Upgrading Fuzzy Extractors

Chloe Cachet* Ariel Hamlin† Maryam Rezapour‡ Benjamin Fuller§

December 21, 2023

Abstract

Fuzzy extractors derive stable keys from noisy sources non-interactively (Dodis et al., SIAM Journal of Computing 2008). Since their introduction, research has focused on two tasks: 1) showing security for as many distributions as possible and 2) providing stronger security guarantees including allowing one to enroll the same value multiple times (reusability), security against an active attacker (robustness), and preventing leakage about the enrolled value (privacy).

Existing constructions of reusable fuzzy extractors are direct and do not support as many distributions as the best non-reusable constructions. Constructions of robust fuzzy extractors require strong assumptions even in the CRS model.

Given the need for progress on the basic fuzzy extractor primitive, it is prudent to seek generic mechanisms to transform a fuzzy extractor into one that is robust, private, and reusable so that it can inherit further improvements.

This work asks if one can generically upgrade fuzzy extractors to achieve robustness, privacy, and reusability. We show positive and negative results: we show upgrades for robustness and privacy, but we provide a negative result on reuse.

1. We upgrade (private) fuzzy extractors to be robust under weaker assumptions than previously known in the common reference string model.
2. We show a generic upgrade for a private fuzzy extractor using multi-bit compute and compare (MBCC) obfuscation (Wichs and Zirdelis, FOCS 2017) that requires less entropy than prior work.
3. We show one cannot arbitrarily compose private fuzzy extractors. It is known one cannot reuse an arbitrary fuzzy extractor; each enrollment can leak a constant fraction of the input entropy.

   We show that one cannot build a reusable private fuzzy extractor by considering other enrollments as auxiliary input. In particular, we show that assuming MBCC obfuscation and collision-resistant hash functions, there does not exist a private fuzzy extractor secure against unpredictable auxiliary inputs strengthening a negative result of Brzuska et al. (Crypto 2014).

Keywords: Fuzzy extractors, obfuscation, biometrics, key derivation.

1 Introduction

Fuzzy extractors [BBR88, DRS04, DORS08, ŠTO05, HAD06, DKRS06, FMR13, CFP+16, FRS16, ACEK17, ABC+18, FP19, WLH18, WL18, DFR21, ACF+22, FRS20, Ful23] derive stable keys from noisy sources. They are used on devices to derive keys from biometrics, physical unclonable functions and quantum information. They also are used in interactive protocols such as distributed key agreement and password-authenticated key exchange. [BBCS91, DKRS06, BG11, EHKM11, DKK+12, BCP13, BDCG13, DCH+16, DHP+18]. A fuzzy extractor is a pair of algorithms called generate (Gen) and reproduce (Rep) with two properties:

Correctness For all $w, w' \in \mathcal{M}$ such that $\mathsf{dist}(w, w') \leq t$, let $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{Gen}(w)$ where pub is a public *helper* value used to provide correctness. Then it should be the case that $\mathsf{key} \leftarrow \mathsf{Rep}(w', \mathsf{pub})$.

*National Research Council of Canada, chloe.cachet@nrc-cnrc.gc.ca. Some work done while at the University of Connecticut.

†Khoury College of Computer Sciences, Northeastern University a.hamlin@northeastern.edu.

‡University of Connecticut, maryam.rezapour@uconn.edu.

§University of Connecticut, benjamin.fuller@uconn.edu.

Scheme	Model	Distribution	Reuse	Robust	Private
[Boy04]	RO*	High entropy	Shift	✓	✗
[DS05]	Plain	High entropy	✗	✗	✓
[CFP+16]		Avg. Subsets Entropy	Correlation	✗	✗
[ACEK17]		Independent	Shift	✗	✗
[BCKP17]	Plain	All	✗	✗	✓
[WLH18]	CRS	High entropy	Shift	✗	✗
[WL18]	CRS	High entropy	Shift	✓	✗
[WLG19]	CRS	High entropy	Shift	✓	✗
[GZ19]		All	✗	✗	✓
[DFR21]		MIPURS	Correlation	✗	✗

Table 1: Previous constructions of fuzzy extractors that are reusable, robust, or private. See Demarest, Fuller, and Russell [DFR21] for descriptions of distributional properties. "High entropy" is used when the construction relies on an information-theoretic error correction component. Such constructions usually require the input source W to have entropy of at least $h_2(t/n)$, see [CFP+16, Proposition 1]. For Boyen's [Boy04] work, the RO model is only required for insider security, when keys are seen from other enrollments.

Security Let W be a probability distribution of noisy values. For $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{Gen}(W)$ it should be the case that key is indistinguishable from uniform even knowing pub.

Security is defined relative to the statistics of the probability distribution. The most common and useful of which is fuzzy min-entropy [FRS16, WCD+17, FRS20] which measures the adversary's success when provided with the functionality of reproduce. For security to be possible, it must be the case that a negligible fraction of the weight of W lies within any ball of radius t. Fuzzy min-entropy measures the adversary's success when providing the fixed "best" point w^* to the reproduce functionality. Even after 25 years of research, the design of fuzzy extractors for distributions with fuzzy min-entropy is an unsettled problem with advancements yet to be made.

There are constructions for distributions with high entropy, where bits are independent, or display additional statistical properties (see [DFR21] for an overview of considered properties). There are two known methods to build a fuzzy extractor for all such distributions, using virtual grey box obfuscation for NC1 evasive circuits [BCKP17][1] or with a new subset-product assumption [GZ19]. Both of these assumptions require additional study before deployment.

Fuzzy extractor security is also insufficient for many applications. There are three primary augmentations to the definition that exist in the literature:

Reusability [Boy04] One can enroll the noisy source multiple times with different devices. Crucially, the multiple enrollments are subject to noise. In prior work [Boy04, CFP+16], this noise is controlled by an adversary.

Robustness [BDK+05] If an attacker modifies pub to a related value pub', this behavior is detectable. That is, $\mathsf{Rep}(w', \mathsf{pub}')$ should only output the original key or $\perp$.

Privacy [DS05] Privacy ensures no information is leaked about the enrolled value. More specifically, it ensures that no predicate of the enrollment value can be guessed better after seeing pub.

Table 1 summarizes prior constructions of fuzzy extractors with at least one of these additional properties. No previous construction that is reusable or robust supports all distributions with fuzzy min-entropy. The prior gray-box obfuscation [BCKP17] and subset product constructions [GZ19] are obfuscations of the fuzzy extractor functionality; they are by definition private. Given the unsettled nature of constructing fuzzy extractors for distributions with fuzzy min-entropy, it is prudent to seek generic mechanisms to transform a fuzzy extractor into a reusable, robust, and private one.

[1] Virtual gray box obfuscation of all evasive programs implies virtual gray box obfuscation for all programs [BBC+14]. Virtual gray box and virtual black box obfuscation are equivalent in the distributional setting for evasive circuit families [BCKP17].

1.1 Our Contribution

We present three contributions, 1) an upgrade for privacy 2) an upgrade for robustness that preserves privacy, and 3) a negative result for reusability.

Privacy We show how to construct a private fuzzy extractor from either a secure sketch or a non-private fuzzy extractor. Our contribution is a strengthening of the previous upgrade from a secure sketch to a private secure sketch using multi-bit compute and compare obfuscation (MBCC) [WZ17]. We support a wider family of distributions with lower entropy than prior work [WZ17]. We first introduce Wichs and Zirdelis' [WZ17] construction and then the advantages of our construction.

Both our work and prior work is based on the the notion of a secure sketch. A secure sketch recovers the original value w rather than deriving a random key. It is a pair of algorithms $(\mathsf{Sketch}, \mathsf{Rec})$ such that

Correctness For all $w, w' \in \mathcal{M}$ such that $\mathsf{dist}(w, w') \leq t$, then $\mathsf{Rec}(w', \mathsf{Sketch}(w)) = w$.

Security Let W be a probability distribution of noisy values. Given $\mathsf{Sketch}(W)$, W has high min-entropy. (One can also use computational notions of security using pseudoentropy or unpredictability [FMR20].)

Wichs and Zirdelis use MBCC program obfuscation at the core of their scheme. MBCC program has three values f, y, z. On input x it computes $f(x)$, if $f(x) = y$ then it outputs z otherwise it outputs $\perp$. In their prior construction they showed how to obfuscate a family of such programs where y has pseudoentropy [HILL99] conditioned on f and z. They show how to upgrade a secure sketch into a private one by obfuscating $h(\mathsf{Rec}(\cdot, ss))$ where h is a pairwise independent hash function. They have to choose the output length of the hash function based on the entropy of the input, which keeps the construction from working for all distributions with sufficient entropy for the MBCC obfuscation to be secure.

In our work, we use a secure sketch to construct a private fuzzy extractor to directly analyze the construction without h, making the same construction work for any distribution where the secure sketch retains (a super-logarithmic amount of) min-entropy. By removing the hash function, we also reduce the amount of entropy required as one doesn't "leak" the hash value in the security analysis. We also show a similar upgrade from fuzzy extractors to private fuzzy extractors. There are stronger negative results on constructing secure sketches [DORS08, FRS16, Ful23], so direct constructions from fuzzy extractors may yield better parameters.

The privacy definition of Wichs and Zirdelis [WZ17] considers predicting predicates of the source w in contrast to Dodis and Smith [DS05] who consider functions. We call this weak-privacy to distinguish from Dodis and Smith's definition. Restriction to predicates is standard in the obfuscation literature as an obfuscation itself is a function that a simulator cannot hope to reproduce.

Robustness We provide a simpler construction of robust fuzzy extractors than prior work. Our result only requires the existence of true simulation extractable NIZKs [DHLAW10]. Prior work of Feng and Tang [FT21] also required the existence of extremely lossy functions or ELFs [Zha19].[2] Additionally, our result shows that this transform preserves privacy, which was not considered by any prior robustness upgrade.

Reuse One cannot expect reuse of arbitrary fuzzy extractors. Each value of pub can leak a constant fraction of the entropy in the source w while remaining secure. However, a (weakly) private fuzzy extractor cannot "leak" on input w. There are multiple private fuzzy extractors (see Table 1) that are not known to be reusable. The most natural approach for a reusable private fuzzy extractor is to construct a fuzzy extractor for all sources W that are unpredictable in the presence of auxiliary input available to the adversary. The security analysis would follow by including other enrollments of the same source in the auxiliary input.

We show this proof technique is not possible. Namely, we show that the existence of MBCC obfuscation and collision-resistant hash functions imply that one cannot construct private fuzzy extractors for all W that are unpredictable conditioned on auxiliary input. We do this by showing that auxiliary-input secure digital lockers cannot be secure in the presence of MBCC obfuscation. We then show that a variant of private fuzzy extractors imply digital lockers. Brzuska et al. [BFM14] proved an analogous result where auxiliary-input

[2] Feng and Tang's primary goal was to construct robust extractors, not robust fuzzy extractors. Unlike Feng and Tang we work in the standard CRS model, they allow the source W to depend on the CRS.

Upgrade	Scheme	Model	Any FE	Tools	Err.
Reusability	[ABC$^+$18]	Plain	✓	composable DL [BC10]	t
Robustness	[BDK$^+$05]	RO	✗	RO	t
	[CDF$^+$08]	CRS	✗	IT	2t
	[DKK$^+$12]	Plain	✗	IT	t
	[FT21]	CRS*	✓	ELFS [Zha19] + true sim. extract NIZK [DHLAW10]	2t
	[ACF$^+$22]	Plain	✗	comp.* DL [BLMZ19, Assumption 3]+ true sim. extract NIZK [DHLAW10]	2t
	This work	CRS	✓	true sim. extract NIZK [DHLAW10]	2t
Private	[WZ17]	Plain	✓	LWE + ELFS [Zha19]	t
	This work	Plain	✓	LWE + ELFS [Zha19]	t

Table 2: Previous upgrades of fuzzy extractors. If there is an ✗ in the Any FE. column the construction requires the use of the syndrome secure sketch. As mentioned in Table 1 this places a lower bound on entropy of the distribution W. CRS* is the CRS model where the distribution W being enrolled can depend on the CRS. Err. column describes how many errors the underlying primitive is required to correct. Multiple robust constructions require a secure sketch or fuzzy extractor that corrects $2t$ errors to be able to extract a value from the adversary.

secure digital lockers were incompatible with indistinguishability obfuscation [GGH$^+$13b, GGH13a].[3] Since MBCC obfuscation implies auxiliary-input secure digital lockers, this shows MBCC obfuscation cannot be safely composed either.

1.2 Related Work

Reusability Alamelou et al. [ABC$^+$18] show how to create reuse for the Hamming and set difference metrics when the source has symbols that are super polynomial size. However, most natural sources consider small, often binary, alphabets. Alamelou et al.'s technique cannot work in this setting.[4] We note this technique is applied before the source is input to the fuzzy extractor.

Robustness In the random oracle model, for a fuzzy extractor with output $\mathsf{key}, \mathsf{pub}$ and random oracle h one can split $\mathsf{key} = (\mathsf{key}_1, \mathsf{key}_2)$ and include $h(\mathsf{key}_2||\mathsf{pub})$ as part of pub. As needed the random oracle can expand the amount of available keying material.[5] Without resorting to random oracles, one can use algebraic manipulation detection codes [CDF$^+$08] and pairwise independent hashes as one-time MACs. In the CRS model, Feng and Tang [FT21] codify the security required from the MAC, and show how to generically lift a secure sketch into a robust fuzzy extractor using a primitive they call a κ-MAC that is secure for low-entropy keys that can be manipulated by an adversary. This upgrade is agnostic in the underlying secure sketch. Apon et al. [ACF$^+$22] propose a standard model upgrade that requires the syndrome secure sketch.

Privacy For privacy, if one has a secure sketch that retains superlogarithmic entropy, one can upgrade it to a private secure sketch using multi-bit compute and compare (MBCC) obfuscation [WZ17]. Roughly, MBCC allows one to compute a function on some input and compare the result with a target value. If the output of the function matches the target, the MBCC circuit returns a fixed value.

1.3 Discussion and Future Work

The privacy upgrade is this work is not yet of practical efficiency. MBCC obfuscation has nearly as much overhead as indistinguishability obfuscation. A natural question is whether an upgrade that preserves privacy

[3] Their actual result showed the impossibility of auxiliary input universal computational extractors. This object implies auxiliary-input secure digital lockers

[4] Alamelou et al. use a *pseudoentropic isometry* that maps points to a new metric space while 1) preserving distance and 2) the value in the new metric space doesn't reveal the value on the original metric space. For the Hamming metric, the only such transforms are equivalent to a per-symbol permutation and a permutation of symbol order. Such a transform can only be one-way if symbols are super-polynomial size. No pseudoentropic isometric exists for the Hamming metric with polynomial size symbols.

[5] Boyen [BDK$^+$05] considers a secure sketch, the same idea works for a fuzzy extractor.

must use a type of obfuscation and if so, can one use a obfuscation of a weaker class of obfuscation?

Our negative result for reuse does leave open the possibility of upgrading fuzzy extractors to be reusable. It does not rule out techniques that transform w to a new metric space [ABC+18]. Furthermore, one may able to use a more fine grained argument for reuse. As a reminder, our negative result only rules out private fuzzy extractors secure in the presence of unpredictable auxiliary input. One may be able to sidestep the result by only showing security when the auxiliary input is a fuzzy extractor enrollment. We tried to extend our negative result to this setting but were not successful.

Organization The rest of this work is organized as follows: Section 2 covers mathematical preliminaries, Section 3 shows our privacy upgrade, Section 4 covers robustness, and Section 5 covers reuse. Appendix A shows that weak-privacy does not imply fuzzy extractor security and Appendix B shows that a composable MBCC obfuscation would yield a reusable upgrade (but is ruled out by our negative result).

2 Preliminaries

Let λ be the security parameter throughout this paper. A function $\mathsf{ngl}(\lambda)$ is negligible in λ if for all $a \in \mathbb{Z}^+$ we have $\mathsf{ngl}(\lambda) = o(\frac{1}{\lambda^a})$. A function $\mathsf{poly}(\lambda)$ is polynomial in λ if there exists some constant $a \in \mathbb{Z}^+$ such that $\mathsf{poly}(\lambda) = O(\lambda^a)$. We use $\mathsf{poly}(\lambda)$ and $\mathsf{ngl}(\lambda)$ to denote unspecified functions that are polynomial and negligible in λ, respectively. The notation id is used to denote the identity function: $\forall x, \mathsf{id}(x) = x$. For some $n \in \mathbb{N}$, $[n]$ denotes the set $\{1, \cdots, n\}$. Let $x \xleftarrow{\$} S$ denote sampling x uniformly at random from the finite set S. We say that distributions X and Y are computationally indistinguishable if for all PPT (in λ) adversaries $\mathcal{A}$, $|\Pr[\mathcal{A}(X) = 1] - \Pr[\mathcal{A}(Y) = 1]| \leq \mathsf{ngl}(\lambda)$.

2.1 Entropy definitions

Definition 1 (Min-entropy)**.** *For a discrete random variable X, the* min-entropy *of X is*

$$\mathrm{H}_\infty(X) = -\log\left(\max_x \Pr[X = x]\right)$$

Definition 2 (Average conditional min-entropy [DORS08])**.** *For a pair of discrete random variables X, Y, the average min-entropy of $X|Y$ is*

$$\tilde{\mathrm{H}}_\infty(X \mid Y) = -\log\left(\mathit{Exp}_{y \in Y}\left(2^{-\mathrm{H}_\infty(X|Y)}\right)\right).$$

Definition 3 (Conditional HILL entropy [HILL99, HLR07])**.** *Let X, Y be ensembles of jointly distributed random variables. The conditional pseudo-entropy of X conditioned on Y, denoted as $H_{\mathsf{HILL}}(X \mid Y)$, is greater or equal to $\ell(\lambda)$ if there exists some ensemble X' such that (X, Y) and (X', Y) are computationally indistinguishable and $\mathrm{H}_\infty(X' \mid Y) \geq \ell(\lambda)$.*

2.2 Obfuscation definitions

Definition 4 (Distributional Virtual Black Box (dist-VBB) obfuscation)**.** *Let $\mathcal{P}$ be a family of programs and Obf be a PPT algorithm that takes as input a program $P \in \mathcal{P}$, a security parameter $\lambda \in \mathbb{N}$ and outputs a program $\tilde{P} \leftarrow \mathsf{Obf}(1^\lambda, P)$. Let $\mathcal{D}$ be a class of distribution ensembles $D = \{D_\lambda\}_{\lambda \in \mathbb{N}}$ which samples $(P, \mathsf{aux}) \leftarrow D_\lambda$ with $P \in \mathcal{P}$. Then Obf is an obfuscator for the distribution class $\mathcal{D}$ over the program family $\mathcal{P}$ if it satisfies the following:*

- ***Functionality preserving:*** *For all $P \in \mathcal{P}$ and for all inputs $x \in \{0,1\}^n$, we have*

$$\Pr[P(x) = \tilde{P}(x)] \geq 1 - \mathsf{ngl}(\lambda)$$

- ***Polynomial slowdown:*** *For all sufficiently large $\lambda \in \mathbb{N}$ and for all $P \in \mathcal{P}_\lambda$,*

$$|\tilde{P}| \leq \mathsf{poly}(|P|)$$

- ***Distributional Virtual Black-Box:*** *For every PPT adversary $\mathcal{A}$ there exists a non-uniform polynomial size simulator* Sim*, such that for every distribution ensemble $D = \{D_\lambda\} \in \mathcal{D}$, and every predicate $\phi : \mathcal{P} \to \{0,1\}$, we have*

$$\Big| \Pr_{(P,\mathsf{aux})\leftarrow D_\lambda}[\mathcal{A}(\mathsf{Obf}(1^\lambda, P), \mathsf{aux}) = \phi(P)] - \Pr_{(P,\mathsf{aux})\leftarrow D_\lambda}[\mathsf{Sim}^P(1^\lambda, 1^{|P|}, \mathsf{aux}) = \phi(P)]\Big| \leq \mathsf{ngl}(\lambda)$$

where Sim^P has black-box access to the program P.

Wichs and Zirdelis [WZ17] build a dist-VBB obfuscator for α-pseudo entropy distributions (see Definition 6) over multi-bit-compute-and-compare circuits.[6]

Definition 5 (Multi-bit compute-and-compare circuit)**.** *Let $n, \ell, \kappa \in \mathbb{N}$ and consider a function $f : \{0,1\}^n \to \{0,1\}^\ell$, a target value $y \in \{0,1\}^\ell$ and some value $z \in \{0,1\}^\kappa$. A multi-bit compute-and-compare circuit is defined for all inputs $x \in \{0,1\}^n$ as*

$$\mathsf{MBCC}_{f,y,z}(x) = \begin{cases} z & \text{if } f(x) = y \\ \perp & \text{otherwise.} \end{cases}$$

Wichs and Zirdelis [WZ17] also define α-pseudo entropy, a specific case of HILL entropy:

Definition 6 (α-pseudo entropy)**.** *For function $\alpha(\lambda)$, the class of α-pseudo-entropy distributions consists of ensembles $D = \{D_\lambda\}$ such that $(\mathsf{MBCC}[f, y, z], \mathsf{aux}) \leftarrow D_\lambda$ satisfies $H_{\mathsf{HILL}}(y \mid f, z, \mathsf{aux}) \geq \alpha(\lambda)$.*

2.3 Fuzzy extractors

Fuzzy extractors allow to generate stable cryptographic keys from noisy sources. We focus on computational fuzzy extractors.

Definition 7 (Computational Fuzzy Extractor [FMR13, FMR20])**.** *An $(\mathcal{M}, \mathcal{W}, \ell, t, \epsilon)$-fuzzy extractor with error δ is a pair of PPT algorithms* $(\mathsf{Gen}, \mathsf{Rep})$ *where for all $w, w' \in \mathcal{M}$,*

- $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{Gen}(w)$*, where* $\mathsf{key} \in \{0,1\}^\ell$ *and* $\mathsf{pub} \in \{0,1\}^*$
- $\mathsf{key}' \leftarrow \mathsf{Rep}(\mathsf{pub}, w')$

the following properties are true:

1. ***Correctness :*** *For all $w, w' \in \mathcal{M}$ such that $\mathsf{dist}(w, w') \leq t$,*

$$\Pr\left[\mathsf{key}' = \mathsf{key} \;\middle|\; (\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{Gen}(w), \mathsf{key}' \leftarrow \mathsf{Rep}(\mathsf{pub}, w')\right] \geq 1 - \delta$$

2. ***Security :*** *For any PPT distinguisher $\mathcal{A}$ and distribution $W \in \mathcal{W}$,*

$$|\Pr[\mathcal{A}(\mathsf{key}, \mathsf{pub}) = 1] - \Pr[\mathcal{A}(U_\ell, \mathsf{pub}) = 1]| \leq \epsilon$$

where $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{Gen}(W)$ and U_ℓ is a uniformly distributed random variable over $\{0,1\}^\ell$.

[6] In an independent and concurrent work, Goyal et al. [GKW17] proposed a similar object they called *lockable obfuscation*.

3 Weakly-private fuzzy extractors

Fuzzy extractor security does not prevent leaking information about the value w, called a *template*. For example, consider a fuzzy extractor where the public value leaks a random bit of the template. This can be problematic, especially if the biometric source is used in different contexts. Preventing such leakage, although not mandatory to achieve fuzzy extractor security, is thus desirable. Constructions that prevent such leakage are said to be *private* [DS05]. We adapt Wichs and Zirdelis [WZ17] privacy definition for secure sketches to fuzzy extractors, we call this weak privacy. This definition differs from Dodis and Smith's [DS05] in that the adversary is restricted to predicting predicates about the value W (in place of general functions). We start by introducing the definition of a weakly private fuzzy extractor.

Definition 8 (Weakly Private Fuzzy Extractor)**.** *Let* $\mathsf{FE} = (\mathsf{Gen}, \mathsf{Rep})$ *satisfy the correctness condition of Definition 7 for parameters* t *and* δ*. We say that* FE *is* η*-weakly-private if for all adversary* $\mathcal{A}$*, there exists a simulator* Sim *such that for every source* W *over* $\mathcal{W}$ *and every predicate* $\phi : \{0,1\}^* \to \{0,1\}$*, we have*

$$\Big| \Pr[\mathcal{A}(\mathsf{pub}, \mathsf{key}) = \phi(W) \mid (\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{FE.Gen}(W)] - \Pr[\mathsf{Sim}(1^\lambda, 1^{|\mathsf{pub}|}, 1^{|\mathsf{key}|}) = \phi(W)] \Big| \leq \eta$$

Fuzzy extractors were originally built following a *sketch-then-extract* approach. First, a secure sketch [DRS04] is used to recover the enrolled w from a close value w', then a randomness extractor is used to derive the secret key. We add the definition for secure sketches:

Definition 9 (Secure sketch)**.** *Let* λ *be a security parameter. Let* $\mathcal{W} = \mathcal{W}_\lambda$ *be a family of random variables over the metric space* $(\mathcal{M}, \mathsf{dist}) = (\mathcal{M}_\lambda, \mathsf{dist}_\lambda)$*. Then* $(\mathsf{Sketch}, \mathsf{Rec})$ *is a* $(\mathcal{M}, \mathcal{W}, \ell, t, \delta)$*-secure sketch if the following hold:*

- ***Correctness:*** *For all* $w, w' \in \mathcal{M}$ *such that* $\mathsf{dist}(w, w') \leq t$*,*

$$\Pr[\mathsf{Rec}(w', \mathsf{Sketch}(w)) = w] \geq 1 - \delta.$$

- ***Security:*** *For all distributions* $W \in \mathcal{W}$ *it is true that*

$$H_\infty(W \mid \mathsf{Sketch}(W)) \geq \ell.$$

We propose two weakly private fuzzy extractors constructions using dist-VBB obfuscation for multi-bit-compute-and-compare (MBCC) circuits. The first construction builds on a non-private fuzzy extractor whereas the second builds on a non-private secure sketch.

Intuitively, we can build weakly private fuzzy extractors as follows: we first build an MBCC circuit for function $f_{w,y,t}$ that outputs target value y on input w' only when $\mathsf{dist}(w, w') \leq t$ and we set the output value z to be sampled uniformly at random. We then set pub to be the obfuscated MBCC program and key to be z. Note that in this case, since z is sampled independently from all other values, the entropy requirement for the MBCC circuit to be obfuscable can be simplified to

$$H_{\mathsf{HILL}}(y \mid f, \mathsf{aux}) \geq \alpha.$$

3.1 Weakly private FE from FE and MBCC obfuscation

Our first construction builds weakly-private fuzzy extractors from non-private fuzzy extractors and MBCC obfuscation.

Construction 1 (Weakly Private FE from MBCC obfuscation and FE)**.** *Let* FE *be an* $(\mathcal{M}, \mathcal{W}, \ell, t, s, \epsilon)$*-fuzzy extractor and* Obf *be an obfuscator for* ℓ*-pseudo-entropy distributions over multi-bit compute-and-compare circuits. We can build an* $(\mathcal{M}, \mathcal{W}, \kappa, t, s, \epsilon)$*-fuzzy extractor* PFE *as follows:*

- $(\mathsf{key}', \mathsf{pub}') \leftarrow \mathsf{PFE.Gen}(w)$*:*

1. *Compute* $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{FE.Gen}(w)$.
2. *Sample* $\mathsf{key}' \xleftarrow{\$} \{0,1\}^\kappa$.
3. *Define the circuit* $f_{\mathsf{pub}}(\cdot) := \mathsf{FE.Rep}(\cdot, \mathsf{pub})$.
4. *Compute* $\mathsf{pub}' \leftarrow \mathsf{Obf}\big(1^\lambda, \mathsf{MBCC}_{f_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}'}\big)$.
5. *Output* $(\mathsf{key}', \mathsf{pub}')$.

- $\mathsf{key}' \leftarrow \mathsf{PFE.Rep}(\mathsf{pub}', w')$*: Interpret* pub' *as an obfuscated program and return* $\mathsf{key}' \leftarrow \mathsf{pub}'(w')$.

Theorem 1. *Construction 1 is a secure and weakly-private* $(\mathcal{M}, \mathcal{W}, \kappa, t, s, \epsilon)$*-fuzzy extractor.*

Proof of Theorem 1.

Correctness: Recall that pub' is an obfuscated MBCC circuit such that

$$\begin{aligned}
\mathsf{pub}'(w') &= \mathsf{Obf}\big(1^\lambda, \mathsf{MBCC}_{f_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}'}\big)(w') \\
&= \mathsf{MBCC}_{f_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}'}(w') \\
&= \begin{cases} \mathsf{key}' & \text{if } \ f_{\mathsf{pub}}(w') = \mathsf{key} \\ \perp & \text{otherwise.} \end{cases} \\
&= \begin{cases} \mathsf{key}' & \text{if } \ \mathsf{FE.Rep}(w', \mathsf{pub}) = \mathsf{key} \\ \perp & \text{otherwise.} \end{cases}
\end{aligned}$$

Then since FE is a fuzzy extractor, it is true that

$$\begin{aligned}
&\Pr\left[\mathsf{PFE.Rep}(\mathsf{pub}', w') = \mathsf{key}' \;\middle|\; (\mathsf{pub}', \mathsf{key}') \leftarrow \mathsf{PFE.Gen}(w) \text{ and } \mathsf{dist}(w, w') \le t\right] \\
&= \Pr\left[\mathsf{FE.Rep}(\mathsf{pub}, w') = \mathsf{key} \mid (\mathsf{pub}, \mathsf{key}) \leftarrow \mathsf{FE.Gen}(w) \text{ and } \mathsf{dist}(w, w') \le t\right] \\
&\ge 1 - \delta
\end{aligned}$$

and PFE is thus correct.

Security: We proceed by contradiction. Suppose PFE is not a secure fuzzy extractor, then there exists some PPT adversary $\mathcal{A}$ and polynomial $p(\lambda)$ such that

$$\left|\Pr[\mathcal{A}(\mathsf{key}', \mathsf{pub}') = 1] - \Pr[\mathcal{A}(U_\kappa, \mathsf{pub}') = 1]\right| > 1/p(\lambda)$$

where $(\mathsf{key}', \mathsf{pub}') \leftarrow \mathsf{PFE.Gen}(W)$ and $U_\kappa \xleftarrow{\$} \{0,1\}^\kappa$. Now note that $\mathsf{pub}' = \mathsf{Obf}(1^\lambda, \mathsf{MBCC}_{f_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}'})$ is distributional VBB secure. Define $r(\lambda) = 3p(\lambda)$ and let Sim be the simulator of $\mathcal{A}$ for polynomial $r(\lambda)$. Then we have

$$\left|\Pr[\mathcal{A}(\mathsf{pub}', \mathsf{key}) = 1] - \Pr[\mathsf{Sim}^{\mathsf{pub}'}(1^\lambda, 1^{|\mathsf{pub}'|}, \mathsf{key}) = 1]\right| \le \frac{1}{3p(\lambda)}. \tag{1}$$

Note that the above is also true if key is replaced by U_κ, a uniform random variable over $\{0,1\}^\kappa$. In other words, we have

$$\left|\Pr[\mathcal{A}(\mathsf{pub}', U_\kappa) = 1] - \Pr[\mathsf{Sim}^{\mathsf{pub}'}(1^\lambda, 1^{|\mathsf{pub}'|}, U_\kappa) = 1]\right| \le \frac{1}{3p(\lambda)}. \tag{2}$$

We adapt Canetti et al.'s lemma [CFP+16, Lemma 2]:

Lemma 1. *Let* U_κ *denote the uniform distribution over* $\{0,1\}^\kappa$*, then*

$$\begin{aligned}
&\Big|\Pr[\mathsf{Sim}^{\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}']}\left(1^\lambda, \left|\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}']\right|, \mathsf{key}'\right) = 1] \\
&- \Pr[\mathsf{Sim}^{\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}']}\left(1^\lambda, \left|\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}']\right|, U_\kappa\right) = 1]\Big| \\
&\le \frac{1}{3p(\lambda)}
\end{aligned}$$

Proof of Lemma 1. Fix any $u \in \{0,1\}^\kappa$, the lemma will follow by averaging over all u. The information about whether the key value, denoted V, is key or u can only be obtained by Sim through the query responses. First, we modify Sim to quit immediately when it gets a response not equal to $\perp$. Such Sim is equally successful at distinguishing between key and u since the first non-$\perp$ response tells Sim if its input is equal to key. Subsequent responses add nothing to this knowledge. Since Sim can make at most q queries, there are $q+1$ possible values for the view of Sim on a given input. Of those, q views consist of some number of non-$\perp$ responses followed by a $\perp$ response, and one view consists of all q responses equal to $\perp$. Then by [DRS04, Lemma 2.2b],

$$\begin{aligned}\tilde{H}_\infty(V|View(\mathsf{Sim}), \mathsf{aux}) &\geq \tilde{H}_\infty(V) - \log(q+1)\\ &\geq \alpha - \log(q+1).\end{aligned}$$

where $\mathsf{aux} = \big(|\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, \mathsf{key}, \mathsf{key}']|\big)$.

Thus, at each query, the probability that Sim gets a non-$\perp$ response and guesses V is at most $(q+1)/2^\alpha$. Since there are q queries of Sim, the overall probability is at most $q(q+1)/2^\alpha$. Then since 2^α is negligible in λ, there exists some λ_0 such that for all $\lambda \geq \lambda_0$, $q(q+1)/2^\alpha \leq 1/(3p(\lambda))$. □

We know continue the proof of Theorem 1, from Lemma 1, we have

$$\big|\Pr[\mathsf{Sim}^{\mathsf{pub}'}(1^\lambda, 1^{|\mathsf{pub}'|}, \mathsf{key}') = 1] - \Pr[\mathsf{Sim}^{\mathsf{pub}'}(1^\lambda, 1^{|\mathsf{pub}'|}, U_\kappa) = 1]\big| \leq \frac{1}{3p(\lambda)}$$

Using the triangle inequality on Equations 1, 2 and 3 we obtain

$$\big|\ \Pr[\mathcal{A}(\mathsf{pub}', \mathsf{key}') = 1] - \Pr[\mathcal{A}(\mathsf{pub}', U_\kappa) = 1]\ \big| \leq \frac{1}{p(\lambda)}$$

which is a contradiction and ends the proof of security.

Weak privacy: Let FE be an $(\mathcal{M}, \mathcal{W}, \ell, t, \epsilon)$-computational fuzzy extractor. Consider random variables W, aux, and U_ℓ, the uniform distribution over ℓ bit strings, and $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{FE.Gen}(W)$. Then by definition, for any PPT adversary $\mathcal{A}$, we have

$$|\Pr[\mathcal{A}(\mathsf{key}, \mathsf{pub}) = 1] - \Pr[\mathcal{A}(U_\ell, \mathsf{pub}) = 1]| \leq \epsilon$$

which implies

$$H_{\mathsf{HILL}}(\mathsf{key} \mid \mathsf{pub}) = \ell.$$

Since key' is sampled independently from key,

$$\begin{aligned}H_{\mathsf{HILL}}\big(\mathsf{key} \mid (\mathsf{FE.Rep}(\cdot, \mathsf{pub}), \mathsf{key}', \mathsf{aux})\big) &= H_{\mathsf{HILL}}\big(\mathsf{key} \mid (\mathsf{FE.Rep}(\cdot, \mathsf{pub}), \mathsf{aux})\big)\\ &= H_{\mathsf{HILL}}\big(\mathsf{key} \mid \mathsf{pub}\big)\\ &\geq \ell\end{aligned}$$

Let Obf be a distributional VBB secure obfuscator for ℓ-pseudo-entropy distributions. Then

$$\mathsf{pub}' = \mathsf{Obf}\big(1^\lambda, \mathsf{MBCC}_{\mathsf{FE.Rep}(\cdot,\mathsf{pub}),\mathsf{key},\mathsf{key}'}\big)$$

can be simulated and for every $\mathcal{A}$, there exists a simulator Sim such that for every predicate ϕ we have:

$$\Big|\Pr[\mathcal{A}(\mathsf{pub}', \mathsf{aux}) = \phi(\mathsf{pub}')] - \Pr[\mathsf{Sim}^{\mathsf{MBCC}_{\mathsf{FE.Rep}(\cdot,\mathsf{pub}),\mathsf{key},\mathsf{key}'}}(1^\lambda, \mathsf{param}, \mathsf{aux}) = \phi(\mathsf{pub}')]\Big| \leq \mathsf{ngl}(\lambda).$$

Note that if we set $\mathsf{aux} = \mathsf{key}'$ and since key' is drawn randomly and independently, it is true that

$$\begin{aligned}&\big|\Pr[\mathcal{A}(\mathsf{pub}', \mathsf{key}') = \phi(\mathsf{pub}')] - \Pr[\mathsf{Sim}(1^\lambda, \mathsf{param}, \mathsf{key}') = \phi(\mathsf{pub}')]\big|\\ &= \big|\Pr[\mathcal{A}(\mathsf{pub}', \mathsf{key}') = \phi(W)] - \Pr[\mathsf{Sim}(1^\lambda, \mathsf{param}) = \phi(W)]\big|\\ &\leq \mathsf{ngl}(\lambda)\end{aligned}$$

which concludes the proof that PFE is a weakly private fuzzy extractor. □

3.2 Weakly private FE from secure sketch and MBCC obfuscation

Our second construction builds weakly-private fuzzy extractors from non-private secure sketches and MBCC obfuscation. Although this construction relies on a secure sketch, like Wichs and Zirdelis's private secure sketch scheme, we show that the pairwise independent hash function they use isn't necessary. This reduces the amount of entropy required and allows support of a wider family of distributions. However, we build a fuzzy extractor not a secure sketch, some constructions may rely on the functionality of a secure sketch.

Construction 2 (Weakly Private Fuzzy Extractor from SS and MBCC)**.** *Let* $(\mathsf{Sketch}, \mathsf{Rec})$ *be an* $(\mathcal{M}, \mathcal{W}, \ell, t, \delta)$*-secure sketch and* Obf *be an obfuscator for* ℓ*-pseudo-entropy distributions over multi-bit compute-and-compare circuits. Then we can build an* $(\mathcal{M}, \mathcal{W}, \kappa, t, s, \epsilon)$*-fuzzy extractor* PFE *as follows:*

- $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{PFE.Gen}(w)$*:*
  1. *Compute* $\mathsf{SS} \leftarrow \mathsf{Sketch}(w)$*.*
  2. *Sample* $\mathsf{key} \xleftarrow{\$} \{0,1\}^\kappa$*.*
  3. *Define the circuit* $f_{\mathsf{SS}}(\cdot) := \mathsf{Rec}(\cdot, \mathsf{SS})$*.*
  4. *Compute* $\mathsf{pub} \leftarrow \mathsf{Obf}\big(1^\lambda, \mathsf{MBCC}_{f_{\mathsf{SS}}, w, \mathsf{key}}\big)$*.*
  5. *Output* $(\mathsf{key}, \mathsf{pub})$*.*
- $\mathsf{key} \leftarrow \mathsf{PFE.Rep}(\mathsf{pub}, w')$*: Interpret* pub *as an obfuscated program and return* $\mathsf{key} \leftarrow \mathsf{pub}(w')$*.*

Theorem 2. *Construction 2 is a secure and weakly private* $(\mathcal{M}, \mathcal{W}, \kappa, t, s, \epsilon)$*-fuzzy extractor.*

Proof of Theorem 2.

Correctness : Recall that pub' is an obfuscated MBCC circuit such that

$$\begin{aligned}
\mathsf{pub}(w') &= \mathsf{Obf}\big(1^\lambda, \mathsf{MBCC}_{f_{\mathsf{SS}}, w, \mathsf{key}}\big)(w') \\
&= \mathsf{MBCC}_{f_{\mathsf{SS}}, w, \mathsf{key}}(w') \\
&= \begin{cases} \mathsf{key} & \text{if } \ f_{\mathsf{SS}}(w') = w \\ \bot & \text{otherwise.} \end{cases} \\
&= \begin{cases} \mathsf{key} & \text{if } \ \mathsf{Rec}(w', \mathsf{SS}) = w \\ \bot & \text{otherwise.} \end{cases}
\end{aligned}$$

Then since $(\mathsf{Sketch}, \mathsf{Rec})$ is a secure sketch, it is true that

$$\begin{aligned}
&\Pr\left[\mathsf{PFE.Rep}(\mathsf{pub}, w') = \mathsf{key} \mid (\mathsf{pub}, \mathsf{key}) \leftarrow \mathsf{PFE.Gen}(w) \text{ and } \mathsf{dist}(w, w') \le t\right] \\
&= \Pr\left[\mathsf{Rec}(w', \mathsf{SS}) = w \mid \mathsf{SS} \leftarrow \mathsf{Sketch}(w) \text{ and } \mathsf{dist}(w, w') \le t\right] \\
&\ge 1 - \delta
\end{aligned}$$

and PFE is thus correct.

Security: This proof is the same as the security proof of Theorem 1.

Weak privacy: Let $(\mathsf{Sketch}, \mathsf{Rec})$ be an $(\mathcal{M}, \mathcal{W}, \ell, t, \delta)$-secure sketch. Then for random variables W, aux, and $\mathsf{SS} \leftarrow \mathsf{Sketch}(W)$ we have

$$H_{\mathsf{HILL}}(W \mid \mathsf{SS}) \ge \ell$$

Since key is sampled independently from all other values,

$$\begin{aligned}
H_{\mathsf{HILL}}\big(w \mid \mathsf{Rec}(\cdot, \mathsf{SS}), \mathsf{key}, \mathsf{aux})\big) &= H_{\mathsf{HILL}}\big(w \mid \mathsf{Rec}(\cdot, \mathsf{SS})\big) \\
&= H_{\mathsf{HILL}}\big(w \mid \mathsf{SS}\big) \ge \ell
\end{aligned}$$

Let Obf be a distributional VBB secure obfuscator for ℓ-pseudo-entropy distributions. Then

$$\mathsf{pub}' = \mathsf{Obf}\big(1^\lambda, \mathsf{MBCC}_{\mathsf{Rec}(\cdot,\mathsf{SS}),w,\mathsf{key}}\big)$$

can be simulated and for every $\mathcal{A}$, there exists a simulator Sim such that for every predicate ϕ we have:

$$\Big|\Pr[\mathcal{A}(\mathsf{pub},\mathsf{aux}) = \phi(\mathsf{pub})] - \Pr[\mathsf{Sim}^{\mathsf{MBCC}_{\mathsf{Rec}(\cdot,\mathsf{SS}),w,\mathsf{key}}}(1^\lambda,\mathsf{param},\mathsf{aux}) = \phi(\mathsf{pub})]\Big| \leq \mathsf{ngl}(\lambda)$$

Note that if we set $\mathsf{aux} = \mathsf{key}$ and since key is drawn randomly and independently, it is true that

$$\begin{aligned}&\Big|\Pr[\mathcal{A}(\mathsf{pub},\mathsf{key}) = \phi(\mathsf{pub})] - \Pr[\mathsf{Sim}(1^\lambda,\mathsf{param},\mathsf{key}) = \phi(\mathsf{pub})]\Big| \\ &= \Big|\Pr[\mathcal{A}(\mathsf{pub},\mathsf{key}') = \phi(W)] - \Pr[\mathsf{Sim}(1^\lambda,\mathsf{param}) = \phi(W)]\Big| \leq \mathsf{ngl}(\lambda)\end{aligned}$$

which concludes the proof that PFE is a weakly private fuzzy extractor. □

4 Robustness

We first define robustness of a fuzzy extractor.

Definition 10 (Robust Fuzzy extractor)**.** *Let* FE *be an* $(\mathcal{M},\mathcal{W},\ell,t,s,\epsilon)$*-fuzzy extractor with error* δ *as defined above.* FE *is a robust fuzzy extractor if for all* $W, W' \in \mathcal{W}$*, such that*

$$\Pr_{(w,w')\leftarrow(W,W')}[\mathsf{dist}(w,w') \leq t] = 1,$$

and for all adversaries $\mathcal{A}$*, the advantage of* $\mathcal{A}$ *in the following experiment is at most* $\mathsf{ngl}(\lambda)$*:*

1. *Sample* $(w,w') \leftarrow (W,W')$*.*
2. *Compute* $(\mathsf{key},\mathsf{pub}) \leftarrow \mathsf{FE.Gen}(w)$ *and send it to* $\mathcal{A}$*.*
3. $\mathcal{A}$ *outputs* pub' *and wins if* $\mathsf{pub}' \neq \mathsf{pub}$ *and* $\mathsf{FE.Rep}(\mathsf{pub}', w') \notin \{\perp, \mathsf{key}\}$*.*

We propose a generic technique to upgrade a fuzzy extractor and achieve robustness. This method relies on non-interactive zero-knowledge (NIZK) [DHLAW10]. We also show that this technique preserves privacy of the underlying fuzzy extractor. This yields a robust, weakly-private fuzzy extractor construction in the common reference string (CRS) model.

Definition 11 (True simulation extractable NIZK)**.** *Let* R *be an NP relation on pairs* (x,w) *with corresponding language* $L_R = \{x : \exists w \text{ such that } (x,w) \in R\}$*. A true-simulation extractable non-interactive zero-knowledge (NIZK) argument for a relation* R *consists of three algorithms* $(\mathsf{Setup},\mathsf{Prove},\mathsf{Verify})$ *with the following syntax:*

- $(\mathtt{crs},\mathsf{TK},\mathsf{EK}) \leftarrow \mathsf{Setup}(1^\lambda)$*: creates a common reference string* $\mathtt{crs}$*, a trapdoor* TK*, and an extraction key* EK*.*
- $\pi \leftarrow \mathsf{Prove}(\mathtt{crs},x,w)$*: creates an argument* π *that* $R(x,w) = 1$*.*
- $0/1 \leftarrow \mathsf{Verify}(\mathtt{crs},x,\pi)$*: verifies whether or not the argument* π *is correct.*

For presentation simplicity, we omit $\mathtt{crs}$ *in the* Prove *and* Verify*. We require that the following three properties hold:*

- ***Completeness.*** *For any* $(x,w) \in R$*, if* $(\mathtt{crs},\mathsf{TK},\mathsf{EK}) \leftarrow \mathsf{Setup}(1^\lambda)$*,* $\pi \leftarrow \mathsf{Prove}(x,w)$*, then* $\mathsf{Verify}(x,\pi) = 1$*.*
- ***Soundness.*** *For any PPT adversary* $\mathcal{A}$*, the following probability is negligible: for* $(\mathtt{crs},\mathsf{TK},\mathsf{EK}) \leftarrow \mathsf{Setup}(1^\lambda)$*,* $(x^*,\pi^*) \leftarrow \mathcal{A}(\mathtt{crs})$ *such that* $x^* \notin L_R$ *but* $\mathsf{Verify}(x^*,\pi^*) = 1$*.*

- ***Composable Zero-knowledge.*** *There exists a PPT simulator* Sim *such that for any PPT* $\mathcal{A}$*, the advantage (the probability* $\mathcal{A}$ *wins minus one half) is negligible in the following game.*
  - *The challenger samples* $(\mathtt{crs}, \mathsf{TK}, \mathsf{EK}) \leftarrow \mathsf{Setup}(1^\lambda)$ *and sends* $(\mathtt{crs}, \mathsf{TK})$ *to* $\mathcal{A}$
  - $\mathcal{A}$ *chooses* $(x, w) \in R$ *and sends to the challenger.*
  - *The challenger generates* $\pi_0 \leftarrow \mathsf{Prove}(x, w)$*,* $\pi_1 \leftarrow \mathsf{Sim}(x, \mathsf{TK})$*, and then samples a random bit* $b \leftarrow \{0, 1\}$*. Then he sends* π_b *to* $\mathcal{A}$*.*
  - $\mathcal{A}$ *outputs a guess bit* b'*, and wins if* $b' = b$*.*
- ***Extractibility.*** *Additionally, true simulation extractability requires that there exists a PPT extractor* Ext *such that for any PPT adversary* $\mathcal{A}$*, the probability* $\mathcal{A}$ *wins is negligible in the following game:*
  - *The challenger picks* $(\mathtt{crs}, \mathsf{TK}, \mathsf{EK}) \leftarrow \mathsf{Setup}(1^\lambda)$ *and sends* $\mathtt{crs}$ *to* $\mathcal{A}$*.*
  - $\mathcal{A}$ *is allowed to make oracle queries to the simulation algorithm* $\mathsf{Sim}'((x, w), \mathsf{TK})$ *adaptively.* Sim' *first checks if* $(x, w) \in R$ *and returns* $\mathsf{Sim}(x, TK)$ *if that is the case.*
  - $\mathcal{A}$ *outputs a tuple* x^*, L^*, π^**.*
  - *The challenger runs the extractor* $w^* \leftarrow \mathsf{Ext}(L^*, (x^*, \pi^*), \mathsf{EK})$*.*
  - $\mathcal{A}$ *wins if 1) the pair* (x^*, L^*) *was not part of the simulator query, 2) the proof* π^* *verifies, and (3)* $R(x^*, w^*) = 0$*.*

Construction 3 (Robust, weakly-private fuzzy extractor)**.** *Let* FE *be a weakly-private fuzzy extractor and* $(\mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify})$ *be a NIZK system for language* $\mathcal{L} = \{\mathsf{pub} \mid \mathsf{FE.Gen}(w; r) = (\mathsf{pub}, \mathsf{key})\}$*. Here, the statement is* $\mathsf{pub} = \mathsf{FE.Gen}(w; r)$ *and the witness is the pair of values* (w, r)*, where* w *is the original reading and* r *the internal randomness of* Gen*.*

- $(\mathsf{key}, \mathsf{pub}^*) \leftarrow \mathsf{FE}'.\mathsf{Gen}(w)$*:*
  1. *Sample* $(\mathtt{crs}, \mathsf{TK}, \mathsf{EK}) \leftarrow \mathsf{Setup}(1^\lambda)$*.*
  2. *Compute* $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{FE.Gen}(w; r)$*.*
  3. *Compute* $\pi \leftarrow \mathsf{Prove}(\mathtt{crs}, \mathsf{pub}, w, r)$ *and set* $\mathsf{pub}^* = (\mathsf{pub}, \pi)$*.*
  4. *Output* $(\mathsf{key}, \mathsf{pub}^*)$*.*
- $\mathsf{key}' \leftarrow \mathsf{FE}'.\mathsf{Rep}(\mathsf{pub}^*, w')$*:*
  1. *Run* $b \leftarrow \mathsf{Verify}(\mathtt{crs}, \mathsf{pub}, \pi)$ *and output* $\perp$ *if* $b = 0$*.*
  2. *Output* $\mathsf{key}' \leftarrow \mathsf{FE.Rep}(\mathsf{pub}, w')$*.*

Theorem 3. *Let* FE *be an weakly-private,* $(\mathcal{M}, \mathcal{W}, \ell, 2t, s, \epsilon)$*-fuzzy extractor and* $(\mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify})$ *be a NIZK system. Then* FE' *as described in Construction 3 is a weakly-private, robust,* $(\mathcal{M}, \mathcal{W}, \ell, t, s, \epsilon)$*-fuzzy extractor.*

Note that in this theorem the underlying fuzzy extractor FE corrects $\mathbf{2t}$ errors while the resulting fuzzy extractor FE' corrects only $\mathbf{t}$ errors. This is important for the corresponding proof to work. This requirement was present in some prior robustness upgrades for fuzzy extractors, see Table 2.

Proof of Theorem 3.

Correctness: Correctness is straightforward from the correctness of the underlying fuzzy extractor and the completeness of the NIZK system.

Security: Security is straightforward from the security of the underlying fuzzy extractor and the zero-knowledge property of the NIZK system.

Privacy: Privacy is straightforward from the privacy of the underlying fuzzy extractor and the zero-knowledge property of the NIZK system. We provide a short sketch below.

Let Sim denote a simulator for the underlying weakly private FE. Suppose FE' is not weakly private, then there exists an adversary $\mathcal{A}'$, such that for any simulators $\mathsf{Sim}'(|\mathsf{pub}|, |\pi|, |\mathsf{key}|)$, we have

$$\left|\Pr[\mathcal{A}'(\mathsf{pub}, \pi, \mathsf{key}) = 1] - \Pr[\mathsf{Sim}'(|\mathsf{pub}|, |\pi|, |\mathsf{key}|) = 1]\right| > \mathsf{ngl}(\lambda)$$

We note that $\mathsf{Sim}'(|\mathsf{pub}|, |\pi|, |\mathsf{key}|) = \mathsf{Sim}(|\mathsf{pub}|, |\mathsf{key}|)$ is one such valid simulator.
Then we can build an adversary $\mathcal{A}$ for FE,

1. Receive inputs pub and key.
2. Run NIZK setup $(\mathsf{TK}, \mathsf{EK}) \leftarrow \mathsf{Setup}(1^\lambda)$.
3. Run the NIZK simulator $\pi \leftarrow \mathsf{Sim}_{\mathrm{NIZK}}(\mathsf{pub}, \mathsf{TK})$.
4. Run the FE' adversary $b \leftarrow \mathcal{A}'(\mathsf{pub}, \pi, \mathsf{key})$.
5. Return b.

Then

$$|\Pr[\mathcal{A}(\mathsf{pub}, \mathsf{key}) = 1] - \Pr[\mathsf{Sim}(|\mathsf{pub}|, |\mathsf{key}|) = 1]| > \mathsf{ngl}(\lambda)$$

which is a contradiction of FE's weak privacy.

Robustness: We proceed by contradiction. Suppose FE' is not a robust fuzzy extractor, that is, for distributions W, W' such that $\mathsf{dist}(W, W') \leq t$, there exists a PPT adversary $\mathcal{A}'_{\mathsf{FE}}$ such that

$$\Pr_{(w,w') \leftarrow (W,W')}\left[\begin{array}{r|r} \mathsf{Verify}(\mathtt{crs}, \mathsf{pub}', \pi') = 1 & (\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{FE.Gen}(w) \\ \wedge\ \mathsf{FE.Rep}(\mathsf{pub}', w') \neq \{\mathsf{key}, \perp\} & \pi \leftarrow \mathsf{Prove}(\mathtt{crs}, \mathsf{pub}, \mathsf{key}, w) \\ \wedge\ (\mathsf{pub}' \neq \mathsf{pub} \vee \pi' \neq \pi) & (\mathsf{pub}', \pi') \leftarrow \mathcal{A}_{\mathsf{PFE}}(\mathsf{key}, \mathsf{pub}, \pi) \end{array}\right] > \mathsf{ngl}(\lambda).$$

We can then build a PPT distinguisher $\mathcal{A}$ for the fuzzy extractor security game as follows:

1. Receive $(\mathsf{pub}, \mathsf{key}_b)$ from the challenger, where for $w \leftarrow W$, $(\mathsf{pub}, \mathsf{key}) \leftarrow \mathsf{FE.Gen}(w; r)$ and for $b \in \{0, 1\}$, $\mathsf{key}_1 = \mathsf{key}$ and $\mathsf{key}_0 = U_\ell$.
2. Sample $(\mathtt{crs}, \mathsf{TK}, \mathsf{EK}) \leftarrow \mathsf{Setup}(1^\lambda)$ from the NIZK proof system.
3. Run the NIZK simulator $\pi \leftarrow \mathsf{Sim}(\mathsf{pub}, \mathsf{TK})$.
4. Send $(\mathsf{key}_b, \mathsf{pub}, \pi)$ to $\mathcal{A}'_{\mathsf{FE}}$ and receives back (pub', π').
5. Run the NIZK extractor $(w^*, r^*) \leftarrow \mathsf{Ext}(\mathsf{pub}', \pi', \mathsf{EK})$.
6. Run $\mathsf{key}' \leftarrow \mathsf{FE.Rep}(\mathsf{pub}, w^*)$.
7. If $\mathsf{key}' = \mathsf{key}_b$ return 1, otherwise return 0.

Recall that a break in robustness requires $(\mathsf{pub}', \pi') \neq (\mathsf{pub}, \pi)$ and π' to be a valid proof. Suppose $\mathsf{pub}' = \mathsf{pub}$, then $\mathsf{dist}(w, w^*) \leq t$ and $\mathsf{FE.Rep}(\mathsf{pub}, w^*) = \mathsf{key}$. So $\mathsf{FE}'.\mathsf{Rep}(\mathsf{pub}'||\pi', w^*) = \mathsf{key}$, which does not count as a break of the robustness property.

So it must be true that $\mathsf{pub}' \neq \mathsf{pub}$. In this situation, $\mathcal{A}'_{\mathsf{FE}}$ outputs the pair (pub', π') such that for some w', with $\mathsf{dist}(w, w') \leq t$, $\mathsf{FE}'.\mathsf{Rep}(\mathsf{pub}'||\pi', w') = \mathsf{key}^* \neq \mathsf{key}$. Then the NIZK extractor outputs point w^* such that $\mathsf{FE.Gen}(w^*; r^*) = (\mathsf{key}^*, \mathsf{pub}'||\pi')$. So $\mathsf{dist}(w, w') \leq t$ and $\mathsf{dist}(w^*, w') \leq t$, which means that $\mathsf{dist}(w, w^*) \leq 2t$. Finally, since FE corrects $2t$ errors, when $b = 1$, $\mathsf{key}' = \mathsf{FE.Rep}(\mathsf{pub}, w^*) = \mathsf{key} = \mathsf{key}_1$ and

$$|\Pr[\mathcal{A}(\mathsf{pub}, \mathsf{key}) = 1] - \Pr[\mathcal{A}(\mathsf{pub}, U_\ell) = 1]| > \mathsf{ngl}(\lambda)$$

which concludes our proof.

□

5 Reuse

In this section, we show that one cannot hope to compose MBCC obfuscation with an auxiliary input secure digital locker. We then show this implies a impossibility of a variant of private fuzzy extractors that can be constructed from MBCC obfuscation. This variant never outputs a value outside of the ball of the enrolled value.

Definition 12. *Let* $(\mathsf{Gen}, \mathsf{Rep})$ *be an* $(\mathcal{M}, \mathcal{W}, \ell, t, \epsilon)$*-fuzzy extractor with error* δ *(Definition 7). The pair is perfectly correct if for all* $w, w' \in \mathcal{M}$ *such that* $\mathsf{dist}(w, w') > t$*:*

$$\Pr\left[\perp \leftarrow \mathsf{Rep}(\mathsf{pub}, w') \mid (\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{Gen}(w)\right] \geq 1 - \mathsf{ngl}(\lambda).$$

We assume that any randomness for Rep *is included in the string* pub *so this probability statement is only over the randomness of* Gen*.*

We now define digital lockers, which have the same functionality as perfectly correct fuzzy extractors for $t = 0$. Digital lockers [CTKVW10] are also a specific case of MBCC obfuscation where the function is the identity function, $f(x) = \mathsf{id}(x) = x$.

Definition 13 (Digital Locker)**.** *An* $(\mathcal{W}, n)$*-digital locker is a pair of PPT algorithms* $(\mathsf{lock}, \mathsf{unlock})$ *where for all* $\mathsf{val} \in D^\lambda$ *and* $\mathsf{key} \in \{0,1\}^n$*,*

- $\mathsf{unlock} \leftarrow \mathsf{lock}(\mathsf{val}, \mathsf{key})$
- $\mathsf{key}' \leftarrow \mathsf{unlock}(\mathsf{val}')$

such that the following properties are true:

1. ***Completeness****: For all* $\mathsf{val} \in \mathsf{D}^\lambda, \mathsf{key} \in \{0,1\}^n$ *it holds that*

$$\Pr[\mathsf{unlock}(\cdot) \equiv I_{\mathsf{val},\mathsf{key}}(\cdot) \mid \mathsf{unlock} \leftarrow \mathsf{lock}(\mathsf{val}, \mathsf{key})] \geq 1 - \mathsf{ngl}(\lambda),$$

   where the probability is over the randomness of lock*. Here* $I_{\mathsf{val},\mathsf{key}}$ *is a function that returns* key *when provided input* val*, otherwise* $I_{\mathsf{val},\mathsf{key}}$ *returns* $\perp$*.*

2. ***Virtual Black Box Security****: For all PPT* $\mathcal{A}$ *and* $p = \mathsf{poly}(\lambda)$*,* $\exists \mathsf{Sim}$ *and* $q(\lambda) = \mathsf{poly}(\lambda)$ *such that for all large enough* $\lambda \in \mathbb{N}$*,* $\forall \mathsf{val} \in \mathsf{D}_\lambda, \mathsf{key} \in \{0,1\}^n, \mathcal{P} : \mathsf{D}_\lambda \times \{0,1\}^n \mapsto \{0,1\}$*,*

$$\left|\Pr[\mathcal{A}(\mathsf{lock}(\mathsf{val}, \mathsf{key})) = \mathcal{P}(\mathsf{val}, \mathsf{key})] - \Pr[\mathsf{Sim}^{I_{\mathsf{val},\mathsf{key}}}(1^\lambda) = \mathcal{P}(\mathsf{val}, \mathsf{key})]\right| \leq \frac{1}{p(\lambda)},$$

   where Sim *is allowed* $q(\lambda)$ *oracle queries to* $I_{\mathsf{val},\mathsf{key}}$ *and the probabilities are over the internal randomness of* $\mathcal{A}$ *and* lock*, and of* Sim*, respectively.*

Construction One can construct a perfectly correct private fuzzy extractor by applying Construction 2 on a well-formed secure sketch [BDK+05, Definition 4]. A well-formed secure sketch on input w' never outputs a value with distance $\geq t$ from w'. One can always construct a well-formed secure sketch with no loss in parameters by adding a distance check before output.

Since the circuit being obfuscated in Construction 2 only has an output when the output of the secure sketch is equal to w, these two modifications suffice to form a (private) perfectly correct fuzzy extractor.

Proposition 1. *Perfectly correct private fuzzy extractors with auxiliary input imply digital lockers with auxiliary inputs.*

Proof. This proposition easily follows by setting the required distance t equal to 0. □

Definition 14 (Collision-resistant Hash function)**.** *Consider function* $h : \{0,1\}^n \to \{0,1\}^m$*,* h *is a collision-resistant hash function if the following are true:*

1. ***Compression****:* $m < n$*.*

2. ***Collision-resistance**: For any PPT adversary $\mathcal{A}$,*

$$\Pr\left[(x_0, x_1) \leftarrow \mathcal{A}(1^n, h) \mid x_0 \neq x_1 \wedge h(x_0) = h(x_1)\right] \leq \mathsf{ngl}(n).$$

Theorem 4 (Private FE with auxiliary input impossibility)**.** *If dist-VBB obfuscation for MBCC programs with α-pseudo entropy and collision-resistant hash functions exist, no perfectly-correct private fuzzy extractor can be secure in the presence of unpredictability auxiliary inputs.*

Theorem 4. This proof is built from a main lemma (see Lemma 2) which is then combined with Proposition 1. Lemma 2 shows that digital lockers with auxiliary input for unpredictable sources cannot exist if dist-VBB obfuscation for MBCC programs with α-pseudo entropy exists.

Lemma 2 (Digital locker with auxiliary input impossibility)**.** *If dist-VBB obfuscation for MBCC programs with α-pseudo entropy and collision-resistant hash functions exist, then security for digital lockers with auxiliary inputs for unpredictable sources cannot be achieved.*

Lemma 2. Let U_x denote the universal circuit that takes as input circuit C and computes $U_x(C) = C(x)$. Define the following MBCC program

$$\mathsf{MBCC}[U_x, \mathsf{key}, x](C) = \begin{cases} x & \text{if } C \text{ is a well-formed } \mathsf{unlock} \text{ program and } C(x) = \mathsf{key}. \\ \perp & \text{otherwise.} \end{cases}$$

Let $h : \{0,1\}^{|x|} \to \{0,1\}^m$, with $m < |x|$, be a collision-resistant hash function. Suppose x and key are independent and let $\mathsf{aux} = h(x)$, then we have

$$H^{\mathsf{HILL}}(\mathsf{key} \mid U_x, x, \mathsf{aux}) \geq \alpha(\lambda)$$

which implies that there exists a dist-VBB obfuscator Obf for this MBCC circuit.
We now need to show that X remains unpredictable given $\mathsf{Obf}(\mathsf{MBCC}[U_X, \mathsf{key}, x])$, that is

$$H^{\mathsf{unp}}(X \mid \mathsf{Obf}(\mathsf{MBCC}[U_X, \mathsf{key}, x]) \geq \omega(\log \lambda)$$

In other words, we want to show that if $\mathsf{Obf}(\mathsf{MBCC}[U_x, \mathsf{key}, x])$ is dist-VBB secure, then for all PPT $\mathcal{A}$, we have

$$\Pr\left[\mathcal{A}\left(\mathsf{Obf}(\mathsf{MBCC}[U_x, \mathsf{key}, x])\right) = x\right] \leq \mathsf{ngl}(\lambda).$$

We proceed by contradiction. Suppose the above is not true and there exists a PPT $\mathcal{A}$ that can predict x from $\mathsf{Obf}(\mathsf{MBCC}[U_x, \mathsf{key}, x])$ with non-negligible probability. Then we can build a distinguisher for the MBCC obfuscation that breaks dist-IND security (which is equivalent to dist-VBB for evasive functions such as MBCC [BC10]). The distinguisher works as follows:

1. Receive P^* and $\mathsf{aux} = h(x)$ as inputs.
2. Run $x^* \leftarrow \mathcal{A}(P^*)$.
3. If $h(x^*) = h(x)$, return 1, otherwise return 0.

If $P^* = \mathsf{Obf}(1^\lambda, P)$, then $\mathcal{A}$ should be able to extract $x^* = x$ and $h(x^*) = h(x)$. However, if $P^* \leftarrow \mathsf{Sim}(1^\lambda, P.\mathsf{params})$, $\mathcal{A}$ should not be able to extract correct x^*. Then the probability that $x^* = x$ is $\frac{1}{2^n}$ and when $x^* \neq x$, $h(x^*) = h(x)$ with negligible probability. This is a contradiction of dist-IND security of the MBCC obfuscator so we conclude that X remains unpredictable.

We now need to show that that this construction breaks digital locker security. Recall that digital locker security is VBB, that is for any PPT adversary $\mathcal{A}$ and any polynomial p, there exists a simulator Sim such that

$$\Pr\left[\mathcal{A}(\mathsf{unlock}, \mathsf{aux}) = 1\right] - \Pr\left[\mathsf{Sim}^{\mathsf{unlock}(\cdot)}(1^\lambda, \mathsf{aux}) = 1\right] \leq \frac{1}{p(\lambda)}$$

where $\mathsf{unlock} \leftarrow \mathsf{lock}(\mathsf{val}, \mathsf{key})$.

It is obvious that this does not hold when we set $\mathsf{aux} = \mathsf{MBCC}[U_{\mathsf{val}}, \mathsf{key}, \mathsf{val}]$. Indeed, $\mathcal{A}$ can then run $\mathsf{aux}(\mathsf{unlock})$ and retrieve the correct val (and then key by running $\mathsf{unlock}(\mathsf{val})$), whereas Sim cannot. □

By chaining Lemma 2 and the contrapositive of Proposition 1, we obtain that if dist-VBB MBCC obfuscation exists then private fuzzy extractors with auxiliary inputs cannot be achieved, which conclude this Theorem's proof. □

Acknowledgements

The authors are grateful to anonymous reviewers for their help improving the manuscript. The authors thank Giorgos Zirdelis for helpful discussions. C.C. was supported by NSF grant #2141033. B.F. and M.R. were supported by NSF grants #2141033 and #2232813.

References

[ABC+18] Quentin Alamélou, Paul-Edmond Berthier, Chloé Cachet, Stéphane Cauchie, Beñjamin Fuller, Philippe Gaborit, and Sailesh Simhadri. Pseudoentropic isometries: A new framework for fuzzy extractor reusability. In *AsiaCCS*, 2018.

[ACEK17] Daniel Apon, Chongwon Cho, Karim Eldefrawy, and Jonathan Katz. Efficient, reusable fuzzy extractors from LWE. In *International Conference on Cyber Security Cryptography and Machine Learning*, pages 1–18. Springer, 2017.

[ACF+22] Daniel Apon, Chloe Cachet, Benjamin Fuller, Peter Hall, and Feng-Hao Liu. Nonmalleable digital lockers and robust fuzzy extractors in the plain model. In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology – ASIACRYPT 2022*, pages 353–383, Cham, 2022. Springer Nature Switzerland.

[BBC+14] Boaz Barak, Nir Bitansky, Ran Canetti, Yael Tauman Kalai, Omer Paneth, and Amit Sahai. Obfuscation for evasive functions. In *Theory of Cryptography Conference*, pages 26–51. Springer, 2014.

[BBCS91] Charles H Bennett, Gilles Brassard, Claude Crépeau, and Marie-Hélene Skubiszewska. Practical quantum oblivious transfer. In *Annual international cryptology conference*, pages 351–366. Springer, 1991.

[BBR88] Charles H. Bennett, Gilles Brassard, and Jean-Marc Robert. Privacy amplification by public discussion. *SIAM Journal on Computing*, 17(2):210–229, 1988.

[BC10] Nir Bitansky and Ran Canetti. On strong simulation and composable point obfuscation. In *Advances in Cryptology–CRYPTO 2010*, pages 520–537. Springer, 2010.

[BCKP17] Nir Bitansky, Ran Canetti, Yael Tauman Kalai, and Omer Paneth. On virtual grey box obfuscation for general circuits. *Algorithmica*, 79(4):1014–1051, 2017.

[BCP13] Julien Bringer, Hervé Chabanne, and Alain Patey. SHADE: Secure hamming distance computation from oblivious transfer. In *International Conference on Financial Cryptography and Data Security*, pages 164–176. Springer, 2013.

[BDCG13] Carlo Blundo, Emiliano De Cristofaro, and Paolo Gasti. EsPRESSo: efficient privacy-preserving evaluation of sample set similarity. In *Data Privacy Management and Autonomous Spontaneous Security*, pages 89–103. Springer, 2013.

[BDK+05] Xavier Boyen, Yevgeniy Dodis, Jonathan Katz, Rafail Ostrovsky, and Adam Smith. Secure remote authentication using biometric data. In *EUROCRYPT*, pages 147–163. Springer, 2005.

[BFM14] Christina Brzuska, Pooya Farshim, and Arno Mittelbach. Indistinguishability obfuscation and uces: The case of computationally unpredictable sources. In *Advances in Cryptology–CRYPTO 2014: 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I 34*, pages 188–205. Springer, 2014.

[BG11] Marina Blanton and Paolo Gasti. Secure and efficient protocols for iris and fingerprint identification. In *European Symposium on Research in Computer Security*, pages 190–209. Springer, 2011.

[BLMZ19] James Bartusek, Tancrède Lepoint, Fermi Ma, and Mark Zhandry. New techniques for obfuscating conjunctions. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 636–666. Springer, 2019.

[Boy04] Xavier Boyen. Reusable cryptographic fuzzy extractors. In *Proceedings of the 11th ACM conference on Computer and Communications Security*, pages 82–91, 2004.

[CDF+08] Ronald Cramer, Yevgeniy Dodis, Serge Fehr, Carles Padró, and Daniel Wichs. Detection of algebraic manipulation with applications to robust secret sharing and fuzzy extractors. In *Advances in Cryptology–EUROCRYPT 2008*, pages 471–488. Springer, 2008.

[CFP+16] Ran Canetti, Benjamin Fuller, Omer Paneth, Leonid Reyzin, and Adam Smith. Reusable fuzzy extractors for low-entropy distributions. In *Advances in Cryptology – EUROCRYPT*, pages 117–146. Springer, 2016.

[CTKVW10] Ran Canetti, Yael Tauman Kalai, Mayank Varia, and Daniel Wichs. On symmetric encryption and point obfuscation. In Daniele Micciancio, editor, *Theory of Cryptography*, pages 52–71, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[DCH+16] Siddhant Deshmukh, Henry Carter, Grant Hernandez, Patrick Traynor, and Kevin Butler. Efficient and secure template blinding for biometric authentication. In *Communications and Network Security (CNS), 2016 IEEE Conference on*, pages 480–488. IEEE, 2016.

[DFR21] Luke Demarest, Benjamin Fuller, and Alexander Russell. Code offset in the exponent. In *2nd Conference on Information-Theoretic Cryptography (ITC 2021)*, 2021.

[DHLAW10] Yevgeniy Dodis, Kristiyan Haralambiev, Adriana López-Alt, and Daniel Wichs. Efficient public-key cryptography in the presence of key leakage. In *Advances in Cryptology-ASIACRYPT 2010: 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings 16*, pages 613–631. Springer, 2010.

[DHP+18] Pierre-Alain Dupont, Julia Hesse, David Pointcheval, Leonid Reyzin, and Sophia Yakoubov. Fuzzy password-authenticated key exchange. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 393–424. Springer, 2018.

[DKK+12] Yevgeniy Dodis, Bhavana Kanukurthi, Jonathan Katz, Leonid Reyzin, and Adam Smith. Robust fuzzy extractors and authenticated key agreement from close secrets. *IEEE Transactions on Information Theory*, 58(9):6207–6222, 2012.

[DKRS06] Yevgeniy Dodis, Jonathan Katz, Leonid Reyzin, and Adam Smith. Robust fuzzy extractors and authenticated key agreement from close secrets. In Cynthia Dwork, editor, *Advances in Cryptology - CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 232–250. Springer Berlin Heidelberg, 2006.

[DORS08] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM Journal on Computing*, 38(1):97–139, 2008.

[DRS04] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In Christian Cachin and Jan L. Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004*, pages 523–540, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[DS05] Yevgeniy Dodis and Adam Smith. Correcting errors without leaking partial information. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 654–663, 2005.

[EHKM11] David Evans, Yan Huang, Jonathan Katz, and Lior Malka. Efficient privacy-preserving biometric identification. In *Proceedings of the 17th conference Network and Distributed System Security Symposium, NDSS*, 2011.

[FMR13] Benjamin Fuller, Xianrui Meng, and Leonid Reyzin. Computational fuzzy extractors. In *Advances in Cryptology-ASIACRYPT 2013*, pages 174–193. Springer, 2013.

[FMR20] Benjamin Fuller, Xianrui Meng, and Leonid Reyzin. Computational fuzzy extractors. *Information and Computation*, page 104602, 2020.

[FP19] Benjamin Fuller and Lowen Peng. Continuous-source fuzzy extractors: source uncertainty and insecurity. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pages 2952–2956. IEEE, 2019.

[FRS16] Benjamin Fuller, Leonid Reyzin, and Adam Smith. When are fuzzy extractors possible? In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 277–306. Springer, 2016.

[FRS20] Benjamin Fuller, Leonid Reyzin, and Adam Smith. When are fuzzy extractors possible? *IEEE Transactions on Information Theory*, 66(8):5282–5298, 2020.

[FT21] Hanwen Feng and Qiang Tang. Computational robust (fuzzy) extractors for crs-dependent sources with minimal min-entropy. In *Theory of Cryptography Conference*, pages 689–717. Springer, 2021.

[Ful23] Benjamin Fuller. Impossibility of efficient information-theoretic fuzzy extraction. Cryptology ePrint Archive, Paper 2023/172, 2023. `https://eprint.iacr.org/2023/172`.

[GGH13a] Sanjam Garg, Craig Gentry, and Shai Halevi. Candidate multilinear maps from ideal lattices. In *Advances in Cryptology–EUROCRYPT 2013*, pages 1–17. Springer, 2013.

[GGH+13b] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. *Proc. of FOCS*, 2013.

[GKW17] R. Goyal, V. Koppula, and B. Waters. Lockable obfuscation. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 612–621, Los Alamitos, CA, USA, oct 2017. IEEE Computer Society.

[GZ19] Steven D. Galbraith and Lukas Zobernig. Obfuscated fuzzy hamming distance and conjunctions from subset product problems. In *Theory of Cryptography*, 2019. https://eprint.iacr.org/2019/620.

[HAD06] Feng Hao, Ross Anderson, and John Daugman. Combining crypto with biometrics effectively. *Computers, IEEE Transactions on*, 55(9):1081–1088, 2006.

[HILL99] Johan HÅstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[HLR07] Chun-Yuan Hsiao, Chi-Jen Lu, and Leonid Reyzin. Conditional computational entropy, or toward separating pseudoentropy from compressibility. In Moni Naor, editor, *Advances in Cryptology - EUROCRYPT 2007*, pages 169–186, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[ŠTO05] Boris Škorić, Pim Tuyls, and Wil Ophey. Robust key extraction from physical uncloneable functions. In *Applied Cryptography and Network Security: Third International Conference, ACNS 2005, New York, NY, USA, June 7-10, 2005. Proceedings 3*, pages 407–422. Springer, 2005.

[WCD+17] Joanne Woodage, Rahul Chatterjee, Yevgeniy Dodis, Ari Juels, and Thomas Ristenpart. A new distribution-sensitive secure sketch and popularity-proportional hashing. In *Annual International Cryptology Conference*, pages 682–710. Springer, 2017.

[WL18] Yunhua Wen and Shengli Liu. Robustly reusable fuzzy extractor from standard assumptions. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 459–489. Springer, 2018.

[WLG19] Yunhua Wen, Shengli Liu, and Dawu Gu. Generic constructions of robustly reusable fuzzy extractor. In *IACR International Workshop on Public Key Cryptography*, pages 349–378. Springer, 2019.

[WLH18] Yunhua Wen, Shengli Liu, and Shuai Han. Reusable fuzzy extractor from the decisional Diffie–Hellman assumption. *Designs, Codes and Cryptography*, Jan 2018.

[WZ17] Daniel Wichs and Giorgos Zirdelis. Obfuscating compute-and-compare programs under lwe. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 600–611. IEEE, 2017.

[Zha19] Mark Zhandry. The magic of elfs. *Journal of Cryptology*, 32:825–866, 2019.

A Privacy vs FE security

Showing that fuzzy extractor security does not imply privacy is straightforward. Let FE' be a fuzzy extractor for which $\mathsf{pub}' = w_1||\mathsf{pub}$, where $w_1 \in \{0,1\}$ denotes the first bit of w and pub is a valid public value such that $\mathsf{key} \leftarrow \mathsf{FE.Rep}(\mathsf{pub}, w^*)$ when $\mathsf{dist}(w, w^*) \leq t$. Then it is obvious that even though FE' is a secure fuzzy extractor, it is not private.

We will now show that the reverse is also not true.

Theorem 5. *Privacy (Definition 8) does not imply fuzzy extractor security (Definition 7).*

Proof of Theorem 5. We will prove this by presenting a counter example. Consider the following construction:

- $(\mathsf{pub}, \mathsf{key}) \leftarrow \mathsf{Gen}(w)$: key is sampled uniformly at random and pub is an obfuscation of the program p such that, for inputs $x \in \{0,1\}^*$ and $b \in \{0,1\}$,

$$p(b,x) = \begin{cases} \mathsf{key} & \text{if } \ b = 1 \text{ and } \mathsf{dist}(w,x) \leq t \\ \top & \text{if } \ b = 0 \text{ and } x = \mathsf{key} \\ \bot & \text{otherwise.} \end{cases}$$

- $\mathsf{key} \leftarrow \mathsf{Rep}(\mathsf{pub}, b, w')$: run $\mathsf{pub}(1, w')$ and return its output.

Notice that for $w, w' \in \mathcal{W}$ such that $\mathsf{dist}(w, w') \leq t$, we have

$$\Pr\left[\mathsf{key} \leftarrow \mathsf{Rep}(\mathsf{pub}, w') \mid (\mathsf{pub}, \mathsf{key}) \leftarrow \mathsf{Gen}(w)\right] \geq 1 - \mathsf{ngl}(\lambda)$$

which is the expected behavior of a fuzzy extractor. Furthermore, note that this construction is private since by the obfuscation definition, for any PPT adversary $\mathcal{A}$, there exists simulator Sim such that for any predicate ϕ

$$\left|\Pr[\mathcal{A}(\mathsf{pub}, \mathsf{key}) = \phi(W)] - \Pr[\mathsf{Sim}(1^\lambda, 1^{|\mathsf{pub}|}, 1^{|\mathsf{key}|}) = \phi(W)]\right| \leq \mathsf{ngl}(\lambda)$$

Now let's check fuzzy extractor security. Consider the following experiment:

1. Run $(\mathsf{key}, \mathsf{pub}) \leftarrow \mathsf{Gen}(w)$.
2. Draw $b \leftarrow \{0,1\}$.
3. If $b = 0$, sample $U_\ell \xleftarrow{\$} \{0,1\}^\ell$ and send (U_ℓ, pub) to $\mathcal{A}$. Otherwise, send $(\mathsf{key}, \mathsf{pub})$ to $\mathcal{A}$.
4. $\mathcal{A}$ outputs $b' \in \{0,1\}$ and wins if $b' = b$.

$\mathcal{A}$ has a straightforward way of winning this experiment by running $\mathsf{pub}(0, x)$, where $x = \mathsf{key}$ or $x = U_\ell$ depending on drawn b. Then $\mathcal{A}$ outputs $b' = 1$ if $\mathsf{pub}(0, x) = \top$ and $b' = 0$ if $\mathsf{pub}(0, x) = \perp$. Thus we have

$$|\Pr[\mathcal{A}(\mathsf{key}, \mathsf{pub}) = 1] - \Pr[\mathcal{A}(U_\ell, \mathsf{pub}) = 1]| > \mathsf{ngl}$$

and we can conclude that this construction, although private, is not a secure fuzzy extractor. □

B Reusability from Composable MBCC Obfuscation

Reusability for Constructions 1 and 2 is achievable when the MBCC obfuscator is composable. We start by defining reuse.

Definition 15 (Reusable Fuzzy extractor [CFP+16])**.** *Let* FE *be an* $(\mathcal{M}, \mathcal{W}, \ell, t, s, \epsilon)$*-fuzzy extractor with error* δ *as defined above. Let* $(W_1, \cdots, W_\rho)$ *be* $\rho \in \mathbb{N}$ *correlated variables such that* $W_i \in \mathcal{W}$*. Let adversary* $\mathcal{A}$ *be a PPT adversary, then for all* $j \in [1, \rho]$*:*

1. *The challenger samples* $w_j \leftarrow W_j$ *and computes* $(\mathsf{key}_j, \mathsf{pub}_j) \leftarrow \mathsf{FE.Gen}(w)$*.*
2. *The challenger samples a uniform* $u \xleftarrow{\$} \{0,1\}^\ell$ *and sets* $K_0 = \mathsf{key}_i$ *and* $K_1 = u$*.*
3. *The challenger draws* $b \xleftarrow{\$} \{0,1\}$ *and sends to* $\mathcal{A}$

$$(\mathsf{key}_1, \cdots, \mathsf{key}_{i-1}, K_b, \mathsf{key}_{i+1}, \cdots, \mathsf{key}_\rho, \mathsf{pub}_1, \cdots, \mathsf{pub}_\rho)$$

4. $\mathcal{A}$ *outputs* $b' \in \{0,1\}$ *and wins if* $b' = b$*.*

We denote the above experiment as $\mathit{Exp}^{\mathsf{reusable}}_{\mathcal{A},b}$*, the advantage of* $\mathcal{A}$ *is*

$$\mathsf{Adv}(\mathcal{A}) = \left|\Pr[\mathit{Exp}^{\mathsf{reusable}}_{\mathcal{A},0} = 1] - \Pr[\mathit{Exp}^{\mathsf{reusable}}_{\mathcal{A},1} = 1]\right|.$$

FE *is a* (ρ, ϵ)*-reusable fuzzy extractor if for all* $\mathcal{A}$*, for all* $i \in [1, \rho]$ *the advantage of* $\mathcal{A}$ *is at most* ϵ*.*

However, as we show in Section 5 this is not possible without restricting the class of circuits being obfuscated.

Definition 16 (ℓ-Composable Obfuscation with auxiliary input)**.** Obf *is a* ℓ*-composable obfuscator for distribution class* $\mathcal{D}$ *over the family of circuits* $\mathcal{P}_\lambda$ *if for any PPT adversary* $\mathcal{A}$ *and polynomial* p*, there exists a simulator* Sim *such that for every distribution ensemble* $D = \{D_\lambda\} \in \mathcal{D}$ *and* $(P_1, \cdots, P_\ell, \mathsf{aux}) \leftarrow D_\lambda$*, with* $\ell = \mathsf{poly}(\lambda)$*,*

$$\Big|\Pr[\mathcal{A}\big(\mathsf{Obf}(P_1), \cdots, \mathsf{Obf}(P_\ell), \mathsf{aux}\big) = 1] - \Pr[\mathsf{Sim}^{P_1, \cdots, P_\ell}(1^{|P_1|}, \cdots, 1^{|P_\ell|}, \mathsf{aux}) = 1]\Big| \leq \frac{1}{p(\lambda)}$$

Theorem 6. *Let* Obf *be a composable dist-VBB obfuscator for MBCC circuits, then Constructions 1 and 2 are reusable.*

Proof of Theorem 6. Suppose PFE is not a reusable fuzzy extractor, that is, there exists a PPT adversary $\mathcal{A}$ and a polynomial $p(\lambda)$ such that for all $1 \leq j \leq \rho$:

$$\begin{aligned}&\big|\ \Pr[\mathcal{A}(\mathsf{key}_1, \cdots, \mathsf{key}_\rho, \mathsf{pub}_1, \cdots, \mathsf{pub}_\rho) = 1] \\ &- \Pr[\mathcal{A}(\mathsf{key}_1, \cdots, \mathsf{key}_{i-1}, U_\ell, \mathsf{key}_{i+1}, \cdots, \mathsf{key}_\rho, \mathsf{pub}_1, \cdots, \mathsf{pub}_\rho) = 1]\ \big| > \frac{1}{p(\lambda)}\end{aligned}$$

where U_ℓ is a uniform random string in $\{0,1\}^\ell$.
Remember that Obf is a composable obfuscator for $\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}]$. Let $r(\lambda) = 3p(\lambda)$ and suppose Sim is the simulator for $\mathcal{A}$ for $r(\lambda)$, then we have

$$\begin{aligned}&\big|\Pr[\mathcal{A}(\{\mathsf{Obf}(1^\lambda, \mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}_i])\}_{i=1}^\rho, \mathsf{aux}) = 1] \\ &- \Pr[\mathsf{Sim}^{\{\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}_i]\}_{i=1}^\rho}(1^\lambda, \{|\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}_i]|\}_{i=1}^\rho, \mathsf{aux}) = 1]\big| \leq \frac{1}{3p(\lambda)}\end{aligned}$$

Note that in Construction 1, $\mathsf{pub}_i = \mathsf{Obf}(1^\lambda, \mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}'}, k, \mathsf{key}_i])$ and set $\mathsf{aux} = \mathsf{key}_1, \cdots, \mathsf{key}_\rho$ so we have

$$\begin{aligned}&\big|\Pr[\mathcal{A}(\{\mathsf{pub}_i\}_{i=1}^\rho, \{\mathsf{key}_i\}_{i=1}^\rho) = 1] \\ &- \Pr[\mathsf{Sim}^{\{\mathsf{pub}_i\}_{i=1}^\rho}(1^\lambda, \{|\mathsf{pub}_i|\}_{i=1}^\rho, \{\mathsf{key}_i\}_{i=1}^\rho) = 1]\big| \leq \frac{1}{3p(\lambda)}\end{aligned} \tag{3}$$

Notice that this also holds if we replace key_j by an independent uniform random variable U_ℓ over $\{0,1\}^\ell$. Then for any $j \in \{1, \rho\}$ we have:

$$\begin{aligned}&\big|\Pr[\mathcal{A}(\{\mathsf{pub}_i\}_{i=1}^\rho, \mathsf{key}_1, \cdots, \mathsf{key}_{j-1}, U_\ell, \mathsf{key}_{j+1}, \cdots, \mathsf{key}_\rho) = 1] \\ &- \Pr[\mathsf{Sim}^{\{\mathsf{pub}_i\}_{i=1}^\rho}(1^\lambda, \{|\mathsf{pub}_i|\}_{i=1}^\rho, \mathsf{key}_1, \cdots, \mathsf{key}_{j-1}, U_\ell, \mathsf{key}_{j+1}, \cdots, \mathsf{key}_\rho) = 1]\big| \leq \frac{1}{3p(\lambda)}\end{aligned} \tag{4}$$

Again we adapt Canetti et al.'s lemma [CFP+16, Lemma 2]:

Lemma 3. *Let U_ℓ denote the uniform distribution over $\{0,1\}^\ell$, then for $1 \leq j \leq \rho$,*

$$\begin{aligned}&\Big|\Pr[\mathsf{Sim}^{\{\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}_i]\}_{i=1}^\rho}\left(1^\lambda, \left\{\left|\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}_i]\right|\right\}_{i=1}^\rho, \{\mathsf{key}\}_{i=1}^\rho\right) = 1] \\ &- \Pr[\mathsf{Sim}^{\{\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}_i]\}_{i=1}^\rho}\left(1^\lambda, \left\{\left|\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}_i]\right|\right\}_{i=1}^\rho, \{\mathsf{key}_i\}_{i=1}^{j-1}, U_\ell, \{\mathsf{key}_i\}_{i=j+1}^\rho\right) = 1]\Big| \\ &\leq \frac{1}{3p(\lambda)}\end{aligned}$$

Proof. Fix any $u \in \{0,1\}^\ell$, the lemma will follow by averaging over all u. The information about whether the j^{th} key value, denoted V_j, is key_j or u can only be obtain by Sim through the query responses. First, we modify Sim to quit immediately when it gets a response not equal to $\perp$. Such Sim is equally successful at distinguishing between key_j and u since the first non-$\perp$ response tells Sim if its input is equal to key_j. Subsequent responses add nothing to this knowledge. Since Sim can make at most q queries, there are $q+1$ possible values for the view of Sim on a given input. Of those, q views consist of some number of non-$\perp$ responses followed by a $\perp$ response, and one view consists of all q responses equal to $\perp$.
Then by [DRS04, Lemma 2.2b],

$$\begin{aligned}\tilde{H}_\infty(V_j | View(\mathsf{Sim}), \mathsf{aux}) &\geq \tilde{H}_\infty(V_j) - \log(q+1) \\ &\geq \alpha - \log(q+1).\end{aligned}$$

where $\mathsf{aux} = \left(\{|\mathsf{MBCC}[\mathsf{Rep}_{\mathsf{pub}}, k, \mathsf{key}_i]|\}_{i=1}^\rho, \mathsf{key}_1, \cdots, \mathsf{key}_{j-1}, \mathsf{key}_{j+1}, \cdots, \mathsf{key}_\rho\right)$.

Thus, at each query, the probability that Sim gets a non-$\perp$ response and guesses V_j is at most $(q+1)/2^\alpha$. Since there are q queries of Sim, the overall probability is at most $q(q+1)/2^\alpha$. Then since 2^α is negligible in λ, there exists some λ_0 such that for all $\lambda \geq \lambda_0$, $q(q+1)/2^\alpha \leq 1/(3p(\lambda))$. □

Then from Lemma 3, we have

$$\begin{aligned}&\big|\Pr[\mathsf{Sim}^{\{\mathsf{pub}_i\}_{i=1}^{\rho}}(1^\lambda,\{|\mathsf{pub}_i|\}_{i=1}^{\rho},\{\mathsf{key}_i\}_{i=1}^{\rho})=1]\\&-\Pr[\mathsf{Sim}^{\{\mathsf{pub}_i\}_{i=1}^{\rho}}(1^\lambda,\{|\mathsf{pub}_i|\}_{i=1}^{\rho},\mathsf{key}_1,\cdots,\mathsf{key}_{j-1},U_\ell,\mathsf{key}_{j+1},\cdots,\mathsf{key}_\rho)=1]\big|\leq\frac{1}{3p(\lambda)}\end{aligned}\tag{5}$$

Using the triangle inequality on Equations 3, 4 and 5 we obtain

$$\begin{aligned}&\big|\ \Pr[\mathcal{A}(\mathsf{key}_1,\cdots,\mathsf{key}_\rho,\mathsf{pub}_1,\cdots,\mathsf{pub}_\rho)=1]\\&-\Pr[\mathcal{A}(\mathsf{key}_1,\cdots,\mathsf{key}_{i-1},U_\ell,\mathsf{key}_{i+1},\cdots,\mathsf{key}_\rho,\mathsf{pub}_1,\cdots,\mathsf{pub}_\rho)=1]\ \big|\leq\frac{1}{p(\lambda)}\end{aligned}$$

which is a contradiction and completes this proof. □

Composable MBCC obfuscation Wichs and Zirdelis [WZ17] build obfuscation for *multi-bit* compute-and-compare circuits from single bit compute-and-compare by composing the function f with a strongly injective PRG. By doing so they ensure that the target values $(y_1,\cdots,y_\ell)$ are indistinguishable from uniform, even when given f, z and aux. Their proof then relies on the security of the obfuscator for the i^{th} circuit by passing all remaining circuits as auxiliary information.

Unfortunately this technique cannot be directly applied to build *composable* MBCC obfuscation since it requires keeping track of which parts of the PRG output have already been used. This is reasonable for their MBCC obfuscation scheme, where all obfuscated compute-and-compare circuits will be generated at the same time. However this is not practical in the case of composable obfuscation, where the obfuscator will typically be run at different times and without a shared state. One could use a PRG with exponential stretch and select a random part of its output, then the probability of reuse should be low. Another issue is that in Wichs and Zirdelis's scheme, the function and the input to the PRG are always the same. For composability, especially with the goal of building reusable FE, it would need to handle distinct but possibly correlated functions and values. It then is unclear what the auxiliary information (i.e. the other obfuscated programs) may leak on the current obfuscated circuit.