as $\mathbf{x} = (x_1, x_2, \cdots, x_{k+2}) = (b_1, b_2, \cdots, b_k, 1, \|\mathbf{b}\|^2)$. Next, call $\mathsf{FE.KGen}(\mathsf{msk}, \mathbf{x}) \to \mathsf{sk}_\mathbf{x}$ and output $\mathsf{em} \leftarrow \mathsf{sk}_\mathbf{x}$.
- $\mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$: On input a template vector $\mathbf{b}' = (b'_1, b'_2, \cdots, b'_k)$, first encode it as $\mathbf{y} = (y_1, y_2, \cdots, y_{k+2}) = (-2b'_1, -2b'_2, \cdots, -2b'_k, \|\mathbf{b}'\|^2, 1)$. Next, call $\mathsf{FE.Enc}(\mathsf{msk}, \mathbf{y}) \to \mathsf{ct}_\mathbf{y}$ and output $\mathsf{pm} \leftarrow \mathsf{ct}_\mathbf{y}$.
- $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm})$: Run $\mathsf{FE.Dec}(\mathsf{pp}, \mathsf{sk}_\mathbf{x}, \mathsf{ct}_\mathbf{y}) \to s$ and output the value $s$.

Fig. 8: $\Pi_E$ instantiated by an fh-IPFE scheme $\mathsf{FE}$.

By the correctness of the fh-IPFE scheme $\mathsf{FE}$, we have

$$s = \mathsf{FE.Dec}(\mathsf{pp}, \mathsf{sk}_\mathbf{x}, \mathsf{ct}_\mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{k} -2b_i b'_i + \|\mathbf{b}\|^2 + \|\mathbf{b}'\|^2 = \|\mathbf{b} - \mathbf{b}'\|^2.$$

which is equal to $\mathsf{BioComp}(\mathbf{b}, \mathbf{b}')$.

**Instantiation $\Pi_H$** Let $\mathsf{getEnroll}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$ and $\mathsf{getProbe}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$ both output vectors in $\{0, 1\}^k$ for all biometric distributions $\mathcal{B} \in \mathbb{B}$. For a predefined real number $\tau \geq 0$, let $\mathsf{BioComp}(\mathbf{b}, \mathbf{b}') \to \mathsf{HD}(\mathbf{b}, \mathbf{b}') = \#\{i : b_i \neq b'_i\}$ and $\mathsf{Vrfy}(s) \to 1$ if $s \leq \tau$ and 0 otherwise. Let $\mathsf{FE}$ be over $\mathbb{Z}_q^k$, where $q > k$. The scheme $\Pi_H$ is given in Fig. 9.

By the correctness of the functional encryption scheme $\mathsf{FE}$, we have

$$s = \langle \mathbf{x}, \mathbf{y} \rangle = k - 2\mathsf{HD}(\mathbf{b}, \mathbf{b}').$$

- $\mathsf{Setup}(1^\lambda)$: Run $\mathsf{FE.Setup}(1^\lambda) \to (\mathsf{pp}, \mathsf{msk})$ and output $\mathsf{esk} \leftarrow \mathsf{msk}$, $\mathsf{psk} \leftarrow \mathsf{msk}$, and $\mathsf{csk} \leftarrow \mathsf{pp}$.
- $\mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$: On input a template vector $\mathbf{b} = (b_1, b_2, \cdots, b_k)$, first encode it as $\mathbf{x} = (x_1, x_2, \cdots, x_k) = (2b_1 - 1, 2b_2 - 1, \cdots, 2b_k - 1) \in \{-1, 1\}^k$. Next, call $\mathsf{FE.KGen}(\mathsf{msk}, \mathbf{x}) \to \mathsf{sk}_\mathbf{x}$ and output $\mathsf{em} \leftarrow \mathsf{sk}_\mathbf{x}$.
- $\mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$: On input a template vector $\mathbf{b}' = (b'_1, b'_2, \cdots, b'_k)$, first encode it as $\mathbf{y} = (y_1, y_2, \cdots, y_k) = (2b'_1 - 1, 2b'_2 - 1, \cdots, 2b'_k - 1) \in \{-1, 1\}^k$. Next, call $\mathsf{FE.Enc}(\mathsf{msk}, \mathbf{y}) \to \mathsf{ct}_\mathbf{y}$ and output $\mathsf{pm} \leftarrow \mathsf{ct}_\mathbf{y}$.
- $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm})$: Run $\mathsf{FE.Dec}(\mathsf{pp}, \mathsf{sk}_\mathbf{x}, \mathsf{ct}_\mathbf{y}) \to s$ and output the value $\frac{k-s}{2}$.

Fig. 9: $\Pi_H$ instantiated by an fh-IPFE scheme $\mathsf{FE}$.

We note that the construction in [12] is applying Theorem 1 on instantiation $\Pi_E$ or $\Pi_H$. The user holds $\mathsf{esk}$ and $\mathsf{psk}$ while the server holds $\mathsf{csk}$, the public parameter of the functional encryption scheme. The use case of [12] is described in Appendix A.1.

### 4.3 IND Security

Consider the following definition and assumption on the biometric distribution family $\mathbb{B}$ and the biometric layer of a scheme.

**Assumption 2.** *Let $t$ be an integer and $\mathcal{B} \in \mathbb{B}$. Define the distribution $\mathcal{D}_\mathcal{B}(t)$:*

$$\mathcal{D}_\mathcal{B}(t) = \left(\mathsf{BioComp}(\mathbf{b}, \mathbf{b}^{(1)}), \mathsf{BioComp}(\mathbf{b}, \mathbf{b}^{(2)}), \ldots, \mathsf{BioComp}(\mathbf{b}, \mathbf{b}^{(t)})\right),$$

*where $\mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$ and $\mathbf{b}^{(i)} \leftarrow \mathsf{getProbe}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$ for all $i \in [t]$. Assume that for any two distributions $\mathcal{B}^{(0)}, \mathcal{B}^{(1)} \in \mathbb{B}$, $\mathcal{D}_{\mathcal{B}^{(0)}}(t)$ and $\mathcal{D}_{\mathcal{B}^{(1)}}(t)$ are identical.*

We use Assumption 2 to be able to prove $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}_\mathsf{pm}\}$-IND security only. We could easily relax it a bit by assuming the computational indistinguishability of $\mathcal{D}_{\mathcal{B}^{(0)}}(t)$ and $\mathcal{D}_{\mathcal{B}^{(1)}}(t)$. However, this assumption is necessary as an adversary in this model has access to a $\mathcal{D}_{\mathcal{B}^{(b)}}(t)$ source by $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm}^{(1)}), \ldots,$ $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm}^{(t)})$, where $b$ is the challege bit.

**Theorem 4.** *Let $\Pi = \Pi_E$ or $\Pi_H$ that is discriminative, and let $\mathsf{option} = \{\mathsf{csk}, \mathsf{em}, \mathsf{O}_\mathsf{pm}\}$. For a distribution family $\mathbb{B}$ satisfying Assumption 2 for any $t = \mathsf{poly}(\lambda)$, if $\mathsf{FE}$ is fh-IND secure, then $\Pi$ is $\mathsf{option}$-IND secure.*

*Proof.* We prove the case $\Pi = \Pi_E$. The case for $\Pi = \Pi_H$ can be analyzed similarly.

Given an adversary $\mathcal{A}$ in the $\mathsf{IND}_\mathsf{option}$ game, consider the reduction adversary $\mathcal{R}$ in Algorithm 3 which plays the $\mathsf{fh\text{-}IND}$ game by running $\mathcal{A}$. $\mathcal{R}$ simulates the $i$-th query to $\mathsf{O}_\mathsf{pm}$ by the following steps.

1. Sample $\mathbf{b'}^{(b')} \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}^{(b')}}}(1^\lambda)$ and let $\mathbf{y}^{(b')} \leftarrow (-2{b'_1}^{(b')}, \cdots, -2{b'_k}^{(b')}, \|\mathbf{b'}^{(b')}\|^2, 1)$
2. Compute $d^{(i)} = \langle \mathbf{x}^{(b')}, \mathbf{y}^{(b')} \rangle$
3. Return $\mathsf{ct}_{\mathbf{y}}^{(i)} \leftarrow \mathsf{O}^{\mathsf{ind}}_{\mathsf{Enc}}((d^{(i)}, 0, \ldots, 0, 1), \mathbf{y}^{(b')})$ [3].

Note that for any query $((d^{(i)}, 0, \ldots, 0, 1), \mathbf{y}^{(b')})$ to $\mathsf{O}^{\mathsf{ind}}_{\mathsf{Enc}}$, it satisfies

$$\langle (1, 0, \ldots, 0), (d^{(i)}, 0, \ldots, 0, 1) \rangle = d^{(i)} = \langle \mathbf{x}^{(b')}, \mathbf{y}^{(b')} \rangle.$$

Hence, $\mathcal{R}$ is an admissible adversary. Also note that the distribution of $(d^{(1)}, \cdots, d^{(t)})$ is $\mathcal{D}_{\mathcal{B}}(t)$ for any $t$.

**Algorithm 3** $\mathcal{R}^{\mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}, \mathsf{O}^{\mathsf{ind}}_{\mathsf{Enc}}}(\mathsf{pp})$

1: $\mathcal{B}^{(0)} \leftarrow\!\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}^{(0)}$
2: $\mathcal{B}^{(1)} \leftarrow\!\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}^{(1)}$
3: $b' \leftarrow\!\!\$\ \{0, 1\}$
4: $\mathbf{b}^{(b')} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}^{(b')}}}(1^\lambda), \mathbf{x}^{(b')} \leftarrow (b_1^{(b')}, \cdots, b_k^{(b')}, 1, \|\mathbf{b}^{(b')}\|^2)$
5: $\mathsf{sk} \leftarrow \mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}((1, 0, \ldots, 0), \mathbf{x}^{(b')})$
6: $\tilde{b} \leftarrow \mathcal{A}^{\mathsf{O}_{\mathcal{B}^{(0)}}, \mathsf{O}_{\mathcal{B}^{(1)}}, \mathsf{O}_{\mathsf{pm}}}(\mathsf{pp}, \mathsf{sk})$
7: **return** $\mathbb{1}_{\tilde{b}=b'}$

Let the challenge bit of the fh-IND game be $b$. Suppose $b = 0$, then the input $(\mathsf{pp}, \mathsf{sk})$ of $\mathcal{A}$ does not depend of $b'$, but the simulation of $\mathsf{O}_{\mathsf{pm}}$ depends on it. However, it only dependes on the random vector $(d^{(1)}, \cdots, d^{(t)})$ with distribution $\mathcal{D}_{\mathcal{B}^{(b')}}(t)$. Thanks to Assumption 2, this is independent from $b'$, and therefore $\tilde{b}$ and $b'$ are independent. Hence, $\Pr[\mathcal{R} \to 0 | b = 0] = \frac{1}{2}$.

For $b = 1$, $\mathcal{R}$ perfectly simulates the $\mathsf{IND}_{\mathsf{option}}$ game with a challenge bit denoted by $b'$. Hence, $\Pr[\mathcal{R} \to 1 | b = 1] = \Pr[\mathsf{IND}_{\mathsf{option}}(\mathcal{A}) \to 1]$.

The advantage of $\mathcal{R}$ in the fh-IND game is

$$\left| \Pr[\mathcal{R} \to b] - \frac{1}{2} \right| = \left| \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \Pr[\mathsf{IND}_{\mathsf{option}}(\mathcal{A}) \to 1] - \frac{1}{2} \right| = \frac{1}{2} \left| \mathbf{Adv}^{\mathsf{IND}}_{\Pi, \mathbb{B}, \mathcal{A}, \mathsf{option}} \right|.$$

□

Recall that a secure public-key encryption scheme can give us option-IND security for any option that does not include csk by Theorem 3. By upgrading the scheme $\Pi_E$ or $\Pi_H$ in the same way as Fig. 6, we therefore obtain a scheme that is $\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$-IND and $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$-IND secure at the same time.

### 4.4 Towards psk-UF Security

We are particularly interested in psk-UF security since we can achieve option-UF security for any option excluding psk with a signature scheme by Theorem 1.

[3] We let the last coordinate be 1 to avoid zero-vector queries.

Unfortunately, the current instantiation $\Pi_E$ or $\Pi_H$ even with the transformation of Theorem 1 (and therefore the construction in [12]) does not achieve psk-UF security. Specifically, we list two main problems:

- With msk of the fh-IPFE, the adversary can encrypt a vector that is not a correct encoding, such as $\mathbf{0}$ vector for example. This makes the scheme not option-UF secure when option includes psk. This drawback occurs for both $\Pi_E$ and $\Pi_H$ using any fh-IPFE schemes.
- With msk and a given secret key $\mathsf{sk}_{\mathbf{x}}$ of the fh-IPFE scheme for the biometric template, one can reconstruct the vector $\mathbf{x}$ that $\mathsf{sk}_{\mathbf{x}}$ corresponds to. This lets the adversary learn the biometrics and makes the scheme not option-UF secure when option includes both psk and em.

Our solution to the first problem is to verify that the encrypted vector is correctly encoded during decryption. For the second problem, we apply the technique as Theorem 3, which adds a PKE to the scheme. The security of the PKE guarantees that the adversary cannot learn any information from em. However, unlike Theorem 3, we need a PKE that is not just multi-ciphertext IND-CPA secure but also allows one decryption call.

**Verifying Encoding for $\Pi_H$** For biometrics in the Hamming distance space, we change the scheme $\Pi_H$ into the following scheme $\Pi_{HC}$ in Fig. 10.

- $\mathsf{Setup}(1^\lambda)$: Run $\mathsf{FE.Setup}(1^\lambda) \to (\mathsf{pp}, \mathsf{msk})$. For $i \in [k]$, sample $c_i \leftarrow\!\!\$\ \{-1, 1\}$, let $\mathbf{e}_i = (0, \ldots, 0, \overbrace{c_i}^{i\text{th index}}, 0, \ldots, 0)$, and run $\mathsf{FE.KGen}(\mathsf{msk}, \mathbf{e}_i) \to \mathsf{sk}_i$. Output $\mathsf{esk} \leftarrow \mathsf{msk}$, $\mathsf{psk} \leftarrow \mathsf{msk}$ and $\mathsf{csk} \leftarrow (\mathsf{pp}, \{\mathsf{sk}_i\}_{i\in[k]})$.
- $\mathsf{Enroll}(\mathsf{esk}, \mathbf{b}), \mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$: The same as $\Pi_H$ in Section 4.2.
- $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm})$: First verify that $\mathsf{FE.Dec}(\mathsf{pp}, \mathsf{sk}_i, \mathsf{ct}_{\mathbf{y}}) \in \{-1, 1\}$ for all $i \in [k]$. If so, call $\mathsf{FE.Dec}(\mathsf{pp}, \mathsf{sk}_{\mathbf{x}}, \mathsf{ct}_{\mathbf{y}}) \to s$ and output $\frac{k-s}{2}$. Otherwise, it aborts.

Fig. 10: $\Pi_{HC}$ instantiated by an fh-IPFE scheme FE.

The scheme $\Pi_{HC}$ checks that every coefficient of the encrypted vector is in $\{-1, 1\}$, which ensures that the vector is correctly encoded. To prevent an adversary who owns csk from learning the real coefficients, we randomize the $\{\mathsf{sk}_i\}_{i\in[k]}$ in csk. We show that this achieves IND security if the adversary has only one $\mathsf{O_{pm}}$ query in Theorem 6.

**Theorem 5.** *Let* option $= \{\mathsf{esk}, \mathsf{psk}, \mathsf{csk}\}$. *Given Assumption 1, the authentication scheme $\Pi_{HC}$ is* option*-UF secure.*

*Proof.* Given an adversary $\mathcal{A}$ in the option-UF game, we can build a reduction $\mathcal{R}$ to break Assumption 1. $\mathcal{R}$ runs $\mathsf{FE.Setup}(1^\lambda)$ and generates esk, psk, csk for $\mathcal{A}$.

Due to the range check of each coefficient, an adversary to win the UF game needs to submit an encryption of $\mathbf{y}^*$ whose coefficients lie in $\{-1, 1\}$. The reduction $\mathcal{R}$ can recover the vector $\mathbf{y}^*$ by checking its inner product with elementary vectors and finding the corresponding vector $\mathbf{b}^* \in \{0,1\}^k$. Moreover, $\mathcal{A}$ wins if $\mathsf{HD}(\mathbf{b}, \mathbf{b}^*) \leq \tau$, and therefore by Assumption 1,

$$\mathbf{Adv}^{\mathsf{UF}}_{\mathcal{A},\mathsf{option}} = \Pr\begin{bmatrix}\mathcal{B} \leftarrow\!\!\$\ \mathbb{B}, \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B} \\ \mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda) : \mathsf{HD}(\mathbf{b}, \mathbf{b}^*) \leq \tau \\ \mathcal{R} \to \mathbf{b}^*\end{bmatrix} = \mathsf{negl}(\lambda).$$

□

While the scheme $\Pi_{HC}$ is $\{\mathsf{esk}, \mathsf{psk}, \mathsf{csk}\}$-UF secure, it lost the $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$-IND security as Theorem 4. Instead, we show that it is $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}^{(1)}_{\mathsf{pm}}\}$-IND secure, where we only allow one single query to $\mathsf{O}_{\mathsf{pm}}$ for the adversary.

**Theorem 6.** *Let* $\mathsf{option} = \{\mathsf{csk}, \mathsf{em}, \mathsf{O}^{(1)}_{\mathsf{pm}}\}$. *For a distribution family* $\mathbb{B}$ *satisfying Assumption 2, if* $\mathsf{FE}$ *is fh-IND secure, then* $\Pi_{HC}$ *is* $\mathsf{option}$*-IND secure.*

*Proof.* Given an adversary $\mathcal{A}$ in the $\mathsf{IND}_{\mathsf{option}}$ game, consider the reduction adversary $\mathcal{R}$ in Algorithm 4 which plays the $\mathsf{fh\text{-}IND}$ game by running $\mathcal{A}$. $\mathcal{R}$ simulates $\mathsf{O}^{(1)}_{\mathsf{pm}}$ by the returning $\mathsf{ct}_{\mathbf{y}}$.

**Algorithm 4** $\mathcal{R}^{\mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}, \mathsf{O}^{\mathsf{ind}}_{\mathsf{Enc}}}(\mathsf{pp})$

$\mathcal{B}^{(0)} \leftarrow\!\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}^{(0)}$
$\mathcal{B}^{(1)} \leftarrow\!\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}^{(1)}$
$b' \leftarrow\!\!\$\ \{0,1\}$
$\mathbf{b}^{(b')} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}^{(b')}}}(1^\lambda), \mathbf{x}^{(i)} \leftarrow (2b_1^{(b')} - 1, \ldots, 2b_k^{(b')} - 1)$
$\mathbf{b}'^{(b')} \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}^{(b')}}}(1^\lambda), \mathbf{y}^{(b')} \leftarrow (2b'_1{}^{(b')} - 1, \ldots, 2b'_k{}^{(b')} - 1)$
$d \leftarrow \langle \mathbf{x}^{(b')}, \mathbf{y}^{(b')} \rangle$
$\mathsf{sk} \leftarrow \mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}((1, 0, \ldots, 0), \mathbf{x}^{(b')})$
$\mathsf{ct}_{\mathbf{y}} \leftarrow \mathsf{O}^{\mathsf{ind}}_{\mathsf{Enc}}((d, 1, 0, \ldots, 0), \mathbf{y}^{(b')})$
**for** $i \in [k]$ **do**
    $c_i^{(1)} \leftarrow\!\!\$\ \{-1, 1\}$ and let $\mathbf{e}_i^{(1)} = (0, \ldots, 0, c_i^{(1)}, 0, \ldots, 0)$
    $c_i^{(0)} \leftarrow c_i^{(1)} y_i{}^{(b')}$ and let $\mathbf{e}_i^{(0)} = (0, c_i^{(0)}, 0, \ldots, 0)$
    $\mathsf{sk}_i \leftarrow \mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}(\mathbf{e}_i^{(0)}, \mathbf{e}_i^{(1)})$
**end for**
$\tilde{b} \leftarrow \mathcal{A}^{\mathsf{O}_{\mathcal{B}^{(0)}}, \mathsf{O}_{\mathcal{B}^{(1)}}, \mathsf{O}^{(1)}_{\mathsf{pm}}}((\mathsf{pp}, \{\mathsf{sk}_i\}_{i \in [k]}), \mathsf{sk})$
**return** $\mathbb{1}_{\tilde{b} = b'}$

Note that for either query $((1,0,\ldots,0),\mathbf{x}^{(b')})$ or queries $(\mathbf{e}_i^{(0)},\mathbf{e}_i^{(1)})$ to $\mathsf{O}^{\mathsf{ind}}_{\mathsf{KGen}}$, it satisfies

$$\langle(1,0,\ldots,0),(d,1,0,\ldots,0)\rangle = d = \langle\mathbf{x}^{(b')},\mathbf{y}^{(b')}\rangle$$
$$\langle\mathbf{e}_i^{(0)},(d,1,\ldots,0)\rangle = c_i^{(0)} = c_i^{(1)}y_i^{(b')} = \langle\mathbf{e}_i^{(1)},\mathbf{y}^{(b')}\rangle$$

Hence, $\mathcal{R}$ is an admissible adversary.

Let the challenge bit of the fh-IND game be $b$. Suppose $b=0$, then the input pp and sk of $\mathcal{A}$ do not depend on $b'$, but the simulation of $\mathsf{O}^{(1)}_{\mathsf{pm}}$ and the input $\{\mathsf{sk}_i\}_{i\in[k]}$ do. However, the first only depends on the random value $d$ and is independent from $b'$ thanks to Assumption 2, and the latter only depend on $c_i^{(1)}y_i^{(b')}$, which is also independent from $b'$ thanks to the uniform bit $c_i^{(1)}$. Hence, $\tilde{b}$ and $b'$ are independent, and $\Pr[\mathcal{R}\to 0|b=0]=\frac{1}{2}$.

For $b=1$, $\mathcal{R}$ perfectly simulates the $\mathsf{IND}_{\mathsf{option}}$ game with a challenge bit denoted by $b'$. Hence, $\Pr[\mathcal{R}\to 1|b=1]=\Pr[\mathsf{IND}_{\mathsf{option}}(\mathcal{A})\to 1]$.

The advantage of $\mathcal{R}$ in the fh-IND game is

$$\left|\Pr[\mathcal{R}\to b]-\frac{1}{2}\right| = \left|\frac{1}{2}\cdot\frac{1}{2}+\frac{1}{2}\cdot\Pr[\mathsf{IND}_{\mathsf{option}}(\mathcal{A})\to 1]-\frac{1}{2}\right| = \frac{1}{2}\left|\mathbf{Adv}^{\mathsf{IND}}_{\Pi,\mathbb{B},\mathcal{A},\mathsf{option}}\right|.$$

□

**Achieving {psk, em}-UF Security** Now, we show that we can achieve option-UF security for an option including both psk and em. For this, we first define the multi-ciphertext IND-1CCA security.

**Definition 12 (Multi-Ciphertext IND-1CCA).** *A public-key encryption scheme* $\mathsf{PKE}=(\mathsf{PKE.KGen},\mathsf{PKE.Enc},\mathsf{PKE.Dec})$ *is multi-ciphertext IND-1CCA secure if for any adversary* $\mathcal{A}$ *, the advantage of* $\mathcal{A}$ *in the game* $\mathsf{MC\text{-}IND\text{-}1CCA}_{\mathsf{PKE}}$ *in Fig. 11 is*

$$\mathbf{Adv}^{\mathsf{MC\text{-}IND\text{-}1CCA}}_{\mathsf{PKE},\mathcal{A}} := \left|\Pr[\mathsf{MC\text{-}IND\text{-}1CCA}_{\mathsf{PKE}}(\mathcal{A})\to 1]-\frac{1}{2}\right| = \mathsf{negl}(\lambda).$$

**Theorem 7.** *Let* PKE *be a multi-ciphertext IND-1CCA secure PKE scheme. Let* $\mathsf{option}=\{\mathsf{esk},\mathsf{psk}\}$ *and* $\mathsf{option}'=\{\mathsf{esk},\mathsf{psk},\mathsf{em},\mathsf{O}_{\mathsf{pm}}\}$*. If a discriminative authentication scheme* $\Pi$ *is* option*-UF secure, then the authentication scheme* $\Pi'$ *upgraded by* PKE *as Fig. 6 is* option′*-UF secure.*

*Proof.* Given an adversary $\mathcal{A}'$ in the $\mathsf{UF}_{\Pi',\mathsf{option}'}$ game, consider the reduction adversary $\mathcal{R}$ in Algorithm 5 which plays the MC-IND-1CCA game by running $\mathcal{A}'$. $\mathcal{R}$ simulates $\mathsf{O}_{\mathsf{pm}}$ by the following steps:

1. $\mathbf{b}'_0 \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}_0}}(1^\lambda)$ and $\mathbf{b}'_1 \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}_1}}(1^\lambda)$
2. $\mathsf{pm}_0 \leftarrow \mathsf{Probe}(\mathsf{psk},\mathbf{b}'_0)$
3. $\mathsf{pm}_1 \leftarrow \mathsf{Probe}(\mathsf{psk},\mathbf{b}'_1)$

$\mathsf{MC\text{-}IND\text{-}1CCA}_{\mathsf{PKE}}(\mathcal{A})$

1 : $b \leftarrow\!\!\$\ \{0,1\}$
2 : $\mathfrak{L} \leftarrow \emptyset$
3 : $(\mathsf{sk}_{\mathsf{PKE}}, \mathsf{pk}_{\mathsf{PKE}}) \leftarrow \mathsf{PKE.KGen}(1^\lambda)$
4 : $\tilde{b} \leftarrow \mathcal{A}^{\mathsf{O}_{\mathsf{MC\text{-}Enc}}, \mathsf{O}_{\mathsf{Dec}}}(\mathsf{pk}_{\mathsf{PKE}})$
5 : **return** $\mathbb{1}_{\tilde{b}=b}$

$\mathsf{O}_{\mathsf{MC\text{-}Enc}}(\mathsf{m}_0, \mathsf{m}_1)$

1 : $\mathsf{ct} \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}_{\mathsf{PKE}}, \mathsf{m}_b)$
2 : $\mathfrak{L} \leftarrow \mathfrak{L} \cup \{\mathsf{ct}\}$
3 : **return** ct

$\mathsf{O}_{\mathsf{Dec}}(\mathsf{ct})$ (only once)

1 : **if** $\mathsf{ct} \in \mathfrak{L}$ **then return** $\perp$
2 : $\mathsf{m} \leftarrow \mathsf{PKE.Dec}(\mathsf{sk}_{\mathsf{PKE}}, \mathsf{ct})$
3 : **return** m

Fig. 11: The $\mathsf{MC\text{-}IND\text{-}1CCA}_{\mathsf{PKE}}$ game for a public-key encryption scheme PKE.

4. Output $\mathsf{pm}' \leftarrow \mathsf{O}_{\mathsf{MC\text{-}Enc}}(1\|\mathsf{pm}_0, 1\|\mathsf{pm}_1)$

If the challenge bit for the MC-IND-1CCA game $b = 0$, $\mathcal{R}$ simualtes a $\mathsf{UF}_{\Pi',\mathsf{option}'}$ game for $\mathcal{A}'$, and $\mathcal{R}$ returns 0 when $\mathcal{A}'$ wins. Indeed, to win the $\mathsf{UF}_{\Pi',\mathsf{option}'}$ game, ct should not be any of the previous answer of $\mathsf{O}_{\mathsf{pm}}$. It is also not $\mathsf{em}'$; otherwise, the prefix $\tilde{b}_z = 0$ and the construction in Fig. 6 aborts. Hence, $\mathsf{ct} \notin \mathfrak{L}$ and $\mathsf{O}_{\mathsf{Dec}}$ correctly decrypts. Therefore, $\Pr[\mathcal{R} \to 0 \mid b = 0] = \Pr[\mathsf{UF}_{\Pi',\mathsf{option}'}(\mathcal{A}') \to 1]$.

For the case when the challenge bit $b = 1$, we consider an adversary $\mathcal{A}$ in Algorithm 6 in the $\mathsf{UF}_{\Pi,\mathsf{option}}$ game. $\mathcal{A}$ runs $\mathcal{A}'$ and simulates $\mathsf{O}_{\mathsf{pm}}$ by first sampling $\mathbf{b}'_1 \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}_1}}(1^\lambda)$, running $\mathsf{pm}_1 \leftarrow \mathsf{Probe}(\mathsf{psk}, \mathbf{b}'_1)$, and returning $\mathsf{pm} \leftarrow \mathsf{PKE.Enc}(1\|\mathsf{pm}_1)$. Now, if the challenge bit $b = 1$, then $\mathcal{R}$ perfectly simulates $\mathcal{A}$ in the $\mathsf{UF}_{\Pi,\mathsf{option}}$ game. Therefore, we have $\Pr[\mathcal{R} \to 0 \mid b = 1] = \Pr[\mathsf{UF}_{\Pi,\mathsf{option}}(\mathcal{A}) \to 1]$.

In summary,

$$\begin{aligned}\Pr[\mathsf{MC\text{-}IND\text{-}1CCA}(\mathcal{R}) \to 1] &= \frac{1}{2} \cdot (\Pr[\mathcal{R} \to 0 \mid b = 0] + \Pr[\mathcal{R} \to 1 \mid b = 1]) \\ &= \frac{1}{2} \cdot [\Pr[\mathsf{UF}_{\Pi',\mathsf{option}'}(\mathcal{A}') \to 1] + (1 - \Pr[\mathsf{UF}_{\Pi,\mathsf{option}}(\mathcal{A}) \to 1])] \\ &= \frac{1}{2} + \frac{1}{2} \left(\Pr[\mathsf{UF}_{\Pi',\mathsf{option}'}(\mathcal{A}') \to 1] - \Pr[\mathsf{UF}_{\Pi,\mathsf{option}}(\mathcal{A}) \to 1]\right).\end{aligned}$$

Since both $\mathbf{Adv}^{\mathsf{MC\text{-}IND\text{-}1CCA}}_{\mathsf{PKE},\mathcal{R}} = \left|\Pr[\mathsf{MC\text{-}IND\text{-}1CCA}(\mathcal{R}) \to 1] - \frac{1}{2}\right|$ and $\mathbf{Adv}^{\mathsf{UF}}_{\Pi,\mathcal{A},\mathsf{option}} = \Pr[\mathsf{UF}_{\Pi,\mathsf{option}}(\mathcal{A}) \to 1]$ are negligilbe,

$$\Pr[\mathsf{UF}_{\Pi',\mathsf{option}'}(\mathcal{A}') \to 1] \leq 2 \cdot \mathbf{Adv}^{\mathsf{MC\text{-}IND\text{-}1CCA}}_{\mathsf{PKE},\mathcal{R}} + \mathbf{Adv}^{\mathsf{UF}}_{\Pi,\mathcal{A},\mathsf{option}} = \mathsf{negl}(\lambda).$$

□

We make a conclusion for this subsection. By Theorem 5, under Assumption 1, we can achieve a $\{\mathsf{esk}, \mathsf{psk}, \mathsf{csk}\}$-UF secure scheme $\Pi_{HC}$. If the fh-IPFE scheme is fh-IND secure and Assumption 2 holds, it is also $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}^{(1)}_{\mathsf{pm}}\}$-IND

**Algorithm 5** $\mathcal{R}^{\mathsf{O}_{\mathsf{MC\text{-}Enc}},\mathsf{O}_{\mathsf{Dec}}}(\mathsf{pk}_{\mathsf{PKE}})$

$\mathcal{B}_0 \leftarrow\!\!\$\ \mathbb{B}, \quad \mathbb{B} \leftarrow \mathbb{B} \setminus \mathcal{B}_0$
$\mathcal{B}_1 \leftarrow\!\!\$\ \mathbb{B}$
$\mathbf{b}_0 \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}_0}}(1^\lambda)$
$\mathbf{b}_1 \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}_1}}(1^\lambda)$
$(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$
$\mathsf{esk}' \leftarrow (\mathsf{esk}, \mathsf{pk}_{\mathsf{PKE}})$
$\mathsf{psk}' \leftarrow (\mathsf{psk}, \mathsf{pk}_{\mathsf{PKE}})$
$\mathsf{em}_0 \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b}_0)$
$\mathsf{em}_1 \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b}_1)$
$\mathsf{em}' \leftarrow \mathsf{O}_{\mathsf{MC\text{-}Enc}}(0\|\mathsf{em}_0, 0\|\mathsf{em}_1)$
$\mathsf{ct} \leftarrow \mathcal{A}'^{\mathsf{O}_{\mathsf{pm}}}(\mathsf{esk}', \mathsf{psk}', \mathsf{em}')$
**if** ct is an answer to $\mathsf{O}_{\mathsf{pm}}$ **then**
  **return** $\perp$
**end if**
$\tilde{b}_z \| \tilde{\mathbf{z}} \leftarrow \mathsf{O}_{\mathsf{Dec}}(\mathsf{ct})$
**if** $\tilde{b}_z = 1$ and $\mathsf{Comp}(\mathsf{csk}, \mathsf{em}_0, \tilde{\mathbf{z}}) \to 1$ **then**
  **return** 0
**else**
  **return** 1
**end if**

**Algorithm 6** $\mathcal{A}(\mathsf{esk}, \mathsf{psk})$

$\mathcal{B}_1 \leftarrow\!\!\$\ \mathbb{B}$
$\mathbf{b}_1 \leftarrow \mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}_1}}(1^\lambda)$
$(\mathsf{sk}_{\mathsf{PKE}}, \mathsf{pk}_{\mathsf{PKE}}) \leftarrow \mathsf{PKE.KGen}(1^\lambda)$
$\mathsf{esk}' \leftarrow (\mathsf{esk}, \mathsf{pk}_{\mathsf{PKE}})$
$\mathsf{psk}' \leftarrow (\mathsf{psk}, \mathsf{pk}_{\mathsf{PKE}})$
$\mathsf{em}_1 \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b}_1)$
$\mathsf{em}' \leftarrow \mathsf{PKE.Enc}(\mathsf{pk}_{\mathsf{PKE}}, 0\|\mathsf{em}_1)$
$\mathsf{ct} \leftarrow \mathcal{A}'^{\mathsf{O}_{\mathsf{pm}}}(\mathsf{esk}', \mathsf{psk}', \mathsf{em}')$
**if** ct is an answer to $\mathsf{O}_{\mathsf{pm}}$, or $\mathsf{ct} = \mathsf{em}'$ **then**
  **return** $\perp$
**end if**
$\tilde{b}_z \| \tilde{\mathbf{z}} \leftarrow \mathsf{PKE.Dec}(\mathsf{sk}_{\mathsf{PKE}}, \mathsf{ct})$
**if** $\tilde{b}_z = 1$ **then**
  **return** $\tilde{\mathbf{z}}$
**else**
  **return** $\perp$
**end if**

secure scheme $\Pi_{HC}$ by Theorem 6. Since $\{\mathsf{esk}, \mathsf{psk}, \mathsf{csk}\}$-UF implies $\{\mathsf{esk}, \mathsf{psk}\}$-UF security, we further have a scheme $\Pi'_{HC}$, upgraded by an multi-ciphertext IND-1CCA secure PKE, that is $\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$-UF secure by Theorem 7.

In Table 1, we summarize the security results of various schemes in this work. Sig denotes an sEUF-CMA secure signature scheme, while PKE and PKE′ represent multi-ciphertext IND-CPA and IND-1CCA secure public-key encryption schemes, respectively. The fh-IPFE scheme is assumed to be fh-IND secure. We note that for the $\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$-UF security of the scheme obtained by adding Sig, PKE′ to $\Pi_{HC}$, we require that the signature scheme Sig be deterministic, and each message has a unique signature. This prevents an adversary from signing the same message and creating a different pm.

### 4.5 Implementation and Evaluation

To evaluate the performance of our schemes, we implemented $\Pi_H$ and $\Pi_{HC}$ instantiated by the fh-IPFE scheme in [14]. We recall the construction of [14] in Appendix C. For FE.Dec, we implemented the baby-step-giant-step algorithm [22] to solve the discrete logarithm problem in the group $\mathbb{G}_T$. The implementation was done using Rust 1.94 and the arkworks libraries [9] for pairing-based cryptography. We employed the BN254 (Barreto-Naehrig) asymmetric curve [1], which provides approximately 100-bit security level [15]. All experiments were

Table 1: Summary of security results for various instantiations. "+$\mathsf{Sig}$" and "+$\mathsf{PKE}/\mathsf{PKE}'$" mean transformations in Fig. 3 and 6, respectively, and "+$\mathsf{Sig}, \mathsf{PKE}/\mathsf{PKE}'$" refers to Fig. 3 on top of Fig. 6 (encrypt-then-sign). For the $\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$-UF security of $\Pi_{HC}$ with $\mathsf{Sig}, \mathsf{PKE}', \mathsf{Sig}$ is a deterministic signature scheme such that each message has a unique signature.

| **Scheme** | **Assumptions** | **UF Security** | **IND Security** |
|---|---|---|---|
| Trivial (Def. 5) | None | ✗ | ✗ |
| | + $\mathsf{Sig}$ | $\{\mathsf{esk}, \mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathcal{B}}, \mathsf{O}_{\mathsf{Probe}}\}$ (Thm. 1) | ✗ |
| | + $\mathsf{PKE}$ | ✗ | $\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 3) |
| | + $\mathsf{Sig}$, $\mathsf{PKE}$ | $\{\mathsf{esk}, \mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathcal{B}}, \mathsf{O}_{\mathsf{Probe}}\}$ (Thm. 1) | $\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 3) |
| fh-IPFE $\Pi_E, \Pi_H$ (Sec. 4.2) | Assump. 2 | ✗ | $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 4) |
| | Assump. 2 + $\mathsf{Sig}$ | $\{\mathsf{esk}, \mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathcal{B}}, \mathsf{O}_{\mathsf{Probe}}\}$ (Thm. 1) | $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 4) |
| | Assump. 2 + $\mathsf{PKE}$ | ✗ | $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 4)<br>$\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 3) |
| | Assump. 2 + $\mathsf{Sig}$, $\mathsf{PKE}$ | $\{\mathsf{esk}, \mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathcal{B}}, \mathsf{O}_{\mathsf{Probe}}\}$ (Thm. 1) | $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 4)<br>$\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 3) |
| fh-IPFE $\Pi_{HC}$ (Sec. 4.4) | Assump. 1, 2 | $\{\mathsf{esk}, \mathsf{psk}, \mathsf{csk}\}$ (Thm. 5) | $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}^{(1)}_{\mathsf{pm}}\}$ (Thm. 6) |
| | Assump. 1, 2 + $\mathsf{Sig}$, $\mathsf{PKE}$ | $\{\mathsf{esk}, \mathsf{psk}, \mathsf{csk}\}$ (Thm. 5)<br>$\{\mathsf{esk}, \mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathcal{B}}, \mathsf{O}_{\mathsf{Probe}}\}$ (Thm. 1) | $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}^{(1)}_{\mathsf{pm}}\}$ (Thm. 6)<br>$\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 3) |
| | Assump. 1, 2 + $\mathsf{Sig}$, $\mathsf{PKE}'$ | $\{\mathsf{esk}, \mathsf{psk}, \mathsf{csk}\}$ (Thm. 5)<br>$\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 7)<br>$\{\mathsf{esk}, \mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathcal{B}}, \mathsf{O}_{\mathsf{Probe}}\}$ (Thm. 1) | $\{\mathsf{csk}, \mathsf{em}, \mathsf{O}^{(1)}_{\mathsf{pm}}\}$ (Thm. 6)<br>$\{\mathsf{esk}, \mathsf{psk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$ (Thm. 3) |

conducted on a machine equipped with an Apple M1 Pro chip and 32 GB of RAM. Each algorithm was executed 10 times, and we report the average time. The source code of our implementation is available at [7].

We measure the running time of each algorithm and the size of each key and message for different values of $k$, the length of the enrolled and probe vectors. The results for $\Pi_H$ and $\Pi_{HC}$ are given in Table 2 and Table 3, respectively. We omit the size of the description of the bilinear group.

Table 2: Performance of the $\Pi_H$ scheme instantiated by the fh-IPFE scheme in [14] with the BN254 curve.

| $k$ | Setup | Enroll | Probe | Comp | \|esk\|, \|psk\| | \|csk\| | \|em\| | \|pm\| |
|---|---|---|---|---|---|---|---|---|
| 32 | 1.8 ms | 2.8 ms | 0.9 ms | 9.1 ms | 64 KB | - | 2 KB | 1 KB |
| 64 | 12.6 ms | 3.4 ms | 1.2 ms | 15.7 ms | 256 KB | - | 4 KB | 2 KB |
| 128 | 93.0 ms | 5.9 ms | 2.1 ms | 29.1 ms | 1,024 KB | - | 8 KB | 4 KB |
| 256 | 741.3 ms | 11.5 ms | 4.4 ms | 56.2 ms | 4,096 KB | - | 16 KB | 8 KB |
| 512 | 5,791.4 ms | 22.4 ms | 8.9 ms | 115.4 ms | 16,384 KB | - | 32 KB | 16 KB |
| 1024 | 49,249.9 ms | 51.0 ms | 25.7 ms | 217.1 ms | 65,536 KB | - | 64 KB | 32 KB |

Table 3: Performance of the $\Pi_{HC}$ scheme instantiated by the fh-IPFE scheme in [14] with the BN254 curve.

| $k$ | Setup | Enroll | Probe | Comp | \|esk\|, \|psk\| | \|csk\| | \|em\| | \|pm\| |
|---|---|---|---|---|---|---|---|---|
| 32 | 67 ms | 2.1 ms | 0.7 ms | 268 ms | 64 KB | 130 KB | 2 KB | 1 KB |
| 64 | 224 ms | 3.3 ms | 1.3 ms | 983 ms | 256 KB | 516 KB | 4 KB | 2 KB |
| 128 | 881 ms | 5.9 ms | 2.1 ms | 3,780 ms | 1,024 KB | 2,056 KB | 8 KB | 4 KB |
| 256 | 4,057 ms | 11.5 ms | 4.7 ms | 14,849 ms | 4,096 KB | 8,208 KB | 16 KB | 8 KB |
| 512 | 17,322 ms | 22.2 ms | 9.0 ms | 58,942 ms | 16,384 KB | 32,800 KB | 32 KB | 16 KB |
| 1024 | 103,998 ms | 50.5 ms | 26.9 ms | 232,344 ms | 65,536 KB | 131,136 KB | 64 KB | 32 KB |

For $\Pi_H$ and $\Pi_{HC}$, the Setup algorithm is computationally expensive primarily due to the $k \times k$ matrix inversion and running the key generation $k$ times, respectively. However, we note that Setup is only executed once and can be done offline before the enrollment and probe phases. The Comp algorithm in $\Pi_{HC}$ is also computationally intensive, as it involves running the decryption $k$ times to verify the validity of the probe message. The execution times of Enroll and Probe are very different, as well as the sizes of em and pm. This is a direct consequence of the disparate computational costs associated with the pairing groups $\mathbb{G}_1$ and $\mathbb{G}_2$ in BN254. Given that Probe is the most frequently invoked operation in practical deployments, our implementation delegates its core computations to the more efficient group.

## Acknowledgements

We would like to thank the anonymous reviewers and our shepherd for their constructive feedback, which significantly improved the quality of this paper.

## References


1. Barreto, P.S.L.M., Naehrig, M.: Pairing-friendly elliptic curves of prime order. pp. 319–331 (2006). https://doi.org/10.1007/11693383_22
2. Bellare, M., Boldyreva, A., Micali, S.: Public-key encryption in a multi-user setting: security proofs and improvements. In: Proceedings of the 19th International Conference on Theory and Application of Cryptographic Techniques. p. 259–274. EUROCRYPT'00, Springer-Verlag, Berlin, Heidelberg (2000)
3. Bishop, A., Jain, A., Kowalczyk, L.: Function-hiding inner product encryption. In: Proceedings, Part I, of the 21st International Conference on Advances in Cryptology – ASIACRYPT 2015 - Volume 9452. p. 470–491. Springer-Verlag, Berlin, Heidelberg (2015). https://doi.org/10.1007/978-3-662-48797-6_20, https://doi.org/10.1007/978-3-662-48797-6_20
4. Boyen, X.: Reusable cryptographic fuzzy extractors. In: Proceedings of the 11th ACM Conference on Computer and Communications Security. p. 82–91. CCS '04, Association for Computing Machinery, New York, NY, USA (2004). https://doi.org/10.1145/1030083.1030096, https://doi.org/10.1145/1030083.1030096
5. Bringer, J., Chabanne, H., Patey, A.: SHADE: Secure HAmming distance computation from oblivious transfer. In: Adams, A.A., Brenner, M., Smith, M. (eds.) Financial Cryptography and Data Security. pp. 164–176. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)
6. Cachet, C., Ahmad, S., Demarest, L., Hamlin, A., Fuller, B.: Proximity searchable encryption for the iris biometric. In: Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security. p. 1004–1018. ASIA CCS '22, Association for Computing Machinery, New York, NY, USA (2022). https://doi.org/10.1145/3488932.3497754, https://doi.org/10.1145/3488932.3497754
7. Chen, K.Y.: The cryptographic layer of biometric authentication (2026), https://github.com/kengyuchen/The_Cryptographic_Layer_of_Biometric_Authentication
8. Cheon, J.H., Kim, D., Kim, D., Lee, J., Shin, J., Song, Y.: Lattice-based secure biometric authentication for Hamming distance. In: Baek, J., Ruj, S. (eds.) Information Security and Privacy. pp. 653–672. Springer International Publishing, Cham (2021)
9. arkworks contributors: `arkworks` zksnark ecosystem (2022), https://arkworks.rs
10. Datta, P., Dutta, R., Mukhopadhyay, S.: Functional encryption for inner product with full function privacy. In: Cheng, C.M., Chung, K.M., Persiano, G., Yang, B.Y. (eds.) Public-Key Cryptography – PKC 2016. pp. 164–195. Springer Berlin Heidelberg, Berlin, Heidelberg (2016)
11. Daugman, J.: Chapter 25 - how iris recognition works. In: Bovik, A. (ed.) The Essential Guide to Image Processing, pp. 715–739. Academic Press, Boston (2009). https://doi.org/https://doi.org/10.1016/B978-0-12-374457-9.00025-1, https://www.sciencedirect.com/science/article/pii/B9780123744579000251
12. Ernst, J., Mitrokotsa, A.: A framework for UC secure privacy preserving biometric authentication using efficient functional encryption. In: Tibouchi, M., Wang, X. (eds.) Applied Cryptography and Network Security. pp. 167–196. Springer Nature Switzerland, Cham (2023)
13. Jarrous, A., Pinkas, B.: Secure Hamming distance based computation and its applications. In: Abdalla, M., Pointcheval, D., Fouque, P.A., Vergnaud, D. (eds.) Applied Cryptography and Network Security. pp. 107–124. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)

14. Kim, S., Lewi, K., Mandal, A., Montgomery, H., Roy, A., Wu, D.J.: Function-hiding inner product encryption is practical. In: Catalano, D., De Prisco, R. (eds.) Security and Cryptography for Networks. pp. 544–562. Springer International Publishing, Cham (2018)
15. Kim, T., Barbulescu, R.: Extended tower number field sieve: A new complexity for the medium prime case. pp. 543–571 (2016). https://doi.org/10.1007/978-3-662-53018-4_20
16. Lee, J., Kim, D., Kim, D., Song, Y., Shin, J., Cheon, J.H.: Instant privacy-preserving biometric authentication for Hamming distance. Cryptology ePrint Archive, Paper 2018/1214 (2018), https://eprint.iacr.org/2018/1214
17. Li, N., Guo, F., Mu, Y., Susilo, W., Nepal, S.: Fuzzy extractors for biometric identification. In: 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). pp. 667–677 (2017). https://doi.org/10.1109/ICDCS.2017.107
18. Mandal, A., Roy, A.: Relational hash: Probabilistic hash for verifying relations, secure against forgery and more. In: Gennaro, R., Robshaw, M. (eds.) Advances in Cryptology – CRYPTO 2015. pp. 518–537. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)
19. Othman, N., Dorizzi, B., Garcia-Salicetti, S.: Osiris: An open source iris recognition software. Pattern Recognition Letters **82**, 124–131 (2016). https://doi.org/https://doi.org/10.1016/j.patrec.2015.09.002, https://www.sciencedirect.com/science/article/pii/S0167865515002986, an insight on eye biometrics
20. Pradel, G., Mitchell, C.: Privacy-preserving biometric matching using homomorphic encryption (2021), https://arxiv.org/abs/2111.12372
21. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2015)
22. Shanks, D.: Class number, a theory of factorization, and genera. In: Proceedings of Symposia in Pure Mathematics, vol. 20, pp. 415–440. American Mathematical Society (1971)
23. Tomida, J., Abe, M., Okamoto, T.: Efficient functional encryption for inner-product values with full-hiding security. In: Bishop, M., Nascimento, A.C.A. (eds.) Information Security. pp. 408–425. Springer International Publishing, Cham (2016)
24. Yasuda, M., Shimoyama, T., Kogure, J., Yokoyama, K., Koshiba, T.: Packed homomorphic encryption based on ideal lattices and its application to biometrics. In: Cuzzocrea, A., Kittl, C., Simos, D.E., Weippl, E., Xu, L. (eds.) Security Engineering and Intelligence Informatics. pp. 55–74. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)

## A Case Studies

We introduce several use cases to which our model can be applied.

### A.1 Use case in [12].

Our framework can be used to analyze the setting of [12]. In their use case, a user authenticates to a server from a personal device (e.g., laptop or smartphone) with the help of a secure hardware that holds cryptographic keys. During both enrollment and authentication, the device captures a biometric sample and forwards it to the secure hardware, which encodes and encrypts the sample under its

stored keys. The resulting message is then relayed to the server for registration or verification.

In our framework, enrollment proceeds as follows: the user provides oracle $\mathsf{O}_{\mathcal{B}}$, and the device invokes $\mathsf{getEnroll}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda)$ to obtain a biometric template $\mathbf{b}$ for the secure hardware. The secure hardware generates a key triplet $(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and computes $\mathsf{em} \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$. The device then forwards $\mathsf{csk}$, $\mathsf{em}$, and necessary information for registration to the server. During authentication, the device captures a fresh biometric sample $\mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_{\mathcal{B}}}(1^\lambda)$ for the secure hardware, which computes $\mathsf{pm} \leftarrow \mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$. The server then compares $\mathsf{pm}$ with $\mathsf{em}$ via $s \leftarrow \mathsf{Comp}(\mathsf{csk}, \mathsf{em}, \mathsf{pm})$ and decides acceptance via $\mathsf{Vrfy}(s)$. Fig. 12 illustrates the authentication phase.

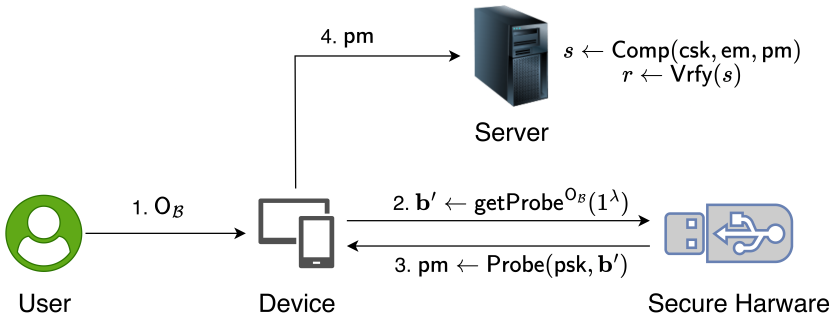



Fig. 12: The authentication phase of the use case in [12].

While [12] analyzes a particular authentication protocol based on fh-IPFE, our abstraction can accommodate and compare multiple instantiations. For instance, in their model, if the secure hardware is stolen and the keys are compromised, an adversary can directly impersonate the user. In contrast, our framework allows a systematic analysis of the consequence of losing $\mathsf{esk}$ or $\mathsf{psk}$ across different instantiations. In addition, their model does not address an adversary's potential knowledge of biometric modalities, which we capture by granting adversaries access to a biometric family $\mathbb{B}$ of distributions. We discuss in detail our security model in Section 3.

### A.2 Identity Document Verification.

In identity document verification, a terminal verifies that the user's live biometrics match the biometric reference stored in the chip of the presented identity document (e.g., passport). During enrollment, the issuing authority captures the user's biometric data and embeds the biometric reference into the chip of the identity document. During authentication, the user presents the identity document to a terminal, which reads the biometric reference from the chip, captures

a fresh biometric sample from the user, and compares the live sample with the stored reference. If the two match, the terminal accepts the authentication and confirms that the presenter is the rightful holder of the document.

Fig. 13 illustrates the enrollment and authentication phases in our framework. During enrollment, the authority generates a key triplet $(\mathsf{esk}, \mathsf{psk}, \mathsf{csk}) \leftarrow \mathsf{Setup}(1^\lambda)$, captures the user's biometrics $\mathbf{b} \leftarrow \mathsf{getEnroll}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$, and embeds $\mathsf{em} \leftarrow \mathsf{Enroll}(\mathsf{esk}, \mathbf{b})$ into the chip of the identity document. The chip may also store additional information to prove its genuineness during the authentication. The authority then provides psk and csk to the terminal for later use.

During authentication, after the terminal verifies the validity of the document, perhaps by protocols based on keys stored in the chip, it reads the biometric reference em from the chip. The terminal then captures a fresh biometric sample $\mathbf{b}' \leftarrow \mathsf{getProbe}^{\mathsf{O}_\mathcal{B}}(1^\lambda)$, computes the probe message $\mathsf{pm} \leftarrow \mathsf{Probe}(\mathsf{psk}, \mathbf{b}')$, and compares em and pm to determine the authentication result.

The protocol can also be modified to store keys psk and/or csk on the chip, either to avoid storing numerous keys in the terminal for multiple users or to improve security. In this scenario, the terminal reads the key(s) from the chip during authentication. However, if the chip is compromised, an adversary then gains more capabilities to impersonate the user or extract biometrics. Our framework can accommodate such variations and analyze the associated risks under different threat models.

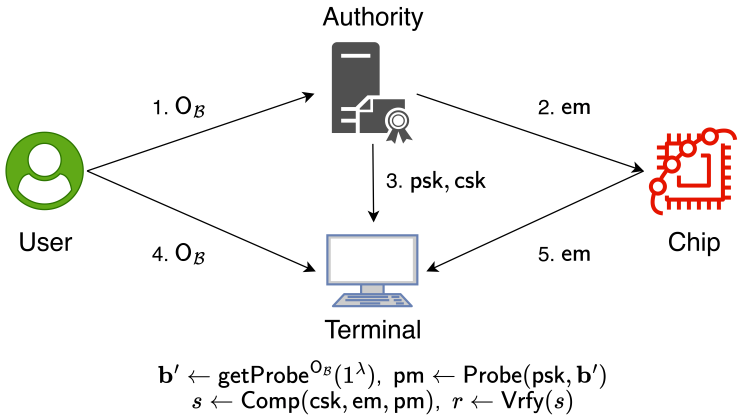



Fig. 13: The enrollment (Steps 1, 2, 3) and authentication (Steps 4, 5) phases of identity document verification.

### A.3 Access Control.

Access control systems use biometric authentication to grant or deny access to physical locations or digital resources. Users first enroll their biometric data with

the authority that manages the system, which securely stores the corresponding enrollment data. When users attempt to access a resource, their biometric sample is captured at a terminal and compared against the stored data to verify their identity, ensuring that only authorized individuals are granted access.

Fig. 14 illustrates the enrollment and authentication phases in our framework, which is similar to those in identity document verification. The key difference is that, in access control, users typically present only their biometrics at the terminal before accessing the resource. Consequently, the enrollment message $\mathsf{em}$ and keys $\mathsf{psk}$ and $\mathsf{csk}$ may be stored at the authority and provided to the terminal on request, or they may be stored directly in the terminal, depending on the deployment.

This configuration can be extended to incorporate a second authentication factor, such as a password, PIN, or physical token, to strengthen security. In such multi-factor settings, the terminal first verifies the second factor and then proceeds with biometric authentication. If a physical token is used, the cryptographic keys $\mathsf{psk}$ and/or $\mathsf{csk}$ may be stored in the token itself, analogous to the identity document verification case.

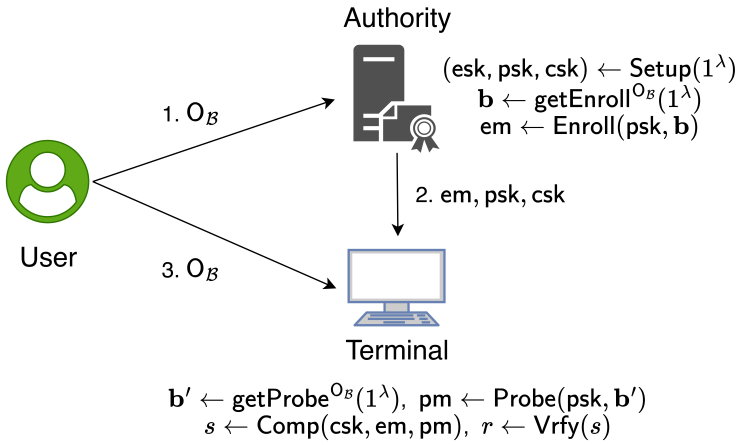
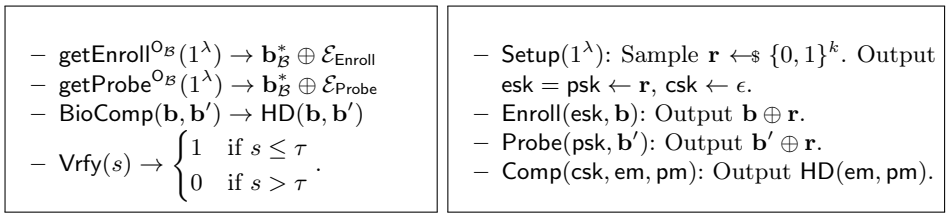



Fig. 14: The enrollment (Steps 1, 2) and authentication (Step 3) phases of an access control system.

## B IND Security for a Particular Biometric Layer

Consider the biometric layer in Fig. 15, where $\mathbf{b}^*_{\mathcal{B}} \in \{0,1\}^k$ is a fixed vector only dependent on $\mathcal{B}$ and $\lambda$, and $\mathcal{E}_{\mathsf{Enroll}}$ and $\mathcal{E}_{\mathsf{Probe}} \subseteq \{0,1\}^k$ are some random *error distributions* independent of $\mathcal{B}$. This biometric layer assumes that a biometric

measurement is composed of user's biometrics, which are fixed for a person, and some random noise that are independent of the user. Two templates are considered to belong to the same person if their Hamming distance is smaller than a threshold $\tau$. Previous works such as [4,18] model biometric template vectors in a similar way.




(a) (b)

Fig. 15: A particular biometric layer (a) and authentication scheme $\Pi$ for it (b).

**Theorem 8.** *Let* $\mathsf{option} = \{\mathsf{csk}, \mathsf{em}, \mathsf{O}_{\mathsf{pm}}\}$. *For the biometric layer in Fig. 15, the authentication scheme $\Pi$ in Fig. 15b is* $\mathsf{option}$*-IND secure.*

*Proof.* Let $\mathbf{b}^*_0$ and $\mathbf{b}^*_1$ be the fixed vectors of $\mathcal{B}^{(0)}$ and $\mathcal{B}^{(1)}$ in the $\mathsf{IND}$ game, respectively. Given any adversary, assume that the number of its queries to $\mathsf{O}_{\mathsf{pm}}$ is bounded by $t$. For any $\mathbf{v}, \mathbf{v}^{(1)}, \cdots, \mathbf{v}^{(t)} \in \{0,1\}^k$,

$$
\begin{aligned}
&\Pr[\mathsf{em} = \mathbf{v},\ \mathsf{pm}^{(1)} = \mathbf{v}^{(1)},\ \cdots,\ \mathsf{pm}^{(t)} = \mathbf{v}^{(t)} \mid b = 0,\ \mathbf{b}^*_0,\ \mathbf{b}^*_1] \\
&= \Pr[\mathbf{b}^*_0 \oplus \mathcal{E}^{(0)}_{\mathsf{Enroll}} \oplus \mathbf{r} = \mathbf{v},\ \mathbf{b}^*_0 \oplus \mathcal{E}^{(1)}_{\mathsf{Probe}} \oplus \mathbf{r} = \mathbf{v}^{(1)},\ \cdots,\ \mathbf{b}^*_0 \oplus \mathcal{E}^{(t)}_{\mathsf{Probe}} \oplus \mathbf{r} = \mathbf{v}^{(t)} \mid \mathbf{b}^*_0,\ \mathbf{b}^*_1] \\
&= \Pr[\mathbf{r} = \mathbf{v} \oplus \mathbf{b}^*_0 \oplus \mathcal{E}^{(0)}_{\mathsf{Enroll}} = \mathbf{v}^{(1)} \oplus \mathbf{b}^*_0 \oplus \mathcal{E}^{(1)}_{\mathsf{Probe}} = \cdots = \mathbf{v}^{(t)} \oplus \mathbf{b}^*_0 \oplus \mathcal{E}^{(t)}_{\mathsf{Probe}} \mid \mathbf{b}^*_0,\ \mathbf{b}^*_1] \\
&= \Pr[\mathbf{r} = \mathbf{v} \oplus \mathbf{b}^*_1 \oplus \mathcal{E}^{(0)}_{\mathsf{Enroll}} = \mathbf{v}^{(1)} \oplus \mathbf{b}^*_1 \oplus \mathcal{E}^{(1)}_{\mathsf{Probe}} = \cdots = \mathbf{v}^{(t)} \oplus \mathbf{b}^*_1 \oplus \mathcal{E}^{(t)}_{\mathsf{Probe}} \mid \mathbf{b}^*_0,\ \mathbf{b}^*_1] \\
&= \Pr[\mathbf{b}^*_1 \oplus \mathcal{E}^{(0)}_{\mathsf{Enroll}} \oplus \mathbf{r} = \mathbf{v},\ \mathbf{b}^*_1 \oplus \mathcal{E}^{(1)}_{\mathsf{Probe}} \oplus \mathbf{r} = \mathbf{v}^{(1)},\ \cdots,\ \mathbf{b}^*_1 \oplus \mathcal{E}^{(t)}_{\mathsf{Probe}} \oplus \mathbf{r} = \mathbf{v}^{(t)} \mid \mathbf{b}^*_0,\ \mathbf{b}^*_1] \\
&= \Pr[\mathsf{em} = \mathbf{v},\ \mathsf{pm}^{(1)} = \mathbf{v}^{(1)},\ \cdots,\ \mathsf{pm}^{(t)} = \mathbf{v}^{(t)} \mid b = 1,\ \mathbf{b}^*_0,\ \mathbf{b}^*_1],
\end{aligned}
$$

where $\mathcal{E}^{(0)}_{\mathsf{Enroll}}$ and $\mathcal{E}^{(i)}_{\mathsf{Probe}}$ for $i = 1, \cdots, t$ are independent samples from $\mathcal{E}_{\mathsf{Enroll}}$ and $\mathcal{E}_{\mathsf{Probe}}$, respectively. Hence, the adversary cannot distinguish between $\mathsf{em}, \mathsf{pm}^{(1)}, \cdots, \mathsf{pm}^{(t)}$ generated from $\mathcal{B}^{(0)}$ and $\mathcal{B}^{(1)}$. □

We remark that the scheme $\Pi$ in Fig. 15b is adding a one-time pad encryption to the biometric templates. Therefore, it is insecure if the adversary has access to a probe of another user.

## C Construction in [14]

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of order a prime number $q$ with generators $g_1$ and $g_2$, respectively. Let $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be a mapping to a target group $\mathbb{G}_T$ also of order $q$.

**Definition 13 (Bilinear asymmetric group [14]).** *A tuple* $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, e)$ *is a* bilinear asymmetric group *if the following hold.*

- *Group operations in* $\mathbb{G}_1, \mathbb{G}_2$*, and* $\mathbb{G}_T$ *and mapping* $e$ *are efficiently computable.*
- $e$ *is bilinear. That is, for* $x, y \in \mathbb{Z}_q$*,* $e(g_1^x, g_2^y) = e(g_1, g_2)^{xy}$*.*
- $e$ *is non-degenerate. That is,* $e(g_1, g_2) \neq 1$*, the identity element of* $\mathbb{G}_T$*.*

For a vector $\mathbf{v} = (v_1, v_2, \cdots, v_n) \in \mathbb{Z}_q^n$ and a group element $g$ in group of order $q$, we write $g^{\mathbf{v}}$ to denote the vector of group elements $(g^{v_1}, g^{v_2}, \cdots, g^{v_n})$. Moreover, for $k \in \mathbb{Z}_q$ and $\mathbf{v}, \mathbf{w} \in \mathbb{Z}_q^n$, we write $(g^{\mathbf{v}})^k = g^{k \cdot \mathbf{v}}$ and $g^{\mathbf{v}} \cdot g^{\mathbf{w}} = g^{\mathbf{v}+\mathbf{w}}$. Finally, the pairing operation is extended to vectors.

$$e(g_1^{\mathbf{v}}, g_2^{\mathbf{w}}) = \prod_{i \in [n]} e(g_1^{v_i}, g_2^{w_i}) = e(g_1, g_2)^{\langle \mathbf{v}, \mathbf{w} \rangle}.$$

We now recall the fh-IPFE construction $\mathsf{FE}$ in [14].

- $\mathsf{FE.Setup}(1^\lambda)$: Sample an asymmetric bilinear group $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, e)$ and choose generators $g_1 \in \mathbb{G}_1$ and $g_2 \in \mathbb{G}_2$. Sample $\mathbf{B} \in \mathbb{GL}_n(\mathbb{Z}_q)$ and find $\mathbf{B}^* = \det(\mathbf{B}) \cdot (\mathbf{B}^{-1})^T$. Finally, output the public parameter $\mathsf{pp} = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, q, e)$ and the master secret key $\mathsf{msk} = (\mathsf{pp}, g_1, g_2, \mathbf{B}, \mathbf{B}^*)$.
- $\mathsf{FE.KGen}(\mathsf{msk}, \mathbf{x})$: Sample $\alpha \leftarrow\!\!\$\, \mathbb{Z}_q$ and output

$$\mathsf{sk}_{\mathbf{x}} = (K_1, K_2) = \left(g_1^{\alpha \cdot \det(\mathbf{B})}, g_1^{\alpha \cdot \mathbf{x} \cdot \mathbf{B}}\right)$$

- $\mathsf{FE.Enc}(\mathsf{msk}, \mathbf{y})$: Sample $\beta \leftarrow\!\!\$\, \mathbb{Z}_q$ and output

$$\mathsf{ct}_{\mathbf{y}} = (C_1, C_2) = \left(g_2^{\beta}, g_2^{\beta \cdot \mathbf{y} \cdot \mathbf{B}^*}\right)$$

- $\mathsf{FE.Dec}(\mathsf{pp}, \mathsf{sk}_{\mathbf{x}}, \mathsf{ct}_{\mathbf{y}}) \to z$: Parse $\mathsf{sk}_{\mathbf{x}} = (K_1, K_2)$ and $\mathsf{ct}_{\mathbf{y}} = (C_1, C_2)$ and compute

$$D_1 = e(K_1, C_1) \quad \text{and} \quad D_2 = e(K_2, C_2)$$

Solve the discrete logarithm to find $z$ such that $D_1^z = D_2$ and output $z$. If it fails to find such $z$, output $\perp$.

**Correctness** We have

$$D_1 = e(K_1, C_1) = e(g_1, g_2)^{\alpha \cdot \beta \cdot \det(\mathbf{B})}$$

and

$$D_2 = e(K_2, C_2) = e(g_1, g_2)^{\alpha \cdot \beta \cdot \mathbf{x} \cdot \mathbf{B} \cdot (\mathbf{B}^*)^T \cdot \mathbf{y}^T \cdot} = e(g_1, g_2)^{\alpha \cdot \beta \cdot \det(\mathbf{B}) \cdot \mathbf{x}\mathbf{y}^T}.$$

Therefore, $(D_1)^{\langle \mathbf{x}, \mathbf{y} \rangle} = D_2$.

*Remark.* In this construction, $q$ is exponential in $\lambda$ to achieve security, and decryption relies on some prior knowledge of possible ranges of the inner product $\langle \mathbf{x}, \mathbf{y} \rangle$. For example, for the instantiation $\Pi_E$ in Section 4.2, one can enumerate $z \in \{0, 1, \cdots, m^2 \cdot k\}$ and return $\perp$ when no valid $z$ in the range such that $D_1^z = D_2$ is found. For schemes $\Pi_H$ and $\Pi_{HC}$, one can enumerate $z \in \{-k, -k+2, \ldots, k-2, k\}$.