Contention in Cryptoland: Obfuscation, Leakage, UCE

Mihir Bellare[1] Igors Stepanovs[2] Stefano Tessaro[3]

Abstract

This paper addresses the fundamental question of whether or not different, exciting primitives now being considered actually exist. We show that we, unfortunately, cannot have them all. We provide results of the form $\neg\mathbf{A} \vee \neg\mathbf{B}$, meaning one of the primitives $\mathbf{A}, \mathbf{B}$ cannot exist. (But we don't know which.) Specifically, we show that: (1) VGBO (Virtual Grey Box Obfuscation) for all circuits, which has been conjectured to be achieved by candidate constructions, cannot co-exist with Canetti's 1997 AI-DHI (auxiliary input DH inversion) assumption, which has been used to achieve many goals including point-function obfuscation (2) iO (indistinguishability obfuscation) for all circuits cannot co-exist with key-message leakage-resilient symmetric encryption (3) iO cannot co-exist with hash functions that are UCE secure for computationally unpredictable split sources.

Keywords: Impossibility results, Assumptions, Obfuscation, Leakage, UCE.

[1] Department of Computer Science & Engineering, University of California San Diego, 9500 Gilman Drive, La Jolla, California 92093, USA. Email: `mihir@eng.ucsd.edu`. URL: `http://cseweb.ucsd.edu/~mihir/`. Supported in part by NSF grants CNS-1116800 and CNS-1228890.

[2] Department of Computer Science & Engineering, University of California San Diego, 9500 Gilman Drive, La Jolla, California 92093, USA. Email: `istepano@eng.ucsd.edu`. Supported in part by NSF grants CNS-1116800 and CNS-1228890.

[3] Department of Computer Science, University of California Santa Barbara, Santa Barbara, California 93106, USA. Email: `tessaro@cs.ucsb.edu`. URL: `http://www.cs.ucsb.edu/~tessaro/`. Supported in part by NSF grant CNS-1423566.

1 Introduction

Cryptographic theory is being increasingly bold with regard to assumptions and conjectures. This is particularly true in the area of obfuscation, where candidate constructions have been provided whose claim to achieve a certain form of obfuscation is either itself an assumption [28] or is justified under other, new and strong assumptions [38, 9, 31]. This is attractive and exciting because we gain new capabilities and applications. But it behoves us also to be cautious and try to ascertain, not just whether the assumptions are true, but whether the goals are even achievable.

But how are we to determine this? The direct route is cryptanalysis, and we have indeed seen some success [36, 24, 30, 25]. But cryptanalysis can be difficult and runs into major open complexity-theoretic questions. There is another rewarding route, that we pursue here. This is to seek and establish relations that we call *contentions*. These take the form $\neg\mathbf{A} \vee \neg\mathbf{B}$ where $\mathbf{A}, \mathbf{B}$ are different primitives or assumptions. This shows that $\mathbf{A}, \mathbf{B}$ are not *both* achievable, meaning they cannot co-exist. We may not know which of the two fails, but at least one of the two must, which is a valuable and sometimes surprising information. Indeed, many intriguing contentions of this form have been provided in recent work [11, 29, 17, 19, 18, 4, 33]. For example, we know that the following cannot co-exist: "Special-purpose obfuscation" and **diO** [29]; Multi-bit point function obfuscation and **iO** [19]; extractable one-way functions and **iO** [11].

In this paper we begin by addressing the question of whether VGBO (Virtual Grey Box Obfuscation) for all circuits is possible, as conjectured by BCKP [10, Section 1.1]. We show that this is in contention with the AI-DHI assumption of [22, 12]. We go on to show that iO is in contention with certain forms of leakage resilient encryption and UCE. The last answers questions raised in prior work.

1.1 VGBO and AI-DHI

We show that $\neg\mathbf{VGBO} \vee \neg\textbf{AI-DHI}$. That is, Virtual Grey Box Obfuscation (VGBO) of all circuits is in contention with Canetti's 1997 AI-DHI (Auxiliary-Input Diffie-Hellman Inversion) assumption [22, 12]. One of the two (or both) must fail. Let us now back up to provide more information on the objects involved and the proof.

The study of obfuscation began with VBBO (Virtual Black Box Obfuscation) [34, 3], which asks that for any PT adversary $\mathcal{A}$ given the obfuscated circuit, there is a PT simulator $\mathcal{S}$ given an oracle for the original circuit, such that the two have about the same probability of returning 1. The impossibility of VBBO [3, 32, 13] has lead to efforts to define and achieve weaker forms of obfuscation. VGBO [8] is a natural relaxation of VBBO allowing the simulator $\mathcal{S}$ to be computationally unbounded but restricted to polynomially-many oracle queries. This bypasses known VBBO impossibility results while still allowing interesting applications. Furthermore BCKP [9, 10] show that VGBO for NC^1 is achievable (under a novel assumption). They then say "existing candidate indistinguishability obfuscators for all circuits [28, 16, 2] may also be considered as candidates for VGB obfuscation, for all circuits" [10, Section 1.1]. This would mean, in particular, that VGBO for all circuits is achievable. In this paper we ask if this "VGB conjecture" is true.

The AI-DHI assumption [22, 12] says that there is an ensemble $\mathcal{G} = \{\mathbb{G}_\lambda : \lambda \in \mathbb{N}\}$ of prime-order groups such that, for r, s chosen at random from $\mathbb{G}_\lambda$, no polynomial-time adversary can distinguish between (r, r^x) and (r, s), even when given auxiliary information a about x, as long as this information a is "x-prediction-precluding," meaning does not allow one to just compute x in polynomial time. The assumption has been used for oracle hashing [22], point-function obfuscation [12] and zero-knowledge proofs [12].

Our result is that $\neg\mathbf{VGBO} \vee \neg\textbf{AI-DHI}$. That is, either VGBO for all circuits is impossible or the AI-DHI assumption is false. To prove this, we take any ensemble $\mathcal{G} = \{\mathbb{G}_\lambda : \lambda \in \mathbb{N}\}$ of prime-order groups. For random x, we define a way of picking the auxiliary information a such that (1) a is x-prediction-precluding, but (2) there is a polynomial-time adversary that, given a, can distinguish between (r, r^x) and (r, s) for random r, s. Consider the circuit C_x that on input u, v returns 1 if $v = u^x$ and 0 otherwise. The auxiliary information a will be a VGB obfuscation $\overline{\mathrm{C}}$ of C_x. Now (2) is easy to see: the adversary, given challenge (u, v), can win by returning $\overline{\mathrm{C}}(u, v)$. But why is (1) true? We use the assumed VGB security of the obfuscator to reduce (1) to showing that *no, even unbounded*, simulator, given an oracle for C_x, can extract x in a polynomial number of queries. This is shown through an information-theoretic argument that exploits the group structure.

The natural question about which one would be curious is, which of **VGBO** and **AI-DHI** is it that fails? This is an intriguing question and we do not know the answer at this point.

1.2 Key-message leakage resilience

DKL [27] and CKVW [23] provide key leakage resilient symmetric encryption (K-LR-SE) schemes. This means they retain security even when the adversary has auxiliary information about the key, as long as this information is key-prediction-precluding, meaning does not allow one to compute the key. We consider a generalization that we call key-message leakage resilient symmetric encryption (KM-LR-SE). Here the auxiliary information is allowed to depend not just on the key but also on the message, the requirement however still being that it is key-prediction-precluding, meaning one cannot compute the key from the auxiliary information. The enhancement would appear to be innocuous, because the strong semantic-security style formalizations of encryption that we employ in any case allow the adversary to have a priori information about the message. However, we show that this goal is impossible to achieve if iO for all circuits is possible. That is, we show $\neg\mathbf{iO} \vee \neg\mathbf{KM\text{-}LR\text{-}SE}$. Since iO seems to be growing to be more broadly accepted, this indicates that **KM-LR-SE** is not likely to exist. Besides being of interest from the perspective of leakage resilience, this result will be the basis of further results for UCE as we discuss below. The result uses techniques introduced by Brzuska and Mittelback (BM1) [19] to show $\neg\mathbf{iO} \vee \neg\mathbf{MB\text{-}AIPO}$.

1.3 UCE for split sources

UCE is a class of assumptions for function families introduced in BHK [5] with the goal of instantiating random oracles. For a class $\mathbf{S}$ of algorithms called sources, BHK define UCE[$\mathbf{S}$] security of a family of functions. The parameterization is necessary because security is not achievable for the class of all sources. BHK and subsequent work [5, 17, 20, 37, 4, 26] have considered several restricted classes of sources and, based on the assumption of UCE security for these, been able to instantiate random oracles to obtain secure, efficient instantiations for primitives including deterministic public-key encryption, message-locked encryption, encryption secure for key-dependent messages, encryption secure under related-key attacks, adaptive garbling, hardcore functions and IND-CCA public-key encryption.

However UCE here has functioned as an assumption. We know little about its achievability. The basic foundational question in this area is, for which source classes $\mathbf{S}$ is UCE[$\mathbf{S}$] security achievable? The first step towards answering this was taken by BFM [17], who showed that $\neg\mathbf{iO} \vee \neg\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}}]$. That is, iO for all circuits is in contention with UCE security relative to the class $\mathbf{S}^{\mathrm{cup}}$ of all computationally unpredictable sources. (These are sources whose leakage is computationally unpredictable when their queries are answered by a random oracle.) This lead BHK [5] to propose restricting attention to “split” sources. Such sources can leak information about an oracle query and its answer separately, but not together. This circumvents the BFM attack. Indeed, UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$] appeared plausible even in the presence of iO. However in this paper we show $\neg\mathbf{iO} \vee \neg\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$, meaning **iO** and $\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$ security cannot co-exist. The interpretation is that $\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$-secure function families are unlikely to exist.

BM [19] show that UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}} \cap \mathbf{S}^1$] security —$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}} \cap \mathbf{S}^q$ is the class of computationally unpredictable split sources making q oracle queries— is achievable. (They assume iO and AIPO.) Our $\neg\mathbf{iO} \vee \neg\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$ result does not contradict this since our source makes a polynomial number of oracle queries. Indeed our result complements the BM one to show that a bound on the number of source oracle queries is necessary for a positive result. Together these results give a close to complete picture of the achievability of UCE for split sources, the remaining open question being achievability of UCE$[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}} \cap \mathbf{S}^{q}]$ for constant $q > 1$.

We obtain our $\neg\mathbf{iO} \vee \neg\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$ result by showing that $\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}] \Rightarrow \mathbf{KM\text{-}LR\text{-}SE}$, meaning we can build a key-message leakage resilient symmetric encryption scheme given any UCE$[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$-secure function family. But we saw above that $\neg\mathbf{iO} \vee \neg\mathbf{KM\text{-}LR\text{-}SE}$ and can thus conclude that $\neg\mathbf{iO} \vee \neg\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$.

We note that we are not aware of any applications assuming UCE$[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$. Prior applications have used either UCE$[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}} \cap \mathbf{S}^{1}]$ or quite different classes like UCE$[\mathbf{S}^{\mathrm{sup}}]$ —$\mathbf{S}^{\mathrm{sup}}$ is the class of statistically unpredictable sources [5, 17]— and neither of these is at risk from our results. However our $\neg\mathbf{iO} \vee \neg\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$ result is of interest towards understanding the achievability of UCE assumptions and the effectiveness of different kinds of restrictions (in this case, splitting) on sources. The achievability of $\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$ security was an open problem from prior work.

In independent and concurrent work, BM [21] show $\neg\mathbf{iO} \vee \neg$UCE$[\mathbf{S}^{\mathrm{s\text{-}cup}}]$, where $\mathbf{S}^{\mathrm{s\text{-}cup}}$ is the class of *strongly* computationally unpredictable sources as defined in [19]. But the latter show that $\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$ is a *strict* subset of $\mathbf{S}^{\mathrm{s\text{-}cup}}$. This means that our $\neg\mathbf{iO} \vee \neg\mathbf{UCE}[\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}]$ result is strictly stronger than the $\neg\mathbf{iO} \vee \neg$UCE$[\mathbf{S}^{\mathrm{s\text{-}cup}}]$ result of [21]. (Under iO, our result rules out UCE security for a smaller, more restricted class of sources.)

2 Preliminaries

<u>Notation.</u> We denote by $\lambda \in \mathbb{N}$ the security parameter and by 1^λ its unary representation. If $x \in \{0,1\}^*$ is a string then $|x|$ denotes its length, $x[i]$ denotes its i-th bit, and $x[i..j] = x[i] \ldots x[j]$ for $1 \leq i \leq j \leq |x|$. We let ε denote the empty string. If s is an integer then $\mathsf{Pad}_s(\mathrm{C})$ denotes circuit C padded to have size s. We say that circuits $\mathrm{C}_0, \mathrm{C}_1$ are equivalent, written $\mathrm{C}_0 \equiv \mathrm{C}_1$, if they agree on all inputs. If $\mathbf{x}$ is a vector then $|\mathbf{x}|$ denotes the number of its coordinates and $\mathbf{x}[i]$ denotes its i-th coordinate. If X is a finite set, we let $x \leftarrow\$ X$ denote picking an element of X uniformly at random and assigning it to x. Algorithms may be randomized unless otherwise indicated. Running time is worst case. "PT" stands for "polynomial-time," whether for randomized algorithms or deterministic ones. If A is an algorithm, we let $y \leftarrow A(x_1, \ldots; r)$ denote running A with random coins r on inputs $x_1, \ldots$ and assigning the output to y. We let $y \leftarrow\$ A(x_1, \ldots)$ be the result of picking r at random and letting $y \leftarrow A(x_1, \ldots; r)$. We let $[A(x_1, \ldots)]$ denote the set of all possible outputs of A when invoked with inputs $x_1, \ldots$. We say that $f\colon \mathbb{N} \to \mathbb{R}$ is negligible if for every positive polynomial p, there exists $\lambda_p \in \mathbb{N}$ such that $f(\lambda) < 1/p(\lambda)$ for all $\lambda > \lambda_p$. We use the code based game playing framework of [6]. (See Fig. 1 for an example.) By $\mathrm{G}^{\mathcal{A}}(\lambda)$ we denote the event that the execution of game G with adversary $\mathcal{A}$ and security parameter λ results in the game returning true.

<u>Auxiliary information generators.</u> Auxiliary information leaving some quantity computationally unpredictable is common to many of the notions we consider and it is convenient to abstract this out. An *auxiliary information generator* X specifies a PT algorithm $\mathsf{X.Ev}$ that takes 1^λ to return a *target* $k \in \{0,1\}^{\mathsf{X.tl}(\lambda)}$, a *payload* $m \in \{0,1\}^{\mathsf{X.pl}(\lambda)}$ and an *auxiliary information* a, where $\mathsf{X.tl}, \mathsf{X.pl}\colon \mathbb{N} \to \mathbb{N}$ are the target and payload length functions associated to X, respectively. Consider game PRED of Fig. 1 associated to X and a predictor adversary $\mathcal{Q}$. For $\lambda \in \mathbb{N}$ let $\mathsf{Adv}^{\mathsf{pred}}_{\mathsf{X},\mathcal{Q}}(\lambda) = \Pr[\mathrm{PRED}^{\mathcal{Q}}_{\mathsf{X}}(\lambda)]$. We say that X is *unpredictable* if $\mathsf{Adv}^{\mathsf{pred}}_{\mathsf{X},\mathcal{Q}}(\cdot)$ is negligible for every PT adversary $\mathcal{Q}$. We say that X is *uniform* if $\mathsf{X.Ev}(1^\lambda)$ picks the target $k \in \{0,1\}^{\mathsf{X.tl}(\lambda)}$ and the payload $m \in \{0,1\}^{\mathsf{X.pl}(\lambda)}$ uniformly and independently. Note that the auxiliary information a may depend on both the target k and the payload m, but unpredictability refers to recovery of the target k alone.

<u>Obfuscators.</u> An *obfuscator* is a PT algorithm Obf that on input 1^λ and a circuit C returns a circuit

Game $\mathrm{PRED}^{\mathcal{Q}}_{\mathsf{X}}(\lambda)$	Game $\mathrm{VGB1}^{\mathcal{A}}_{\mathsf{Obf},\mathsf{Smp}}(\lambda)$	Game $\mathrm{VGB0}^{\mathcal{S}}_{\mathsf{Smp},q}(\lambda)$	Game $\mathrm{AIDHI}^{\mathcal{A}}_{\mathcal{G},\mathsf{X}}(\lambda)$
$(k, m, a) \leftarrow\$ \mathsf{X.Ev}(1^\lambda)$ $k' \leftarrow\$ \mathcal{Q}(1^\lambda, a)$ Return $(k = k')$	$\mathrm{C} \leftarrow\$ \mathsf{Smp}(1^\lambda)$ $\overline{\mathrm{C}} \leftarrow\$ \mathsf{Obf}(1^\lambda, \mathrm{C})$ $b' \leftarrow\$ \mathcal{A}(1^\lambda, \overline{\mathrm{C}})$ Return $(b' = 1)$	$\mathrm{C} \leftarrow\$ \mathsf{Smp}(1^\lambda)$; $i \leftarrow 0$ $b' \leftarrow\$ \mathcal{S}^{\mathrm{CIRC}}(1^\lambda)$ Return $(b' = 1)$ $\mathrm{CIRC}(x)$ $i \leftarrow i + 1$ If $i > q(\lambda)$ then return $\perp$ $y \leftarrow \mathrm{C}(x)$; Return y	$b \leftarrow\$ \{0, 1\}$; $g \leftarrow\$ \mathbb{G}^*_\lambda$ $(k, \varepsilon, a) \leftarrow\$ \mathsf{X.Ev}(1^\lambda)$ $K_1 \leftarrow g^k$; $K_0 \leftarrow\$ \mathbb{G}_\lambda$ $b' \leftarrow\$ \mathcal{A}(1^\lambda, g, K_b, a)$ Return $(b = b')$

Figure 1: Game defining unpredictabilty of auxiliary information generator X, games defining VGB security of obfuscator Obf and game defining the AI-DHI assumption.

$\overline{\mathrm{C}}$ such that $\overline{\mathrm{C}} \equiv \mathrm{C}$. (That is, $\overline{\mathrm{C}}(x) = \mathrm{C}(x)$ for all x.) We refer to the latter as the *correctness condition.* We will consider various notions of security for obfuscators, including VGBO and iO.

3 VGBO and the AI-DHI assumption

BCKP [9, 10] conjecture that existing candidate constructions of iO also achieve VGBO and thus in particular that VGB obfuscation for all circuits is possible. Here we explore the plausibility of this "VGB conjecture." We show that it implies the failure of Canetti's AI-DHI assumption. Either this assumption is false or VGBO for all circuits is not possible. (In fact, our result refers to an even weaker VGBO assumption.) That is, the long-standing AI-DHI assumption and VGBO are in contention; at most one of these can exist. We start by defining VGBO and recalling the AI-DHI assumption, and then give our result and its proof. Finally we discuss some extensions.

VGBO. Let Obf be an obfuscator as defined in Section 2. We define what it means for it to be a VGB obfuscator. We will use a weak variant of the notion used in some of the literature [8, 9], which strengthens our results since they are negative relations with starting point VGBO.

A *sampler* Smp in this context is an algorithm that takes 1^λ to return a circuit C. Let q be a polynomial, $\mathcal{A}$ an adversary and $\mathcal{S}$ a (not necessarily PT) algorithm called a simulator. For $\lambda \in \mathbb{N}$ let

$$\mathsf{Adv}^{\mathsf{vgb}}_{\mathsf{Obf},\mathsf{Smp},q,\mathcal{A},\mathcal{S}}(\lambda) = \left|\Pr\left[\mathrm{VGB1}^{\mathcal{A}}_{\mathsf{Obf},\mathsf{Smp}}(\lambda)\right] - \Pr\left[\mathrm{VGB0}^{\mathcal{S}}_{\mathsf{Smp},q}(\lambda)\right]\right|$$

where the games are in Fig. 1. Let SAMP be a set of samplers. We say that Obf is a VGB obfuscator for SAMP if for every PT adversary $\mathcal{A}$ there exists a (not necessarily PT) simulator $\mathcal{S}$ and a polynomial q such that $\mathsf{Adv}^{\mathsf{vgb}}_{\mathsf{Obf},\mathsf{Smp},q,\mathcal{A},\mathcal{S}}(\cdot)$ is negligible for all $\mathsf{Smp} \in \mathsf{SAMP}$.

We note that [9] use a VGB variant stronger than the above where the advantage measures the difference in probabilities of $\mathcal{A}$ and $\mathcal{S}$ guessing a predicate $\pi(\mathrm{C})$, rather than just the probabilities of outputting one, which is all we need here. Also note that our VGB definition is trivially achievable whenever $|\mathsf{SAMP}| = 1$, since $\mathcal{S}$ can simulate game $\mathrm{VGB1}^{\mathcal{A}}_{\mathsf{Obf},\mathsf{Smp}}(\lambda)$ for any fixed choice of $\mathcal{A}$ and Smp. Our applications however use a SAMP of size 2.

THE AI-DHI ASSUMPTION. Let $\mathcal{G} = \{\mathbb{G}_\lambda : \lambda \in \mathbb{N}\}$ be an ensemble of groups where for every $\lambda \in \mathbb{N}$ the order $p(\lambda)$ of group $\mathbb{G}_\lambda$ is a prime in the range $2^{\lambda-1} < p(\lambda) < 2^\lambda$. We assume that relevant operations are computable in time polynomial in λ, including computing $p(\cdot)$, testing membership in $\mathbb{G}_\lambda$ and performing operations in $\mathbb{G}_\lambda$. By $\mathbb{G}^*_\lambda$ we denote the non-identity members of the group, which is the set of generators since the group has prime order. An auxiliary information generator X for $\mathcal{G}$ is an auxiliary information generator as per Section 2 with the additional property that the target k returned by $\mathsf{X.Ev}(1^\lambda)$ is in $\mathbb{Z}_{p(\lambda)}$ (i.e. is an exponent) and the payload m is ε (i.e. is effectively absent).

Now consider game AIDHI of Fig. 1 associated to $\mathcal{G}, \mathsf{X}$ and an adversary $\mathcal{A}$. For $\lambda \in \mathbb{N}$ let $\mathsf{Adv}^{\mathsf{aidhi}}_{\mathcal{G},\mathsf{X},\mathcal{A}}(\lambda) = 2\Pr[\text{AIDHI}^{\mathcal{A}}_{\mathcal{G},\mathsf{X}}(\lambda)] - 1$. We say that $\mathcal{G}$ is AI-DHI-secure if $\mathsf{Adv}^{\mathsf{aidhi}}_{\mathcal{G},\mathsf{X},\mathcal{A}}(\cdot)$ is negligible for every unpredictable X for $\mathcal{G}$ and every PT adversary $\mathcal{A}$. The AI-DHI assumption [22, 12] is that there exists a family of groups $\mathcal{G}$ that is AI-DHI secure.

$\neg$VGBO $\vee$ $\neg$AI-DHI. The following says if VGB obfuscation is possible then the AI-DHI assumption is false: there exists *no* family of groups $\mathcal{G}$ that is AI-DHI secure. Our theorem only assumes a very weak form of VGB obfuscation for a class with two samplers (given in the proof).

Theorem 3.1 *Let $\mathcal{G}$ be a family of groups. Then there is a pair $\mathsf{Smp}, \mathsf{Smp}_0$ of PT samplers (defined in the proof) such that if there exists a VGB-secure obfuscator for the class $\mathsf{SAMP} = \{\mathsf{Smp}, \mathsf{Smp}_0\}$, then $\mathcal{G}$ is not AI-DHI-secure.*

Proof of Theorem 3.1: Let Obf be the assumed obfuscator. Let X be the auxiliary information generator for $\mathcal{G}$ defined as follows:

```
Algorithm $\mathsf{X.Ev}(1^\lambda)$
$k \leftarrow\$ \mathbb{Z}_{p(\lambda)}$
$\overline{\mathrm{C}} \leftarrow\$ \mathsf{Obf}(1^\lambda, \mathrm{C}_{1^\lambda,k})$
Return $(k, \varepsilon, \overline{\mathrm{C}})$
```

```
Circuit $\mathrm{C}_{1^\lambda,k}(g, K)$
If ($g \notin \mathbb{G}^*_\lambda$ or $K \notin \mathbb{G}_\lambda$) then return 0
If ($g^k = K$) then return 1
Else return 0
```

The auxiliary information $a = \overline{\mathrm{C}}$ produced by X is an obfuscation of the circuit $\mathrm{C}_{1^\lambda,k}$ shown on the right above. The circuit has 1^λ and the target value k embedded inside. The circuit takes inputs g, K and checks that the first is a group element different from the identity —and thus a generator— and the second is a group element. It then returns 1 if g^k equals K, and 0 otherwise.

We first construct a PT adversary $\mathcal{A}^*$ such that $\mathsf{Adv}^{\mathsf{aidhi}}_{\mathcal{G},\mathsf{X},\mathcal{A}^*}(\cdot)$ is non-negligible. On input $1^\lambda, g, K_b, \overline{\mathrm{C}}$, it simply returns $\overline{\mathrm{C}}(g, K_b)$. That is, it runs the obfuscated circuit $\overline{\mathrm{C}}$ on g and K_b and returns its outcome. If the challenge bit b in game $\text{AIDHI}^{\mathcal{A}^*}_{\mathcal{G},\mathsf{X}}(\lambda)$ is 1 then the adversary always outputs $b' = 1$. Otherwise, the adversary outputs $b' = 1$ with probability $1/p(\lambda)$. We have $\mathsf{Adv}^{\mathsf{aidhi}}_{\mathcal{G},\mathsf{X},\mathcal{A}^*}(\lambda) = 1 - 1/p(\lambda) \geq 1 - 2^{1-\lambda}$, which is not negligible.

We now show that the constructed auxiliary information generator X is unpredictable. In particular, for any PT adversary $\mathcal{Q}$ we construct a PT adversary $\mathcal{A}$ and samplers Smp, Smp_0 such that for all simulators $\mathcal{S}$ and all polynomials q,

$$\mathsf{Adv}^{\mathsf{pred}}_{\mathsf{X},\mathcal{Q}}(\lambda) \leq \mathsf{Adv}^{\mathsf{vgb}}_{\mathsf{Obf},\mathsf{Smp},q,\mathcal{A},\mathcal{S}}(\lambda) + \mathsf{Adv}^{\mathsf{vgb}}_{\mathsf{Obf},\mathsf{Smp}_0,q,\mathcal{A},\mathcal{S}}(\lambda) + \frac{q(\lambda)}{2^{\lambda-1}} \,. \tag{1}$$

Concretely, the adversary $\mathcal{A}$ and the samplers $\mathsf{Smp}, \mathsf{Smp}_0$ operate as follows:

```
Adversary $\mathcal{A}(1^\lambda, \overline{\mathrm{C}})$
$k' \leftarrow\$ \mathcal{Q}(1^\lambda, \overline{\mathrm{C}})$
$\bar{g} \leftarrow\$ \mathbb{G}^*_\lambda$
Return $\overline{\mathrm{C}}(\bar{g}, \bar{g}^{k'})$
```

```
Algorithm $\mathsf{Smp}(1^\lambda)$
$k \leftarrow\$ \mathbb{Z}_{p(\lambda)}$
Return $\mathrm{C}_{1^\lambda,k}$
```

```
Algorithm $\mathsf{Smp}_0(1^\lambda)$
Return $\mathrm{C}_0$
```

Here, in Smp_0, the circuit C_0 takes as input a pair of group elements g, g' from $\mathbb{G}_\lambda$ and always returns 0.

To show Equation (1), we first note that by construction

$$\mathsf{Adv}^{\mathsf{pred}}_{\mathsf{X},\mathcal{Q}}(\lambda) = \Pr\left[\text{VGB1}^{\mathcal{A}}_{\mathsf{Obf},\mathsf{Smp}}(\lambda)\right] \,, \tag{2}$$

because an execution of $\mathrm{PRED}_{\mathsf{X}}^{\mathcal{Q}}(\lambda)$ results in the same output distribution as in $\mathrm{VGB1}_{\mathsf{Obf},\mathsf{Smp}}^{\mathcal{A}}(\lambda)$. The only difference is that in the latter, the check of whether the guess is correct is done via the obfuscated circuit $\overline{\mathrm{C}}$. Now, for all simulators $\mathcal{S}$ and polynomials q, we can rewrite Equation (2) as

$$\begin{aligned}\mathsf{Adv}_{\mathsf{X},\mathcal{Q}}^{\mathsf{pred}}(\lambda) = &\Pr\left[\,\mathrm{VGB1}_{\mathsf{Obf},\mathsf{Smp}}^{\mathcal{A}}(\lambda)\,\right] - \Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp},q}^{\mathcal{S}}(\lambda)\,\right]\\ &+\Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp},q}^{\mathcal{S}}(\lambda)\,\right] - \Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp}_0,q}^{\mathcal{S}}(\lambda)\,\right]\\ &+\Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp}_0,q}^{\mathcal{S}}(\lambda)\,\right] - \Pr\left[\,\mathrm{VGB1}_{\mathsf{Obf},\mathsf{Smp}_0}^{\mathcal{A}}(\lambda)\,\right]\\ &+\Pr\left[\,\mathrm{VGB1}_{\mathsf{Obf},\mathsf{Smp}_0}^{\mathcal{A}}(\lambda)\,\right]\,.\end{aligned}$$

To upper bound $\mathsf{Adv}_{\mathsf{X},\mathcal{Q}}^{\mathsf{pred}}(\lambda)$, we first note that

$$\Pr\left[\,\mathrm{VGB1}_{\mathsf{Obf},\mathsf{Smp}}^{\mathcal{A}}(\lambda)\,\right] - \Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp},q}^{\mathcal{S}}(\lambda)\,\right] \leq \mathsf{Adv}_{\mathsf{Obf},\mathsf{Smp},q,\mathcal{A},\mathcal{S}}^{\mathsf{vgb}}(\lambda)$$

and

$$\Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp}_0,q}^{\mathcal{S}}(\lambda)\,\right] - \Pr\left[\,\mathrm{VGB1}_{\mathsf{Obf},\mathsf{Smp}_0}^{\mathcal{A}}(\lambda)\,\right] \leq \mathsf{Adv}_{\mathsf{Obf},\mathsf{Smp}_0,q,\mathcal{A},\mathcal{S}}^{\mathsf{vgb}}(\lambda)\,.$$

Moreover, we have $\Pr\left[\,\mathrm{VGB1}_{\mathsf{Obf},\mathsf{Smp}_0}^{\mathcal{A}}(\lambda)\,\right] = 0$ by constructon. Namely, adversary $\mathcal{A}$ never outputs 1 in game $\mathrm{VGB1}_{\mathsf{Obf},\mathsf{Smp}_0}^{\mathcal{A}}(\lambda)$, since it is given an obfuscation of the constant zero circuit C_0.

We are left with upper bounding the difference between $\Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp},q}^{\mathcal{S}}(\lambda)\,\right]$ and $\Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp}_0,q}^{\mathcal{S}}(\lambda)\,\right]$. Note that $\mathcal{S}$ is allowed to issue at most $q(\lambda)$ queries to the given circuit, which is either $\mathrm{C}_{1^\lambda,k}$ for a random $k \leftarrow\!\!\$\, \mathbb{Z}_{p(\lambda)}$ or C_0. Denote by Hit the event that $\mathcal{S}$ makes a query (g, K) in $\mathrm{VGB0}_{\mathsf{Smp},q}^{\mathcal{S}}(\lambda)$ such that $g^k = K$. Then, by a standard argument,

$$\Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp},q}^{\mathcal{S}}(\lambda)\,\right] - \Pr\left[\,\mathrm{VGB0}_{\mathsf{Smp}_0,q}^{\mathcal{S}}(\lambda)\,\right] \leq \Pr\left[\,\mathsf{Hit}\,\right]\,.$$

Namely, to compute $\Pr[\,\mathsf{Hit}\,]$, we move from $\mathrm{VGB0}_{\mathsf{Smp},q}^{\mathcal{S}}(\lambda)$ to the simpler $\mathrm{VGB0}_{\mathsf{Smp}_0,q}^{\mathcal{S}}(\lambda)$, where all of $\mathcal{S}$'s queries are answered with 0. We extend the game to sample a random key $k \leftarrow\!\!\$\, \mathbb{Z}_{p(\lambda)}$, and we define Hit' as the event that for one of $\mathcal{S}$'s queries (g, K) we have $g^k = K$. It is not hard to see that $\Pr[\,\mathsf{Hit}'\,]$ and $\Pr[\,\mathsf{Hit}\,]$ are identical, as both games are identical as long as the corresponding events do not occur. Since there are at most $q(\lambda)$ queries, and exactly one k can produce the answer 1 for each query, the union bound yields

$$\Pr[\,\mathsf{Hit}\,] = \Pr[\,\mathsf{Hit}'\,] \leq \frac{q(\lambda)}{p(\lambda)} \leq \frac{q(\lambda)}{2^{\lambda-1}}\,,$$

which concludes the proof. ▌

<u>Extensions.</u> Our technique can be applied to show further contentions between VGBO and other primitives as we discuss briefly here.

In AI-DHI, the group is fixed for each value of the security parameter. One could instead formulate an AI-DHI2 assumption in which there is a family of groups for each value of the security parameter. The AI-DHI2 game would be like the AI-DHI one except that it first picks a group at random from this family to play the role of $\mathbb{G}_\lambda$. This group is provided to $\mathcal{A}$, but not to $\mathsf{X}.\mathsf{Ev}$. This only partially reduces the contention with VGBO. We can show that VGBO and this novel AI-DHI2 assumption are still in contention as long as the groups are verifiable in the sense that one can deterministically in PT check that a given description is a group in the family. This condition is true for most candidate families.

We have noted that one of the important applications of AI-DHI is for AIPO, and indeed the construction of AIPO from AI-DHI is almost a tautology. This leads one to ask whether VGBO and

```
Game $\mathrm{IND}^{\mathcal{A}}_{\mathsf{SE},\mathsf{X}}(\lambda)$
$b \leftarrow_{\$} \{0,1\}$
$(k, m_1, a) \leftarrow_{\$} \mathsf{X.Ev}(1^\lambda)$
$m_0 \leftarrow_{\$} \{0,1\}^{\mathsf{SE.ml}(\lambda)}$
$c \leftarrow_{\$} \mathsf{SE.Enc}(1^\lambda, k, m_b)$
$b' \leftarrow_{\$} \mathcal{A}(1^\lambda, a, c)$
Return $(b = b')$
```

```
Game $\mathrm{DEC}_{\mathsf{SE}}(\lambda)$
$k \leftarrow_{\$} \{0,1\}^{\mathsf{SE.kl}(\lambda)}$
$m \leftarrow_{\$} \{0,1\}^{\mathsf{SE.ml}(\lambda)}$
$c \leftarrow_{\$} \mathsf{SE.Enc}(1^\lambda, k, m)$
$m' \leftarrow \mathsf{SE.Dec}(1^\lambda, k, c)$
Return $(m = m')$
```

Figure 2: Games defining X-KM-leakage resilience of symmetric encryption scheme SE and decryption correctness of symmetric encryption scheme SE.

AIPO are in contention. Again we can show this is the case if the AIPO has a verifiability property under which one can deterministically check whether a circuit is an output of the point obfuscator. But in general VGBO and AIPO do not seem to be in contention.

4 KM-leakage resilient encryption

We refer to a symmetric encryption scheme as K-leakage-resilient if it retains security in the presence of any leakage about the key that leaves the key computationally unpredictable [27]. Such schemes have been designed in [27, 23]. Here, we extend the model by allowing the leakage to depend not just on the key but also on the message, still leaving the key computationally unpredictable. The extension seems innocuous, since the indistinguishability style formalizations used here already capture the adversary having some information about the message. But we show that KM-leakage-resilience is in contention with iO. The interpretation is KM-leakage-resilience is not achievable.

This result has double utility. First it is of direct interest with regard to understanding what is and is not achievable in leakage-resilient cryptography. Second, it is a tool to rule out other primitives assuming iO: to rule out X we can show that X implies KM-leakage-resilient symmetric encryption. We will use this later to rule out UCE for computationally unpredictable split sources assuming iO.

We use standard definitions of indistinguishability obfuscation [3, 28, 39, 15, 1] and pseudorandom generators [14, 40], as recalled in Appendix A. We now start by formalizing KM-leakage resilience.

KM-LEAKAGE RESILIENT ENCRYPTION. A symmetric encryption scheme SE specifies the following. PT encryption algorithm $\mathsf{SE.Enc}$ takes 1^λ, a key $k \in \{0,1\}^{\mathsf{SE.kl}(\lambda)}$ and a message $m \in \{0,1\}^{\mathsf{SE.ml}(\lambda)}$ to return a ciphertext c, where $\mathsf{SE.kl}, \mathsf{SE.ml}\colon \mathbb{N} \to \mathbb{N}$ are the key length and message length functions of SE, respectively. Deterministic PT decryption algorithm $\mathsf{SE.Dec}$ takes $1^\lambda, k, c$ to return a plaintext $m \in \{0,1\}^{\mathsf{SE.ml}(\lambda)}$. Note that there is a key length but no prescribed key-generation algorithm.

For security, let X be an auxiliary information generator with $\mathsf{X.tl} = \mathsf{SE.kl}$ and $\mathsf{X.pl} = \mathsf{SE.ml}$. Consider game $\mathrm{IND}^{\mathcal{A}}_{\mathsf{SE},\mathsf{X}}(\lambda)$ of Fig. 2 associated to SE, X and adversary $\mathcal{A}$. The message m_0 is picked uniformly at random. The adversary $\mathcal{A}$ must determine which message has been encrypted, given not just the ciphertext but auxiliary information a on the key and message m_1. For $\lambda \in \mathbb{N}$ we let $\mathsf{Adv}^{\mathsf{ind}}_{\mathsf{SE},\mathsf{X},\mathcal{A}}(\lambda) = 2\Pr[\mathrm{IND}^{\mathcal{A}}_{\mathsf{SE},\mathsf{X}}(\lambda)] - 1$. We say that SE is X-KM-leakage resilient if the function $\mathsf{Adv}^{\mathsf{ind}}_{\mathsf{SE},\mathsf{X},\mathcal{A}}(\cdot)$ is negligible for all PT adversaries $\mathcal{A}$. This is of course not achievable if a allowed the adversary to compute k, so we restrict attention to unpredictable X. Furthermore, weakening the definition, we restrict attention to uniform X, meaning k and m_1 are uniformly and independently distributed. Thus we say that SE is KM-leakage-resilient if it is X-KM-leakage resilient for all unpredictable, uniform X.

The above requirement is strong in that security is required in the presence of (unpredictable) leakage on the key and first message. But beyond that, in other ways, it has been made weak, because this strengthens our negative results. Namely, we are only requiring security on random messages, not chosen ones, with the key being uniformly distributed, and the key and the two messages all being independently distributed. Furthermore, in contrast to a typical indistinguishability definition, the adversary does not get the messages as input.

The standard correctness condition would ask that $\mathsf{SE.Dec}(1^\lambda, k, \mathsf{SE.Enc}(1^\lambda, k, m)) = m$ for all $k \in \{0,1\}^{\mathsf{SE.kl}(\lambda)}$, all $m \in \{0,1\}^{\mathsf{SE.ml}(\lambda)}$ and all $\lambda \in \mathbb{N}$. We call this perfect correctness. We formulate and use a weaker correctness condition because we can show un-achievability even under this and the weakening is crucial to our applications building KM-leakage-resilient encryption schemes to obtain further impossibility results. Specifically, we require correctness only for random messages and random keys with non-negligible probability. Formally, consider game $\mathrm{DEC}_{\mathsf{SE}}(\lambda)$ of Fig. 2 associated to SE, and for $\lambda \in \mathbb{N}$ let $\mathsf{Adv}^{\mathsf{dec}}_{\mathsf{SE}}(\lambda) = \Pr[\mathrm{DEC}_{\mathsf{SE}}(\lambda)]$ be the decryption correctness function of SE. We require that $\mathsf{Adv}^{\mathsf{dec}}_{\mathsf{SE}}(\cdot)$ be non-negligible.

$\neg$iO $\vee$ $\neg$KM-LR-SE. We present our result saying that KM-leakage-resilient symmetric encryption is not achievable if iO (and PRGs, which can be obtained from one-way functions [35]) exist:

Theorem 4.1 *Let* SE *be a symmetric encryption scheme. Let* Obf *be an indistinguishability obfuscator. Let* R *be a PR-secure PRG with* $\mathsf{R.sl} = \mathsf{SE.ml}$. *Assume that* $2^{-\mathsf{SE.kl}(\lambda)}$ *and* $2^{-\mathsf{R.sl}(\lambda)}$ *are negligible. Then there exists a uniform auxiliary information generator* X *such that the following holds: (1)* X *is unpredictable, but (2)* SE *is not* X*-KM-leakage resilient.*

Proof of Theorem 4.1: The construction and proof extend ideas of [19]. Let a uniform auxiliary information generator X be specified as follows:

```
Algorithm $\mathsf{X.Ev}(1^\lambda)$
$k \leftarrow_\$ \{0,1\}^{\mathsf{SE.kl}(\lambda)}$
$m \leftarrow_\$ \{0,1\}^{\mathsf{SE.ml}(\lambda)}$ ; $y \leftarrow \mathsf{R.Ev}(1^\lambda, m)$
$\mathrm{C} \leftarrow \mathsf{Pad}_{s(\lambda)}(\mathrm{C}_{1^\lambda,k,y})$ ; $\overline{\mathrm{C}} \leftarrow_\$ \mathsf{Obf}(1^\lambda, \mathrm{C})$
Return $(k, m, \overline{\mathrm{C}})$
```

```
Circuit $\mathrm{C}_{1^\lambda,k,y}(c)$
$m \leftarrow \mathsf{SE.Dec}(1^\lambda, k, c)$
$y' \leftarrow \mathsf{R.Ev}(1^\lambda, m)$
If $(y = y')$ then return 1
Else return 0
```

The circuit $\mathrm{C}_{1^\lambda,k,y}$ takes as input a ciphertext c, decrypts it under the embedded key k to get back a $\mathsf{SE.ml}(\lambda)$-bit message m, applies the PRG to m to get a string y', and returns 1 iff y' equals the embedded string y. The auxiliary information generator creates this circuit as shown and outputs its obfuscation.

We define a polynomial s so that $s(\lambda)$ is an upper bound on $\max(|\mathrm{C}^1_{1^\lambda,k,y}|, |\mathrm{C}^2|)$ where the circuits are defined in Fig. 3 and the maximum is over all $k \in \{0,1\}^{\mathsf{SE.kl}(\lambda)}$ and $y \in \{0,1\}^{2 \cdot \mathsf{R.sl}(\lambda)}$. Let us first present an attack proving part (2) of the theorem. Below we define an adversary $\mathcal{A}$ against the X-KM-leakage resilience of SE and an adversary $\mathcal{R}$ against the PR-security of R:

```
Adversary $\mathcal{A}(1^\lambda, \overline{\mathrm{C}}, c)$
$b' \leftarrow \overline{\mathrm{C}}(c)$
Return $b'$
```

```
Adversary $\mathcal{R}(1^\lambda, y)$
$k \leftarrow_\$ \{0,1\}^{\mathsf{SE.kl}(\lambda)}$ ; $m_0 \leftarrow_\$ \{0,1\}^{\mathsf{SE.ml}(\lambda)}$
$c \leftarrow_\$ \mathsf{SE.Enc}(1^\lambda, k, m_0)$ ; $m \leftarrow \mathsf{SE.Dec}(1^\lambda, k, c)$ ; $y' \leftarrow \mathsf{R.Ev}(1^\lambda, m)$
If $(y' = y)$ then $g' \leftarrow 1$ else $g' \leftarrow 0$ ; Return $g'$
```

Adversary $\mathcal{A}$ has input 1^λ, the auxiliary information (leakage) which here is the obfuscated circuit $\overline{\mathrm{C}}$, and a ciphertext c. It simply computes and returns the bit $\overline{\mathrm{C}}(c) = \mathrm{C}_{1^\lambda,k,y}(c)$. For the analysis, consider game $\mathrm{IND}^{\mathcal{A}}_{\mathsf{SE},\mathsf{X}}(\lambda)$ of Fig. 2. If the challenge bit b is 1 and the decryption performed by $\overline{\mathrm{C}}$ is correct then $y' = y$, so

$$\Pr[\, b' = 1 \,|\, b = 1 \,] \geq \mathsf{Adv}^{\mathsf{dec}}_{\mathsf{SE}}(\lambda) \; . \tag{3}$$

```
Games $\mathrm{G}_0$–$\mathrm{G}_3$
$k \leftarrow_{\$} \{0,1\}^{\mathsf{SE.kl}(\lambda)}$ ; $m \leftarrow_{\$} \{0,1\}^{\mathsf{SE.ml}(\lambda)}$
    $y \leftarrow \mathsf{R.Ev}(1^\lambda, m)$ ; $\overline{\mathrm{C}} \leftarrow_{\$} \mathsf{Obf}(1^\lambda, \mathsf{Pad}_{s(\lambda)}(\mathrm{C}^1_{1^\lambda,k,y}))$   // $\mathrm{G}_0$
    $y \leftarrow_{\$} \{0,1\}^{2\cdot\mathsf{R.sl}(\lambda)}$ ; $\overline{\mathrm{C}} \leftarrow_{\$} \mathsf{Obf}(1^\lambda, \mathsf{Pad}_{s(\lambda)}(\mathrm{C}^1_{1^\lambda,k,y}))$   // $\mathrm{G}_1$
    $y \leftarrow_{\$} \{0,1\}^{2\cdot\mathsf{R.sl}(\lambda)}$ ; $\overline{\mathrm{C}} \leftarrow_{\$} \mathsf{Obf}(1^\lambda, \mathsf{Pad}_{s(\lambda)}(\mathrm{C}^2))$   // $\mathrm{G}_2$
$k' \leftarrow_{\$} \mathcal{Q}(1^\lambda, \overline{\mathrm{C}})$ ; Return $(k = k')$

Circuit $\mathrm{C}^1_{1^\lambda,k,y}(c)$
$m \leftarrow \mathsf{SE.Dec}(1^\lambda, k, c)$ ; $y' \leftarrow \mathsf{R.Ev}(1^\lambda, m)$
If $(y = y')$ then return 1 else return 0

Circuit $\mathrm{C}^2(c)$
Return 0
```

Figure 3: **Games for proof of part (1) of Theorem 4.1.**

In the case $b = 0$, the corresponding analysis in [19] for the insecurity of MB-AIPO relied on the fact that PRGs have low collision probability on random seeds. This will not suffice for us because of our weak correctness condition. The latter means that when $b = 0$, we do not know that $\mathsf{SE.Dec}(1^\lambda, k, c)$ equals m_0 and indeed have no guarantees on the distribution of decrypted plaintext message. Instead, we directly exploit the assumed PR-security of the PRG. Thus, consider game $\mathrm{PRG}^{\mathcal{R}}_{\mathsf{R}}(\lambda)$ with adversary $\mathcal{R}$ as above. Letting g denote the challenge bit in the game, we have

$$\begin{aligned}\mathsf{Adv}^{\mathsf{pr}}_{\mathsf{R},\mathcal{R}}(\lambda) &= \Pr[\,g' = 1 \,|\, g = 1\,] - \Pr[\,g' = 1 \,|\, g = 0\,] \\ &\geq \Pr[\,b' = 1 \,|\, b = 0\,] - 2^{-2\cdot\mathsf{R.sl}(\lambda)} \,. \end{aligned} \tag{4}$$

From Equation (4) and Equation (3), we have

$$\begin{aligned}\mathsf{Adv}^{\mathsf{ind}}_{\mathsf{SE},\mathsf{X},\mathcal{A}}(\lambda) &= \Pr[\,b' = 1 \,|\, b = 1\,] - \Pr[\,b' = 1 \,|\, b = 0\,] \\ &\geq \mathsf{Adv}^{\mathsf{dec}}_{\mathsf{SE}}(\lambda) - \mathsf{Adv}^{\mathsf{pr}}_{\mathsf{R},\mathcal{R}}(\lambda) - 2^{-2\cdot\mathsf{R.sl}(\lambda)} \,. \end{aligned} \tag{5}$$

Our weak correctness condition says the first term of Equation (5) is non-negligible. On the other hand, the second and third terms are negligible. This means $\mathsf{Adv}^{\mathsf{ind}}_{\mathsf{SE},\mathsf{X},\mathcal{A}}(\cdot)$ is not negligible, proving claim (2) of Theorem 4.1.

We proceed to prove part (1) of the theorem statement. Let $\mathcal{Q}$ be a PT adversary. Consider the games and associated circuits of Fig. 3. Lines not annotated with comments are common to all three games. Game G_0 is equivalent to $\mathrm{PRED}^{\mathcal{Q}}_{\mathsf{X}}(\lambda)$, so

$$\mathsf{Adv}^{\mathsf{pred}}_{\mathsf{X},\mathcal{Q}}(\lambda) = \Pr[\mathrm{G}_2] + (\Pr[\mathrm{G}_0] - \Pr[\mathrm{G}_1]) + (\Pr[\mathrm{G}_1] - \Pr[\mathrm{G}_2]) \,. \tag{6}$$

We have $\Pr[\mathrm{G}_2] = 2^{-\mathsf{SE.kl}(\lambda)}$, where the latter is assumed to be negligible, because k is uniformly random and the circuit $\overline{\mathrm{C}}$ that is passed to adversary $\mathcal{Q}$ does not depend on k. We now show that $\Pr[\mathrm{G}_i] - \Pr[\mathrm{G}_{i+1}]$ is negligible for $i \in \{0, 1\}$, which by Equation (6) implies that $\mathsf{Adv}^{\mathsf{pred}}_{\mathsf{X},\mathcal{Q}}(\cdot)$ is negligible and hence proves the claim.

First, we construct a PT adversary $\mathcal{R}$ against PRG R, as follows:

```
Adversary $\mathcal{R}(1^\lambda, y)$
$k \leftarrow_{\$} \{0,1\}^{\mathsf{SE.kl}(\lambda)}$ ; $\overline{\mathrm{C}} \leftarrow_{\$} \mathsf{Obf}(1^\lambda, \mathsf{Pad}_{s(\lambda)}(\mathrm{C}^1_{1^\lambda,k,y}))$ ; $k' \leftarrow_{\$} \mathcal{Q}(1^\lambda, \overline{\mathrm{C}})$
If $(k = k')$ then return 1 else return 0
```

We have $\Pr[\mathrm{G}_0] - \Pr[\mathrm{G}_1] = \mathsf{Adv}^{\mathsf{pr}}_{\mathsf{R},\mathcal{R}}(\lambda)$, where the advantage is negligible by the assumed PR-security of R.

Next, we construct a circuit sampler S and an iO-adversary $\mathcal{O}$, as follows:

```
Circuit Sampler $\mathsf{S}(1^\lambda)$
$k \leftarrow\!\$\ \{0,1\}^{\mathsf{SE.kl}(\lambda)}$ ; $y \leftarrow\!\$\ \{0,1\}^{2\cdot\mathsf{R.sl}(\lambda)}$
$\mathrm{C}_1 \leftarrow \mathsf{Pad}_{s(\lambda)}(\mathrm{C}^1_{1^\lambda,k,y})$ ; $\mathrm{C}_0 \leftarrow \mathsf{Pad}_{s(\lambda)}(\mathrm{C}^2)$
$aux \leftarrow k$ ; return $(\mathrm{C}_0, \mathrm{C}_1, aux)$
```

```
Adversary $\mathcal{O}(1^\lambda, \overline{\mathrm{C}}, aux)$
$k \leftarrow aux$ ; $k' \leftarrow\!\$\ \mathcal{Q}(1^\lambda, \overline{\mathrm{C}})$
If $(k = k')$ then return 1
Else return 0
```

It follows that $\Pr[\mathrm{G}_1] - \Pr[\mathrm{G}_2] = \mathsf{Adv}^{\mathsf{io}}_{\mathsf{Obf},\mathsf{S},\mathcal{O}}(\lambda)$. We now show that $\mathsf{S} \in \boldsymbol{S}_{\mathrm{eq}}$, and hence $\mathsf{Adv}^{\mathsf{io}}_{\mathsf{Obf},\mathsf{S},\mathcal{O}}(\lambda)$ is negligible by the assumed iO-security of Obf. Specifically, note that $\mathrm{C}^1_{1^\lambda,k,y}$ and C^2 are not equivalent only if y belongs to the range of R, which contains at most $2^{\mathsf{R.sl}(\lambda)}$ values. However, y is sampled uniformly at random from a set of size $2^{2\cdot\mathsf{R.sl}(\lambda)}$. It follows that

$$\Pr\Big[\ \mathrm{C}_0 \equiv \mathrm{C}_1\ :\ (\mathrm{C}_0, \mathrm{C}_1, aux) \leftarrow\!\$\ \mathsf{S}(1^\lambda)\ \Big] \geq 1 - 2^{-\mathsf{R.sl}(\lambda)},$$

where $2^{-\mathsf{R.sl}(\lambda)}$ is assumed to be negligible, and hence $\mathsf{S} \in \boldsymbol{S}_{\mathrm{eq}}$. ▌

A consequence of Theorem 4.1 is the following.

Corollary 4.2 *Let* SE *be a symmetric encryption scheme such that* $\mathsf{SE.ml}(\cdot) \in \Omega((\cdot)^\epsilon)$ *for some constant* $\epsilon > 0$*. Assume the existence of an indistinguishability obfuscator and a one-way function. Then* SE *is not KM-leakage resilient.*

5 UCE for split sources

BFM [17] showed that UCE[$\mathbf{S}^{\mathrm{cup}}$]-security is not possible if iO exists. We improve this to show that UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$]-security is not possible if iO exists. We obtain this by giving a construction of a KM-leakage-resilient symmetric encryption scheme from UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$] and then invoking our above-mentioned result. Definitions of UCE-secure function families [5] are recalled in Appendix A.

UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$] $\Rightarrow$ KM-LR-SE. We give a construction of a KM-leakage resilient symmetric encryption scheme from a **UCE**[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$] family H, which will allow us to rule out such families under iO. Assume for simplicity that $\mathsf{H.il}$ is odd, and let $\ell = (\mathsf{H.il} - 1)/2$. We call the symmetric encryption scheme $\mathsf{SE} = \mathbf{H\&C}[\mathsf{H}]$ that we associate to H the Hash-and-Check scheme. It is defined as follows. Let $\mathsf{SE.kl}(\lambda) = \mathsf{SE.ml}(\lambda) = \ell(\lambda)$ for all $\lambda \in \mathbb{N}$. Let the encryption and decryption algorithms be as follows:

```
Algorithm $\mathsf{SE.Enc}(1^\lambda, k, m)$
$hk \leftarrow\!\$\ \mathsf{H.Kg}(1^\lambda)$
For $i = 1, \ldots, |m|$ do
    $\mathbf{y}[i] \leftarrow \mathsf{H.Ev}(1^\lambda, hk, k \| m[i] \| \langle i \rangle_{\ell(\lambda)})$
Return $(hk, \mathbf{y})$
```

```
Algorithm $\mathsf{SE.Dec}(1^\lambda, k, (hk, \mathbf{y}))$
For $i = 1, \ldots, |\mathbf{y}|$ do
    If $(\mathsf{H.Ev}(1^\lambda, hk, k \| 1 \| \langle i \rangle_{\ell(\lambda)}) = \mathbf{y}[i])$
    Then $m[i] \leftarrow 1$ else $m[i] \leftarrow 0$
Return $m$
```

Here $\langle i \rangle_{\ell(\lambda)} = 1^i 0^{\ell(\lambda)-i}$ denotes a particular, convenient encoding of integer $i \in \{1, \ldots, \ell(\lambda)\}$ as a string of $\ell(\lambda)$ bits, and $m[i]$ denotes the i-th bit of m. The ciphertext $(hk, \mathbf{y})$ consists of a key hk for H chosen randomly and anew at each encryption, together with the vector $\mathbf{y}$ whose i-th entry is the hash of the i-th message bit along with the key and index i. This scheme will have perfect correctness if H is injective, but we do not want to assume this. The following theorem says that the scheme is KM-leakage resilient and also has (somewhat better than) weak correctness under UCE-security of H.

Theorem 5.1 *Let* H *be a family of functions that is* UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$]*-secure. Assume* $\mathsf{H.il}(\cdot) \in \Omega((\cdot)^\epsilon)$ *for some constant* $\epsilon > 0$ *and* $2^{-\mathsf{H.ol}(\cdot)}$ *is negligible. Let* $\mathsf{SE} = \mathbf{H\&C}[\mathsf{H}]$*. Then (1) symmetric encryption scheme* SE *is KM-leakage resilient, and (2)* $1 - \mathsf{Adv}^{\mathsf{dec}}_{\mathsf{SE}}(\cdot)$ *is negligible.*

Proof of Theorem 5.1: Assuming for simplicity as in the construction that $\mathsf{H.il}$ is odd, let $\ell(\cdot) = (\mathsf{H.il}(\cdot) - 1)/2$. We now prove part (1). Let X be an unpredictable, uniform auxiliary information generator. Let $\mathcal{A}$ be a PT adversary. We build a PT source $\mathcal{S} \in \mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$ and a PT distinguisher $\mathcal{D}$ such that

$$\mathsf{Adv}^{\mathsf{ind}}_{\mathsf{SE},\mathsf{X},\mathcal{A}}(\lambda) \;\le\; 2 \cdot \mathsf{Adv}^{\mathsf{uce}}_{\mathsf{H},\mathcal{S},\mathcal{D}}(\lambda) \tag{7}$$

for all $\lambda \in \mathbb{N}$. The assumption that H is UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$]-secure now implies part (1) of the theorem.

We proceed to build $\mathcal{S}, \mathcal{D}$. We let $\mathcal{S}$ be the split source $\mathcal{S} = \mathsf{Splt}[\mathcal{S}_0, \mathcal{S}_1]$, where algorithms $\mathcal{S}_0, \mathcal{S}_1$ are shown below, along with distinguisher $\mathcal{D}$:

```
Algorithm $\mathcal{S}_0(1^\lambda)$
$(k, m_1, a) \leftarrow\!\!\$\; \mathsf{X.Ev}(1^\lambda)$
$m_0 \leftarrow\!\!\$\; \{0,1\}^{\ell(\lambda)}$ ; $d \leftarrow\!\!\$\; \{0,1\}$
For $i = 1, \ldots, \ell(\lambda)$ do
    $\mathbf{x}[i] \leftarrow k \| m_d[i] \| \langle i \rangle_{\ell(\lambda)}$
Return $((d, a), \mathbf{x})$
```

```
Algorithm $\mathcal{S}_1(1^\lambda, \mathbf{y})$
Return $\mathbf{y}$
```

```
Distinguisher $\mathcal{D}(1^\lambda, hk, L)$
$((d, a), \mathbf{y}) \leftarrow L$ ; $c \leftarrow (hk, \mathbf{y})$
$d' \leftarrow\!\!\$\; \mathcal{A}(1^\lambda, a, c)$
If $(d = d')$ then $b' \leftarrow 1$
Else $b' \leftarrow 0$
Return $b'$
```

Here $\mathcal{S}_0$ calls the auxiliary information generator X to produce a key, a plaintext message and the corresponding auxiliary input. It then picks another plaintext message and the challenge bit d at random, and lets $\mathbf{x}$ consist of the inputs on which the hash function would be applied to create the challenge ciphertext. It leaks the challenge bit and auxiliary information. Algorithm $\mathcal{S}_1$ takes as input the result $\mathbf{y}$ of oracle HASH on $\mathbf{x}$, and leaks the entire vector $\mathbf{y}$. The distinguisher gets the leakage from both stages, together with the key hk. Using the latter, it can create the ciphertext c, which it passes to $\mathcal{A}$ to get back a decision. Its output reflects whether $\mathcal{A}$ wins its game.

Letting b denote the challenge bit in game $\mathrm{UCE}^{\mathcal{S},\mathcal{D}}_{\mathsf{H}}(\lambda)$, we claim that

$$\Pr[\, b' = 1 \,|\, b = 1 \,] = \frac{1}{2} + \frac{1}{2}\mathsf{Adv}^{\mathsf{ind}}_{\mathsf{SE},\mathsf{X},\mathcal{A}}(\lambda) \qquad \text{and} \qquad \Pr[\, b' = 1 \,|\, b = 0 \,] = \frac{1}{2} \;,$$

from which Equation (7) follows. The first equation above should be clear from the construction. For the second, when $b = 0$, we know that HASH is a random oracle. But the entries of $\mathbf{x}$ are all distinct, due to the $\langle i \rangle_{\ell(\lambda)}$ components. So the entries of $\mathbf{y}$ are uniform and independent, and in particular independent of the challenge bit d.

This however does not end the proof: We still need to show that $\mathcal{S} \in \mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$. We have ensured that $\mathcal{S} \in \mathbf{S}^{\mathrm{splt}}$ by construction. The crucial remainig step is to show that $\mathcal{S} \in \mathbf{S}^{\mathrm{cup}}$. This will exploit the assumed unpredictability of X. Let $\mathcal{P}$ be a PT predictor. We build PT adversary $\mathcal{Q}$ such that

$$\mathsf{Adv}^{\mathsf{pred}}_{\mathcal{S},\mathcal{P}}(\lambda) \;\le\; \mathsf{Adv}^{\mathsf{pred}}_{\mathsf{X},\mathcal{Q}}(\lambda) \tag{8}$$

for all $\lambda \in \mathbb{N}$. The assumption that X is unpredictable now implies that $\mathcal{S} \in \mathbf{S}^{\mathrm{cup}}$. The construction of $\mathcal{Q}$ is as follows:

```
Adversary $\mathcal{Q}(1^\lambda, a)$
For $i = 1, \ldots, \ell(\lambda)$ do $\mathbf{y}[i] \leftarrow\!\!\$\; \{0,1\}^{\mathsf{H.ol}(\lambda)}$
$d \leftarrow\!\!\$\; \{0,1\}$ ; $x' \leftarrow\!\!\$\; \mathcal{P}(1^\lambda, ((d, a), \mathbf{y}))$ ; $k \leftarrow x'[1..\ell(\lambda)]$ ; Return $k$
```

Adversary $\mathcal{Q}$ computes leakage $((d, a), \mathbf{y})$ distributed exactly as it would be in game $\mathrm{PRED}^{\mathcal{P}}_{\mathcal{S}}(\lambda)$, where HASH is a random oracle. It then runs $\mathcal{P}$ to get a prediction x' of some oracle query of $\mathcal{S}$. If game $\mathrm{PRED}^{\mathcal{P}}_{\mathcal{S}}(\lambda)$ returns true, then x' must have the form $k \| m_d[i] \| \langle i \rangle_{\ell(\lambda)}$ for some $i \in \{1, \ldots, \ell(\lambda)\}$, where k, d are the key and challenge bit, respectively, chosen by $\mathcal{S}$. Adversary $\mathcal{Q}$ can then win its $\mathrm{PRED}^{\mathcal{Q}}_{\mathsf{X}}(\lambda)$ game by simply returning k, which establishes Equation (8).

This completes the proof of part (1) of the theorem. We prove part (2) by building a PT source $\mathcal{S} \in \mathbf{S}^{\mathrm{sup}} \cap \mathbf{S}^{\mathrm{splt}}$ and a PT distinguisher $\mathcal{D}$ such that

$$1 - \mathsf{Adv}^{\mathsf{dec}}_{\mathsf{SE}}(\lambda) \;\leq\; \mathsf{Adv}^{\mathsf{uce}}_{\mathsf{H},\mathcal{S},\mathcal{D}}(\lambda) + \frac{\ell(\lambda)}{2^{\mathsf{H.ol}(\lambda)}} \tag{9}$$

for all $\lambda \in \mathbb{N}$. But we have assumed that H is UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$]-secure, so it is also UCE[$\mathbf{S}^{\mathrm{sup}} \cap \mathbf{S}^{\mathrm{splt}}$]-secure. We have also assumed $2^{-\mathsf{H.ol}(\cdot)}$ is negligible. Part (2) of the theorem follows.

We proceed to build $\mathcal{S}, \mathcal{D}$. We let $\mathcal{S}$ be the split source $\mathcal{S} = \mathsf{Splt}[\mathcal{S}_0, \mathcal{S}_1]$, where algorithms $\mathcal{S}_0, \mathcal{S}_1$ are shown below, along with distinguisher $\mathcal{D}$:

```
Algorithm $\mathcal{S}_0(1^\lambda)$
$k \leftarrow_{\$} \{0,1\}^{\ell(\lambda)}$
For $i = 1, \ldots, \ell(\lambda)$ do
   $\mathbf{x}[2i-1] \leftarrow k \| 1 \| \langle i \rangle_{\ell(\lambda)}$
   $\mathbf{x}[2i] \leftarrow k \| 0 \| \langle i \rangle_{\ell(\lambda)}$
Return $(\varepsilon, \mathbf{x})$
```

```
Algorithm $\mathcal{S}_1(1^\lambda, \mathbf{y})$
Return $\mathbf{y}$
```

```
Distinguisher $\mathcal{D}(1^\lambda, hk, (\varepsilon, \mathbf{y}))$
$b' \leftarrow 0$
For $i = 1, \ldots, \ell(\lambda)$ do
   If $(\mathbf{y}[2i-1] = \mathbf{y}[2i])$
   Then $b' \leftarrow 1$
Return $b'$
```

Letting b denote the challenge bit in game $\mathrm{UCE}^{\mathcal{S},\mathcal{D}}_{\mathsf{H}}(\lambda)$, we claim that

$$\Pr[\, b' = 1 \,|\, b = 1 \,] \geq 1 - \mathsf{Adv}^{\mathsf{dec}}_{\mathsf{SE}}(\lambda) \qquad \text{and} \qquad \Pr[\, b' = 1 \,|\, b = 0 \,] \leq \frac{\ell(\lambda)}{2^{\mathsf{H.ol}(\lambda)}} \ ,$$

from which Equation (9) follows. The first equation above is true because decryption errors only happen when hash outputs collide for different values of the message bit. For the second, when $b = 0$, we know that HASH is a random oracle. But the entries of $\mathbf{x}$ are all distinct. So the entries of $\mathbf{y}$ are uniform and independent. The chance of a collision of two entries is thus $2^{-\mathsf{H.ol}(\lambda)}$, and the equation then follows from the union bound.

$\mathcal{S}$ is a split source by construction. To conclude the proof we need to show that $\mathcal{S} \in \mathbf{S}^{\mathrm{sup}}$. In the case HASH is a random oracle, the distinctness of the oracle queries of $\mathcal{S}$ means that the entries of $\mathbf{y}$ are uniformly and independently distributed. Since there is no leakage beyond $\mathbf{y}$, the leakage gives the predictor $\mathcal{P}$ no extra information about the entries of $\mathbf{x}$. The uniform choice of k by $\mathcal{S}$ then means that $\mathsf{Adv}^{\mathsf{pred}}_{\mathcal{S},\mathcal{P}}(\cdot) \leq 2^{-\ell(\cdot)}$, even if $\mathcal{P}$ is not restricted to PT. But our assumption on $\mathsf{H.il}(\cdot)$ in the theorem statement implies that $2^{-\ell(\cdot)}$ is negligible. ▌

$\neg$**iO** $\vee$ $\neg$**UCE**[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$]. In the BFM [17] iO-based attack on UCE[$\mathbf{S}^{\mathrm{cup}}$], the source builds a circuit which embeds an oracle query x and its answer y, and outputs an obfuscation of this circuit in the leakage. Splitting is a restriction on sources introduced in BHK [5] with the aim of preventing such attacks. A split source cannot build the BFM circuit because the split structure denies it the ability to leak information that depends both on a query and its answer. Thus, the BFM attack does not work for UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$]. However, we show that in fact UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$]-security is still not achievable assuming iO. This is now a simple corollary of Theorems 4.1 and 5.1 that in particular was the motivation for the latter:

Theorem 5.2 *Let* H *be a family of functions such that* $\mathsf{H.il}(\cdot) \in \Omega((\cdot)^\epsilon)$ *for some constant* $\epsilon > 0$ *and* $2^{-\mathsf{H.ol}(\cdot)}$ *is negligible. Assume the existence of an indistinguishability obfuscator and a one-way function. Then* H *is not* UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$]*-secure.*

BM [20] show that UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}} \cap \mathbf{S}^{1}$]-security is achievable assuming iO and AIPO. Our negative result of Theorem 5.2 does not contradict this, and in fact complements it to give a full picture of the achievability of UCE security for split sources.

References

[1] P. Ananth, D. Boneh, S. Garg, A. Sahai, and M. Zhandry. Differing-inputs obfuscation and applications. Cryptology ePrint Archive, Report 2013/689, 2013. `http://eprint.iacr.org/2013/689`. 7

[2] B. Barak, S. Garg, Y. T. Kalai, O. Paneth, and A. Sahai. Protecting obfuscation against algebraic attacks. In P. Q. Nguyen and E. Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 221–238. Springer, May 2014. 1

[3] B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S. P. Vadhan, and K. Yang. On the (im)possibility of obfuscating programs. In J. Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 1–18. Springer, Aug. 2001. 1, 7, 16

[4] M. Bellare and V. T. Hoang. Resisting randomness subversion: Fast deterministic and hedged public-key encryption in the standard model. In E. Oswald and M. Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 627–656. Springer, Apr. 2015. 1, 2

[5] M. Bellare, V. T. Hoang, and S. Keelveedhi. Instantiating random oracles via UCEs. Cryptology ePrint Archive, Report 2013/424, 2013. Preliminary version in CRYPTO 2013. 2, 3, 10, 12, 15, 16

[6] M. Bellare and P. Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In S. Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 409–426. Springer, May / June 2006. 3

[7] M. Bellare, I. Stepanovs, and S. Tessaro. Poly-many hardcore bits for any one-way function and a framework for differing-inputs obfuscation. In P. Sarkar and T. Iwata, editors, *ASIACRYPT 2014, Part II*, volume 8874 of *LNCS*, pages 102–121. Springer, Dec. 2014. 16

[8] N. Bitansky and R. Canetti. On strong simulation and composable point obfuscation. In T. Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 520–537. Springer, Aug. 2010. 1, 4

[9] N. Bitansky, R. Canetti, Y. T. Kalai, and O. Paneth. On virtual grey box obfuscation for general circuits. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 108–125. Springer, Aug. 2014. 1, 4

[10] N. Bitansky, R. Canetti, Y. T. Kalai, and O. Paneth. On virtual grey box obfuscation for general circuits. Cryptology ePrint Archive, Report 2014/554, 2014. `http://eprint.iacr.org/2014/554`. 1, 4

[11] N. Bitansky, R. Canetti, O. Paneth, and A. Rosen. On the existence of extractable one-way functions. In D. B. Shmoys, editor, *46th ACM STOC*, pages 505–514. ACM Press, May / June 2014. 1

[12] N. Bitansky and O. Paneth. Point obfuscation and 3-round zero-knowledge. In R. Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 190–208. Springer, Mar. 2012. 1, 5

[13] N. Bitansky and O. Paneth. On the impossibility of approximate obfuscation and applications to resettable cryptography. In D. Boneh, T. Roughgarden, and J. Feigenbaum, editors, *45th ACM STOC*, pages 241–250. ACM Press, June 2013. 1

[14] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudorandom bits. *SIAM Journal on Computing*, 13(4):850–864, 1984. 7, 16

[15] E. Boyle, K.-M. Chung, and R. Pass. On extractability obfuscation. In Y. Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 52–73. Springer, Feb. 2014. 7

[16] Z. Brakerski and G. N. Rothblum. Virtual black-box obfuscation for all circuits via generic graded encoding. In Y. Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 1–25. Springer, Feb. 2014. 1

[17] C. Brzuska, P. Farshim, and A. Mittelbach. Indistinguishability obfuscation and UCEs: The case of computationally unpredictable sources. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 188–205. Springer, Aug. 2014. 1, 2, 3, 10, 12, 16

[18] C. Brzuska, P. Farshim, and A. Mittelbach. Random-oracle uninstantiability from indistinguishability obfuscation. In Y. Dodis and J. B. Nielsen, editors, *TCC 2015, Part II*, volume 9015 of *LNCS*, pages 428–455. Springer, Mar. 2015. 1

[19] C. Brzuska and A. Mittelbach. Indistinguishability obfuscation versus multi-bit point obfuscation with auxiliary input. In P. Sarkar and T. Iwata, editors, *ASIACRYPT 2014, Part II*, volume 8874 of *LNCS*, pages 142–161. Springer, Dec. 2014. 1, 2, 3, 8, 9

[20] C. Brzuska and A. Mittelbach. Using indistinguishability obfuscation via UCEs. In P. Sarkar and T. Iwata, editors, *ASIACRYPT 2014, Part II*, volume 8874 of *LNCS*, pages 122–141. Springer, Dec. 2014. 2, 13, 16

[21] C. Brzuska and A. Mittelbach. Universal computational extractors and the superfluous padding assumption for indistinguishability obfuscation. Cryptology ePrint Archive, Report 2015/581, 2015. `http://eprint.iacr.org/2015/581`. 3

[22] R. Canetti. Towards realizing random oracles: Hash functions that hide all partial information. In B. S. Kaliski Jr., editor, *CRYPTO'97*, volume 1294 of *LNCS*, pages 455–469. Springer, Aug. 1997. 1, 5

[23] R. Canetti, Y. T. Kalai, M. Varia, and D. Wichs. On symmetric encryption and point obfuscation. In D. Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 52–71. Springer, Feb. 2010. 2, 7

[24] J. H. Cheon, K. Han, C. Lee, H. Ryu, and D. Stehlé. Cryptanalysis of the multilinear map over the integers. Cryptology ePrint Archive, Report 2014/906, 2014. `http://eprint.iacr.org/2014/906`. 1

[25] J.-S. Coron, T. Lepoint, and M. Tibouchi. Cryptanalysis of two candidate fixes of multilinear maps over the integers. Cryptology ePrint Archive, Report 2014/975, 2014. `http://eprint.iacr.org/2014/975`. 1

[26] Y. Dodis, C. Ganesh, A. Golovnev, A. Juels, and T. Ristenpart. A formal treatment of backdoored pseudorandom generators. In E. Oswald and M. Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 101–126. Springer, Apr. 2015. 2

[27] Y. Dodis, Y. T. Kalai, and S. Lovett. On cryptography with auxiliary input. In M. Mitzenmacher, editor, *41st ACM STOC*, pages 621–630. ACM Press, May / June 2009. 2, 7

[28] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th FOCS*, pages 40–49. IEEE Computer Society Press, Oct. 2013. 1, 7, 16

[29] S. Garg, C. Gentry, S. Halevi, and D. Wichs. On the implausibility of differing-inputs obfuscation and extractable witness encryption with auxiliary input. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 518–535. Springer, Aug. 2014. 1

[30] C. Gentry, S. Halevi, H. K. Maji, and A. Sahai. Zeroizing without zeroes: Cryptanalyzing multilinear maps without encodings of zero. Cryptology ePrint Archive, Report 2014/929, 2014. `http://eprint.iacr.org/2014/929`. 1

[31] C. Gentry, A. Lewko, A. Sahai, and B. Waters. Indistinguishability obfuscation from the multilinear subgroup elimination assumption. Cryptology ePrint Archive, Report 2014/309, 2014. `http://eprint.iacr.org/2014/309`. 1

[32] S. Goldwasser and Y. T. Kalai. On the impossibility of obfuscation with auxiliary input. In *46th FOCS*, pages 553–562. IEEE Computer Society Press, Oct. 2005. 1

[33] M. D. Green, J. Katz, A. J. Malozemoff, and H.-S. Zhou. A unified approach to idealized model separations via indistinguishability obfuscation. Cryptology ePrint Archive, Report 2014/863, 2014. `http://eprint.iacr.org/2014/863`. 1

[34] S. Hada. Zero-knowledge and code obfuscation. In T. Okamoto, editor, *ASIACRYPT 2000*, volume 1976 of *LNCS*, pages 443–457. Springer, Dec. 2000. 1

[35] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999. 8

[36] H. T. Lee and J. H. Seo. Security analysis of multilinear maps over the integers. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 224–240. Springer, Aug. 2014. 1

[37] T. Matsuda and G. Hanaoka. Chosen ciphertext security via UCE. In H. Krawczyk, editor, *PKC 2014*, volume 8383 of *LNCS*, pages 56–76. Springer, Mar. 2014. 2

```
Game $\mathrm{UCE}_{\mathsf{H}}^{\mathcal{S},\mathcal{D}}(\lambda)$
$b \leftarrow\$ \{0,1\}$ ; $hk \leftarrow\$ \mathsf{H.Kg}(1^\lambda)$
$L \leftarrow\$ \mathcal{S}^{\mathrm{HASH}}(1^\lambda)$
$b' \leftarrow\$ \mathcal{D}(1^\lambda, hk, L)$
Return $(b' = b)$

$\mathrm{HASH}(x)$
If $T[x] = \bot$ then
    If $b = 0$ then $T[x] \leftarrow\$ \{0,1\}^{\mathsf{H.ol}(\lambda)}$
    Else $T[x] \leftarrow \mathsf{H.Ev}(1^\lambda, hk, x)$
Return $T[x]$
```

```
Game $\mathrm{PRED}_{\mathcal{S}}^{\mathcal{P}}(\lambda)$
$X \leftarrow \emptyset$
$L \leftarrow\$ \mathcal{S}^{\mathrm{HASH}}(1^\lambda)$
$x' \leftarrow\$ \mathcal{P}(1^\lambda, L)$
Return $(x' \in X)$

$\mathrm{HASH}(x)$
If $T[x] = \bot$ then
    $T[x] \leftarrow\$ \{0,1\}^{\mathsf{H.ol}(\lambda)}$
$X \leftarrow X \cup \{x\}$
Return $T[x]$
```

```
Source $\mathcal{S}^{\mathrm{HASH}}(1^\lambda)$
$(L_0, \mathbf{x}) \leftarrow\$ \mathcal{S}_0(1^\lambda)$
For $i = 1, \ldots, |\mathbf{x}|$ do
    $\mathbf{y}[i] \leftarrow\$ \mathrm{HASH}(\mathbf{x}[i])$
$L_1 \leftarrow\$ \mathcal{S}_1(1^\lambda, \mathbf{y})$
$L \leftarrow (L_0, L_1)$
Return $L$
```

Figure 4: Games defining UCE security of function family H, unpredictability of source $\mathcal{S}$, and the split source $\mathcal{S} = \mathsf{Splt}[\mathcal{S}_0, \mathcal{S}_1]$ associated to $\mathcal{S}_0, \mathcal{S}_1$.

[38] R. Pass, K. Seth, and S. Telang. Indistinguishability obfuscation from semantically-secure multilinear encodings. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 500–517. Springer, Aug. 2014. 1

[39] A. Sahai and B. Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In D. B. Shmoys, editor, *46th ACM STOC*, pages 475–484. ACM Press, May / June 2014. 7, 16

[40] A. C.-C. Yao. Theory and applications of trapdoor functions (extended abstract). In *23rd FOCS*, pages 80–91. IEEE Computer Society Press, Nov. 1982. 7, 16

A Definitions

We recall various relevant definitons.

Function families. A family of functions F specifies the following. PT key generation algorithm $\mathsf{F.Kg}$ takes 1^λ to return a key $fk \in \{0,1\}^{\mathsf{F.kl}(\lambda)}$, where $\mathsf{F.kl}\colon \mathbb{N} \to \mathbb{N}$ is the key length function associated to F. Deterministic, PT evaluation algorithm $\mathsf{F.Ev}$ takes 1^λ, key $fk \in [\mathsf{F.Kg}(1^\lambda)]$ and an input $x \in \{0,1\}^{\mathsf{F.il}(\lambda)}$ to return an output $\mathsf{F.Ev}(1^\lambda, fk, x) \in \{0,1\}^{\mathsf{F.ol}(\lambda)}$, where $\mathsf{F.il}, \mathsf{F.ol}\colon \mathbb{N} \to \mathbb{N}$ are the input and output length functions associated to F, respectively. We say that F is *injective* if the function $\mathsf{F.Ev}(1^\lambda, fk, \cdot)\colon \{0,1\}^{\mathsf{F.il}(\lambda)} \to \{0,1\}^{\mathsf{F.ol}(\lambda)}$ is injective for every $\lambda \in \mathbb{N}$ and every $fk \in [\mathsf{F.Kg}(1^\lambda)]$.

UCE security. We recall the Universal Computational Extractor (UCE) framework of BHK [5]. Let H be a family of functions. Let $\mathcal{S}$ be an adversary called the *source* and $\mathcal{D}$ an adversary called the *distinguisher*. We associate to them and H the game $\mathrm{UCE}_{\mathsf{H}}^{\mathcal{S},\mathcal{D}}(\lambda)$ in the left panel of Fig. 4. The source has access to an oracle HASH and we require that any query x made to this oracle have length $\mathsf{H.il}(\lambda)$. When the challenge bit b is 1 (the "real" case) the oracle responds via $\mathsf{H.Ev}$ under a key hk that is chosen by the game and *not* given to the source. When $b = 0$ (the "random" case) it responds as a random oracle. The source then leaks a string L to its accomplice distinguisher. The latter *does* get the key hk as input and must now return its guess $b' \in \{0,1\}$ for b. The game returns true iff $b' = b$, and the uce-advantage of $(\mathcal{S}, \mathcal{D})$ is defined for $\lambda \in \mathbb{N}$ via $\mathsf{Adv}^{\mathsf{uce}}_{\mathsf{H},\mathcal{S},\mathcal{D}}(\lambda) = 2\Pr[\mathrm{UCE}_{\mathsf{H}}^{\mathcal{S},\mathcal{D}}(\lambda)] - 1$. If $\mathbf{S}$ is a class (set) of sources, we say that H is UCE[$\mathbf{S}$]-secure if $\mathsf{Adv}^{\mathsf{uce}}_{\mathsf{H},\mathcal{S},\mathcal{D}}(\cdot)$ is negligible for all sources $\mathcal{S} \in \mathbf{S}$ and all PT distinguishers $\mathcal{D}$.

It is easy to see that UCE[$\mathbf{S}$]-security is not achievable if $\mathbf{S}$ is the class of all PT sources [5]. To obtain meaningful notions of security, BHK [5] impose restrictions on the source. A central restriction is unpredictability. A source is unpredictable if it is hard to guess the source's HASH queries even given the leakage, in the *random case* of the UCE game. Formally, let $\mathcal{S}$ be a source and $\mathcal{P}$ an adversary

```
Game $\mathrm{IO}^{\mathcal{O}}_{\mathsf{Obf},\mathsf{S}}(\lambda)$
$b \leftarrow\!\!\$ \{0,1\}$
$(\mathrm{C}_0, \mathrm{C}_1, aux) \leftarrow\!\!\$ \mathsf{S}(1^\lambda)$
$\overline{\mathrm{C}} \leftarrow\!\!\$ \mathsf{Obf}(1^\lambda, \mathrm{C}_b)$
$b' \leftarrow\!\!\$ \mathcal{O}(1^\lambda, \overline{\mathrm{C}}, aux)$
Return $(b = b')$
```

```
Game $\mathrm{PRG}^{\mathcal{R}}_{\mathsf{R}}(\lambda)$
$b \leftarrow\!\!\$ \{0,1\}$
$m \leftarrow\!\!\$ \{0,1\}^{\mathsf{R.sl}(\lambda)}$
$y_1 \leftarrow \mathsf{R.Ev}(1^\lambda, m)$
$y_0 \leftarrow\!\!\$ \{0,1\}^{2\cdot\mathsf{R.sl}(\lambda)}$
$b' \leftarrow\!\!\$ \mathcal{R}(1^\lambda, y_b)$
Return $(b = b')$
```

Figure 5: Games defining iO-security of obfuscator Obf relative to circuit sampler S and PR-security of pseudorandom generator R.

called a predictor and consider game $\mathrm{PRED}^{\mathcal{P}}_{\mathcal{S}}(\lambda)$ in the middle panel of Fig. 4. For $\lambda \in \mathbb{N}$ we let $\mathsf{Adv}^{\mathsf{pred}}_{\mathcal{S},\mathcal{P}}(\lambda) = \Pr[\mathrm{PRED}^{\mathcal{P}}_{\mathcal{S}}(\lambda)]$. We say that $\mathcal{S}$ is computationally unpredictable if $\mathsf{Adv}^{\mathsf{pred}}_{\mathcal{S},\mathcal{P}}(\cdot)$ is negligible for all PT predictors $\mathcal{P}$, and let $\mathbf{S}^{\mathrm{cup}}$ be the class of all PT computationally unpredictable sources. We say that $\mathcal{S}$ is statistically unpredictable if $\mathsf{Adv}^{\mathsf{pred}}_{\mathcal{S},\mathcal{P}}(\cdot)$ is negligible for all (not necessarily PT) predictors $\mathcal{P}$, and let $\mathbf{S}^{\mathrm{sup}} \subseteq \mathbf{S}^{\mathrm{cup}}$ be the class of all PT statistically unpredictable sources.

BFM [17] show that UCE[$\mathbf{S}^{\mathrm{cup}}$]-security is not achievable assuming that indistinguishability obfuscation is possible. This has lead applications to either be based on UCE[$\mathbf{S}^{\mathrm{sup}}$] or on subsets of UCE[$\mathbf{S}^{\mathrm{cup}}$], meaning to impose further restrictions on the source. UCE[$\mathbf{S}^{\mathrm{sup}}$], introduced in [5, 17], seems at this point to be a viable assumption. In order to restrict the computational case, one can consider split sources as defined in BHK [5]. Let $\mathcal{S}_0, \mathcal{S}_1$ be algorithms, neither of which have access to any oracles. The *split source* $\mathcal{S} = \mathsf{Splt}[\mathcal{S}_0, \mathcal{S}_1]$ associated to $\mathcal{S}_0, \mathcal{S}_1$ is defined in the right panel of Fig. 4. Algorithm $\mathcal{S}_0$ returns a pair $(L_0, \mathbf{x})$. Here $\mathbf{x}$ is a vector over $\{0,1\}^{\mathsf{H.il}(\lambda)}$ all of whose entries are required to be distinct. The first adversary creates the oracle queries for the source $\mathcal{S}$, the latter making these queries and passing the replies to the second adversary to get the leakage. In this way, neither $\mathcal{S}_0$ nor $\mathcal{S}_1$ have an input-output pair from the oracle, limiting their ability to create leakage useful to the distinguisher. A source $\mathcal{S}$ is said to belong to the class $\mathbf{S}^{\mathrm{splt}}$ if there exist PT $\mathcal{S}_0, \mathcal{S}_1$ such that $\mathcal{S} = \mathsf{Splt}[\mathcal{S}_0, \mathcal{S}_1]$, meaning is defined as above. The class of interest is now UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}}$], meaning UCE-security for computationally unpredictable, split sources.

Another way to restrict a UCE source is by limiting the number of queries it can make. Let $\mathbf{S}^q$ be the class of sources making $q(\cdot)$ oracle queries. This allows to consider $\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}} \cap \mathbf{S}^1$, a class of computationally unpredictable split sources that make a single query. BM [20] show that UCE[$\mathbf{S}^{\mathrm{cup}} \cap \mathbf{S}^{\mathrm{splt}} \cap \mathbf{S}^1$]-security is achievable assuming iO and AIPO.

Indistinguishability obfuscation. We use the definitional framework of BST [7] which parameterizes security via classes of circuit samplers. Let Obf be an obfuscator. A *sampler* in this context is a PT algorithm S that on input 1^λ returns a triple $(\mathrm{C}_0, \mathrm{C}_1, aux)$ where $\mathrm{C}_0, \mathrm{C}_1$ are circuits of the same size, number of inputs and number of outputs, and aux is a string. If $\mathcal{O}$ is an adversary and $\lambda \in \mathbb{N}$ we let $\mathsf{Adv}^{\mathsf{io}}_{\mathsf{Obf},\mathsf{S},\mathcal{O}}(\lambda) = 2\Pr[\mathrm{IO}^{\mathcal{O}}_{\mathsf{Obf},\mathsf{S}}(\lambda)] - 1$ where game $\mathrm{IO}^{\mathcal{O}}_{\mathsf{Obf},\mathsf{S}}(\lambda)$ is defined in Fig. 5. Now let $\boldsymbol{S}$ be a class (set) of circuit samplers. We say that Obf is $\boldsymbol{S}$-secure if $\mathsf{Adv}^{\mathsf{io}}_{\mathsf{Obf},\mathsf{S},\mathcal{O}}(\cdot)$ is negligible for every PT adversary $\mathcal{O}$ and every circuit sampler $\mathsf{S} \in \boldsymbol{S}$. We say that circuit sampler S produces equivalent circuits if there exists a negligible function ν such that $\Pr\left[\,\mathrm{C}_0 \equiv \mathrm{C}_1 \;:\; (\mathrm{C}_0, \mathrm{C}_1, aux) \leftarrow\!\!\$ \mathsf{S}(1^\lambda)\,\right] \geq 1 - \nu(\lambda)$ for all $\lambda \in \mathbb{N}$. Let $\boldsymbol{S}_{\mathrm{eq}}$ be the class of all circuit samplers that produce equivalent circuits. We say that Obf is an indistinguishability obfuscator if it is $\boldsymbol{S}_{\mathrm{eq}}$-secure [3, 28, 39].

PRGs. A pseudorandom generator R [14, 40] specifies a deterministic PT algorithm $\mathsf{R.Ev}$ where $\mathsf{R.sl}\colon \mathbb{N} \to \mathbb{N}$ is the seed length function of R such that $\mathsf{R.Ev}(1^\lambda, \cdot)\colon \{0,1\}^{\mathsf{R.sl}(\lambda)} \to \{0,1\}^{2\cdot\mathsf{R.sl}(\lambda)}$ for all $\lambda \in \mathbb{N}$. We say that R is PR-secure if the function $\mathsf{Adv}^{\mathsf{pr}}_{\mathsf{R},\mathcal{R}}(\cdot)$ is negligible for every PT adversary $\mathcal{R}$, where for $\lambda \in \mathbb{N}$ we let $\mathsf{Adv}^{\mathsf{pr}}_{\mathsf{R},\mathcal{R}}(\lambda) = 2\Pr[\mathrm{PRG}^{\mathcal{R}}_{\mathsf{R}}(\lambda)] - 1$ and game PRG is specified in Fig. 5.