Functional Encryption: New Perspectives and Lower Bounds

Shweta Agrawal * Sergey Gorbunov† Vinod Vaikuntanathan‡ Hoeteck Wee§

Abstract

Functional encryption is an emerging paradigm for public-key encryption that enables fine-grained control of access to encrypted data. In this work, we present new perspectives on security definitions for functional encryption, as well as new lower bounds on what can be achieved. Our main contributions are as follows:

- We show a lower bound for functional encryption that satisfies a weak (non-adaptive) simulation-based security notion, via pseudo-random functions. This is the *first* lower bound that exploits *unbounded* collusions in an essential way.
- We put forth and discuss a simulation-based notion of security for functional encryption, with an unbounded simulator (called USIM). We show that this notion interpolates indistinguishability and simulation-based security notions, and has strong correlations to results and barriers in the zero-knowledge and multi-party computation literature.

Keywords: Functional encryption, Lower Bounds, Pseudo-Random Functions, Simulation-based Definitions.

*UCLA. Email: shweta@cs.ucla.edu. Research supported in part from a DARPA/ONR PROCEED award, NSF grants 1136174, 1118096, 1065276, 0916574 and 0830803, a Xerox Faculty Research Award, a Google Faculty Research Award, an equipment grant from Intel, and an Okawa Foundation Research Grant (Amit Sahai). This material is based upon work supported by the Defense Advanced Research Projects Agency through the U.S. Office of Naval Research under Contract N00014-11-1-0389. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense, the National Science Foundation, or the U.S. Government.

†University of Toronto. Email: sgorbunov@cs.toronto.edu. Supported by NSERC Alexander Graham Bell Graduate Scholarship.

‡University of Toronto. Email: vinodv@cs.toronto.edu. Supported by an NSERC Discovery Grant and by DARPA under Agreement number FA8750-11-2-0225. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

§George Washington University. Email: hoeteck@alum.mit.edu. Supported by NSF CAREER Award CNS-1237429.

1 Introduction

Functional encryption [SW05, SW] is a new paradigm for public-key encryption that enables fine-grained control of access to encrypted data. It extends several previous notions, most notably identity-based encryption [Sha84, BF01, Coc01], and provides, for instance, the ability to generate and release secret keys associated with a keyword that can decrypt only those documents that contain the keyword. More generally, functional encryption allows the owner of a "master" secret key to release restricted secret keys that reveal a specific function of encrypted data. This stands in stark contrast to traditional encryption, where access to the encrypted data is all or nothing: namely, given the secret key, one can decrypt and read the entire plaintext, but without it, nothing about the plaintext is revealed at all (other than its length).

Functional Encryption. A functional encryption scheme for a circuit family [BSW11, O'N10] $\mathcal{C}$, associates secret keys SK_C with every circuit $C \in \mathcal{C}$ and ciphertext CT with every input message x. [1]

In broad terms, functional encryption requires that the owner of a secret key SK_C and a ciphertext CT (corresponding to an input message x) be able to compute $C(x)$, but learn nothing else about x itself. (Typically, and throughout this work, we assume that the circuit family $\mathcal{C}$ as well as the circuit queries C are public.)

Moreover, security should hold in the presence of collusions amongst "key holders", that is, malicious users should not be able to combine their secret keys to learn unauthorized information. More formally, a collusion of users that hold secret keys $\mathsf{SK}_{C_1}, \ldots, \mathsf{SK}_{C_q}$ and an encryption of x should learn nothing else about x apart from $C_1(x), \ldots, C_q(x)$, for any polynomial q.

An important subclass of functional encryption is that of public-index predicate encryption. Here, the input x is a pair (ind, μ) where ind is an index and μ the payload message. Let P be a Boolean predicate defined on indices, the circuit family $\mathcal{C}$ is given by:

$$C_P(\mathsf{ind}, \mu) = \begin{cases} (\mathsf{ind}, \mu) & \text{if } P(\mathsf{ind}) = 1 \\ (\mathsf{ind}, \perp) & \text{otherwise} \end{cases}$$

Even though public index predicate encryption seems like a weak object, it already captures identity-based encryption, and is also very useful in constructing protocols for verifiably delegating computation as shown recently by Parno, Raykova and Vaikuntanathan [PRV12].

Predicate encryption captures and generalizes a large number of previous constructions, including identity-based encryption (IBE) [Sha84, BF01, Coc01, BW06], fuzzy IBE [SW05, ABV+12], attribute-based encryption (ABE) [GPSW06, LOS+10], and inner product encryption [KSW08, LOS+10, AFV11]. Specifically, IBE corresponds to P encoding a point function. Moreover, essentially all known constructions are examples of public-index predicate encryption schemes or its variants, with a few exceptions – constructions in [BF01, BW06, KSW08] achieve a stronger private-index security notion in which the index ind also remains hidden from the adversary.

Recent Work. Boneh, Sahai and Waters [BSW11] and O'Neill [O'N10] were the first to put forth a general definitional framework for functional encryption. They considered two security notions for functional encryption, namely: *indistinguishability* (IND) based security and *simulation* (SIM) based security. The former stipulates that it is infeasible to distinguish encryptions of any two

[1] An alternative approach is associate secret keys to inputs and ciphertexts to circuits. This is equivalent to our approach by taking a new "universal" family U_x that on input C outputs $C(x)$.

	realizable for public-index	realizable for all circuits
xx-yy-IND	open	open
xx-yy-SIM (xx = 1 OR yy = NA)	open	**no** (Section 4)
many-AD-SIM	no [BSW11]	no ←
xx-yy-USIM (xx = 1 OR yy = NA)	open	open
many-AD-USIM	**no** [BSW11] ♯	**no** ←

Figure 1: Summary of results and open problems. Results from this work are marked with boldface. Results implicit in previous works are marked with ♯. Results that are trivially implied by results in a previous column are marked with ←. The second and third columns indicate whether the definition is realizable for all public-index predicate encryption schemes (e.g. IBE) and for all circuits respectively. USIM refers to the notion of unbounded simulation discussed in Section 1.2.

messages, without getting a secret key that decrypts the ciphertexts to distinct values; the latter stipulates the existence of an efficient simulator that given $C_1(x), \ldots, C_q(x)$, outputs the view of the colluders that are given an encryption of x as well as secret keys $\mathsf{SK}_{C_1}, \ldots, \mathsf{SK}_{C_q}$.

Both of these notions may be further refined in two ways: *adaptive* (AD) versus *non-adaptive* (NA) which capture whether the adversary's queries to the key derivation oracle may or may not depend on the challenge ciphertext; and *one* versus *many*, referring to whether the adversary receives a single or multiple challenge ciphertexts. Together, these give rise to eight security notions xx-yy-zzz, where xx $\in \{1, \text{many}\}$, yy $\in \{\mathsf{NA}, \mathsf{AD}\}$, and zzz $\in \{\mathsf{IND}, \mathsf{SIM}\}$.

We note that in general, indistinguishability based security provides a weaker guarantee than simulation based security (that is, xx-yy-SIM implies xx-yy-IND and xx-yy-IND does not imply xx-yy-SIM in general); on the other hand, we have that 1-yy-IND implies many-yy-IND. Boneh, et al. [BSW11] pointed out that indistinguishability based security is vacuous and inadequate for certain circuit families, which indicate that we should opt for simulation-based security whenever possible.[2] O'Neill [O'N10] showed that NA-IND and NA-SIM are equivalent for some subclass of circuit families that are roughly speaking, "easy to invert".

All prior positive results achieve many-AD-IND security or relaxations there-of.[3] The only known impossibility result we have for general functional encryption is that of Boneh et al. [BSW11] for realizing the IBE functionality under many-AD-SIM security. In particular, in light of known results, it is entirely conceivable that we can realize functional encryption for all poly-size circuits under either 1-AD-SIM security (thus 1-AD-IND and many-AD-IND security) or many-NA-SIM security.

In this work, we narrow the gap between existing security definitions for functional encryption, as well as that between existing constructions and impossibility results. Our results are as follows.

[2] [BSW11, Section 5.3] presents an "equivalence" between many-AD-IND and many-AD-SIM in the *programmable* random oracle model for public-index predicate encryption. For this work, we consider only the standard model.

[3] A commonly used relaxation of AD-IND security for predicate encryption is that of "selective security" [CHK03].

1.1 New Lower Bound: Impossibility for Simulation-based Definitions

Our main result rules out general functional encryption under the one message secure, non-adaptive simulation definition (1-NA-SIM). In particular, this rules out both of the scenarios presented at the end of the preceding section (i.e. 1-AD-SIM or many-NA-SIM for all circuits) in a strong sense. This is the *first* lower bound that exploits *unbounded* collusions in an essential way. We compare the impossibility result from [BSW11] with ours in Figure 2.

Theorem 1.1 (Informal)**.** *There exists a circuit family $\mathcal{C}$ for which there is no* 1-NA-SIM*-secure function encryption scheme.*

Specifically, assuming the existence of a family of weak pseudo-random function $\mathsf{wPRF}(\cdot,\cdot)$, we show that there does not exist a functional encryption scheme for the family:

$$C_d(x) = \mathsf{wPRF}(x, d), \text{where the input message } x \text{ is the PRF seed}$$

We show that the ciphertext size in a 1-NA-SIM-secure scheme realizing this circuit family must grow with the size of the collusion; this yields a contradiction, since the scheme must handle unbounded collusions. In fact, the result is unconditional since any non-trivial functional encryption scheme gives rise to a one-way function and thus pseudo-random functions.

The key observation is as follows. Suppose the adversary requests for q secret keys corresponding to random inputs $C_{d_1}, \ldots, C_{d_q}$ and then requests for an encryption of a random x. Then, the simulated ciphertext together with the q simulated secret keys constitute a description of the values $\mathsf{wPRF}(x, d_1), \ldots, \mathsf{wPRF}(x, d_q)$, which is essentially a sequence of q truly random bits via pseudo-randomness. By a standard information-theoretic argument, this means that the length of the ciphertext plus the secret keys must grow with q. To obtain a lower bound on the ciphertext size, we carefully exploit the fact that the simulator has to generate the secret keys before it sees the output of $\mathsf{wPRF}(x, \cdot)$. Then, the simulator has to generate a small ciphertext that "explains" all these pseudorandom values which is impossible using a compressibility argument. More generally, we show that (1) weak pseudo-random family is "incompressible", and (2) NA-SIM-secure functional encryption only exists for "compressible" circuit families. (In particular, the circuit family for all *public-index* predicate encryption is compressible.)

This idea is reminiscent of the obfuscation impossibility result of Goldwasser and Kalai [GK05], although the precise settings are quite different (in particular, functional encryption and program obfuscation seem incomparable, although related, objects).

Implications. The basic idea described above can be extended to a lower bound for even weaker forms of the simulation-based definition, including (a non-adaptive variant of) the definition of Boneh, Sahai and Waters [BSW11]. Here, we mention yet another implication of this idea.

Gorbunov, Vaikuntanathan and Wee [GVW12] recently presented a 1-AD-SIM-secure functional encryption scheme for all circuits, assuming that the adversary can only corrupt an a-priori bounded number of users (and thus, get the corresponding secret keys). One of the shortcomings of their bounded-collusion security notion as well as their construction is that the parameters of the system, and especially the size of the ciphertext depends on the collusion bound q. A natural question is whether their ciphertexts can be made to have size independent of q (or, at the very least, $o(q)$).[4] Indeed, in light of the results of Dodis, Katz, Xu and Yung [DKXY02] and most recently,

[4]The previous lower bound for many-AD-SIM IBE in [BSW11] (which says that the secret key size must grow with the number of challenge ciphertexts) is not applicable here as the [GVW12] construction considers only a single challenge ciphertext.

Goldwasser, Lewko and Wilson [GLW12] in the context of bounded-collusion IBE, one might expect that achieving "short" ciphertexts is actually be possible in general.

Unfortunately, our techniques result in a strong negative answer to this question.

Corollary 1.2. *There exists a family of circuits $\mathcal{C}$ such that for every $q = q(\kappa)$, there are no q-collusion resistant 1-*NA-SIM*-secure functional encryption schemes with ciphertexts of size $o(q)$.*

1.2 New Perspectives: Unbounded Simulation

The preceding lower bound together with those of Boneh, Sahai and Waters [BSW11] show that even fairly weak simulation-based definitions of functional encryption are unachievable for a large and natural class of circuits. This state of affairs begs the question:

What is a meaningful *and* generally realizable *security notion for functional encryption?*

While we do not provide a definitive answer to this question in our work, we firmly believe that the quest for the right definition should incorporate insights from secure computation and zero knowledge. Indeed, several recent works [GVW12, SS10] exploited techniques and insights from secure computation [Yao86, BGW88, BMR90] to derive general feasibility results for functional encryption with bounded collusions.

We put forth USIM security, where the simulator has unbounded computational power. In particular, this would allow us to circumvent our lower bound in the previous section. Similar notions have been considered for zero knowledge and secure computation [Pas03, PS04, BS05].[5]

Before presenting our results for USIM security, we first provide an intuitive interpretation of what USIM security buys us, via the real/ideal paradigm. Recall that polynomial-time simulation-based security for functional encryption guarantees that against a computational bounded adversary holding a secret key SK_C, an encryption of x leaks no more information about x than what an efficient adversary can deduce given $C(x)$ (and C); allowing unbounded simulation means that an encryption of x leaks no more information about x than what a computationally unbounded adversary can deduce given $C(x)$ (and x). Indeed, in the case of public-index predicate encryption, $C(x)$ does hide x completely against a computationally unbounded adversary that only holds keys for which the predicate is false. One could even make the case that for public-index predicate encryption, USIM security is "as good as" SIM security![6] On the other hand, for circuits that only hide information about x computationally, USIM security would be inadequate and SIM security remains the desirable notion.

Next, we establish basic relations between USIM security and SIM, IND security, namely it is "sandwiched" between the two, that is, for $\mathsf{yy} \in \{\mathsf{NA}, \mathsf{AD}\}$:

$$\mathsf{yy\text{-}IND} \;\Leftarrow\; \mathsf{yy\text{-}USIM} \;\Leftarrow\; \mathsf{yy\text{-}SIM}$$

This inclusion yields a simple "litmus test" for checking if IND security is inadequate for a circuit family $\mathcal{C}$: IND security is inadequate whenever USIM security is inadequate, namely $C(x)$ reveals more information about x to an unbounded adversary than an efficient adversary.

Furthermore, with this notion in mind, we refine and further clarify two results in [BSW11]:

[5] The works on zero knowledge and secure computation focus on quasi-polynomial-time simulators. We observe that our lower bound also rules out quasi-polynomial-time simulators assuming the existence of one-way functions with sub-exponential hardness.

[6] O'Neill [O'N10, Section 4] showed that NA-IND and NA-SIM are equivalent for public-index predicate encryption. This does not subsume the point we are making because our argument applies also to the adaptive setting. Indeed, AD-IND and AD-SIM are provably *not* equivalent for public-index predicate encryption; under standard assumptions, we have AD-IND-secure IBE, whereas AD-SIM-secure IBE do not exist.

- the counter-example separating indistinguishability and simulation-based notion (which encodes a one-way permutation into the circuit family) in fact separates efficient and unbounded simulation; there, the circuit inherently leaks more information to an unbounded adversary than an efficient adversary. That is, the result really points to the inadequacy of the unbounded simulation security (and not indistinguishability-based notion) for certain families.

- the lower bound for IBE under many-AD-SIM security extends to many-AD-USIM; that is, the result is fundamentally about a simulation-based security notion, and not about efficiency.

The reader is referred to Figure 1 for a survey of our results and open problems, and to Appendix C for results on the unbounded simulator definition.

1.3 Discussion

FunctoMania. Let's be wishful thinkers for a minute – suppose we can have whatever we hope for in functional encryption, call this world "Functomania". What does Functomania look like? In light of the existing (im)possibilities, there will be two incomparable "dream results":

- 1-AD-SIM secure public index predicate encryption for all efficient predicates; such schemes also satisfy 1-AD-IND, 1-AD-USIM, and many-AD-IND security.

- 1-AD-USIM secure functional encryption for all poly-size circuits; such schemes also satisfy 1-AD-IND and many-AD-IND security.

Given the current state of affairs in functional encryption, establishing either result in the affirmative (or even under the weaker 1-AD-IND security) will be considered a major break-through.

The IND-(U)SIM Conundrum. From a definitional stand-point, SIM/USIM-based security notions are preferable to IND-based security notion, as they offer a stronger security guarantee that has a natural, intuitive and aesthetically pleasing interpretation via the real/ideal paradigm. On the other hand, IND-based security notion allows us to bypass the impossibility results given in [BSW11] and in this work; in addition, they guarantee message *composability* in that security with a single ciphertext implies security for multiple ciphertexts (and so does NA-SIM considered in [GVW12]). We do not offer a complete answer to this conundrum; instead, we point out that 1-AD-SIM and 1-AD-USIM appear to be an adequate compromise for predicate encryption and general functional encryption respectively. We also note that such a conundrum is not unique to functional encryption, and has indeed previously surfaced and widely studied in the context of zero knowledge [FS90, Pas03] and secure multi-party computation [PS04, BS05, MPR06]. One notable difference is that in zero knowledge and secure computation, super-polynomial time simulation offers concurrency; this is not the case for functional encryption. (The lower bound for many-AD-USIM-secure IBE indicates that even unbounded-time simulation does not help with message composability.)

Beyond Black-Box Simulation? In an independent work, Bellare and O'Neill [BO12] put forth simulation-based definitions for functional encryption with non-black-box simulators. (The definitions we study in this work are "inherently black-box" since the simulator must explicitly provide the adversary with secret keys and ciphertexts.) In addition, they extended the [BSW11] lower bound for IBE to the setting of efficient, non-black-box simulators, assuming the existence of collision-resistant hash functions. This leaves as an open problem the question of realizing (or ruling out) many-AD-USIM IBE with a non-black-box simulator.

	Our impossibility result (Theorem 4.2)	Boneh, Sahai and Waters ([BSW11, Theorem 2])
adaptive vs. non-adaptive	non-adaptive	adaptive
one vs. many messages	one message	many messages
one vs. many secret-key queries	many queries	one query
class of circuits	weak PRFs	IBE

Figure 2: A comparison between the BSW lower bound (see also Section C.2) and ours for functional encryption. The underlines indicates the stronger result. For example, the first row says that our impossibility result rules out even a non-adaptive notion of security and is thus, stronger than the BSW result that rules out an adaptive notion.

2 Functional Encryption

Let $\mathcal{X} = \{\mathcal{X}_\kappa\}_{\kappa\in\mathbb{N}}$ and $\mathcal{Y} = \{\mathcal{Y}_\kappa\}_{\kappa\in\mathbb{N}}$ denote ensembles where each $\mathcal{X}_\kappa$ and $\mathcal{Y}_\kappa$ is a finite set. Let $\mathcal{C} = \{\mathcal{C}_\kappa\}_{\kappa\in\mathbb{N}}$ denote an ensemble where each $\mathcal{C}_\kappa$ is a finite collection of circuits, and each circuit $C \in \mathcal{C}_\kappa$ takes as input a string $x \in \mathcal{X}_\kappa$ and outputs $C(x) \in \mathcal{Y}_\kappa$.

A functional encryption scheme $\mathcal{FE}$ for $\mathcal{C}$ consists of four algorithms $\mathcal{FE} = (\mathsf{FE.Setup}, \mathsf{FE.Keygen}, \mathsf{FE.Enc}, \mathsf{FE.Dec})$ defined as follows.

- **Setup** $\mathsf{FE.Setup}(1^\kappa)$ is a p.p.t. algorithm takes as input the unary representation of the security parameter and outputs the master public and secret keys $(\mathsf{MPK}, \mathsf{MSK})$.
- **Key Generation** $\mathsf{FE.Keygen}(\mathsf{MSK}, C)$ is a p.p.t. algorithm that takes as input the master secret key MSK and a circuit $C \in \mathcal{C}_\kappa$ and outputs a corresponding secret key SK_C.
- **Encryption** $\mathsf{FE.Enc}(\mathsf{MPK}, x)$ is a p.p.t. algorithm that takes as input the master public key MPK and an input message $x \in \mathcal{X}_\kappa$ and outputs a ciphertext CT.
- **Decryption** $\mathsf{FE.Dec}(\mathsf{SK}_C, \mathsf{CT})$ is a deterministic algorithm that takes as input the secret key SK_C and a ciphertext CT and outputs $C(x)$.

Definition 2.1 (Correctness)**.** *A functional encryption scheme $\mathcal{FE}$ is correct if for all $C \in \mathcal{C}_\kappa$ and all $x \in \mathcal{X}_\kappa$,*

$$\Pr\left[\begin{array}{l}(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{FE.Setup}(1^\kappa);\\ \quad \mathsf{FE.Dec}(\mathsf{FE.Keygen}(\mathsf{MSK}, C), \mathsf{FE.Enc}(\mathsf{MPK}, x)) \neq C(x)\end{array}\right] = \mathrm{negl}(\kappa)$$

where the probability is taken over the coins of $\mathsf{FE.Setup}$*,* $\mathsf{FE.Keygen}$*, and* $\mathsf{FE.Enc}$*.*

2.1 A Simulation-based Definition of Security

In this section, we present a simulation-based definition of functional encryption, similar in spirit to the way one defines security for secure computation via the ideal/real paradigm. We define the security game for a single message since our lower bounds apply to this weaker setting. However, this definition can be easily extended to many messages setting (see Appendix B).

Definition 2.2 (1-NA-SIM- and 1-AD-SIM- Security)**.** *Let $\mathcal{FE}$ be a functional encryption scheme for a circuit family $\mathcal{C}$. Consider a p.p.t. adversary $A = (A_1, A_2)$ and a stateful p.p.t. simulator* Sim.[7] *Let $U_x(\cdot)$ denote a universal oracle, such that $U_x(C) = C(x)$. Consider the following two experiments:*

$\underline{\mathsf{Exp}^{\mathsf{real}}_{\mathcal{FE},A}(1^\kappa)\textbf{:}}$

1: $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{FE.Setup}(1^\kappa)$
2: $(x, st) \leftarrow A_1^{\mathsf{FE.Keygen}(\mathsf{MSK},\cdot)}(\mathsf{MPK})$
3: $\mathsf{CT} \leftarrow \mathsf{FE.Enc}(\mathsf{MPK}, x)$
4: $\alpha \leftarrow A_2^{\mathcal{O}(\mathsf{MSK},\cdot)}(\mathsf{MPK}, \mathsf{CT}, st)$
5: *Output* (x, α)

$\underline{\mathsf{Exp}^{\mathsf{ideal}}_{\mathcal{FE},\mathrm{Sim}}(1^\kappa)\textbf{:}}$

1: $\mathsf{MPK} \leftarrow \mathrm{Sim}(1^\kappa)$
2: $(x, st) \leftarrow A_1^{\mathrm{Sim}(\cdot)}(\mathsf{MPK})$
3: $\mathsf{CT} \leftarrow \mathrm{Sim}^{U_x(\cdot)}(1^\kappa, 1^{|x|})$
4: $\alpha \leftarrow A_2^{\mathcal{O}'(\cdot)}(\mathsf{MPK}, \mathsf{CT}, st)$
5: *Output* (x, α)

We distinguish between two cases of the above experiment:

1. The adaptive experiment*, where:*

   - *the oracle $\mathcal{O}(\mathsf{MSK}, \cdot) = \mathsf{FE.Keygen}(\mathsf{MSK}, \cdot)$ and*
   - *the oracle $\mathcal{O}'(\cdot)$ is the simulator, namely $\mathrm{Sim}^{U_x(\cdot)}(\cdot)$*

   We call a stateful simulator algorithm Sim admissible *if, on each input C,* Sim *makes just a single query to its oracle $U_x(\cdot)$ on C itself.*

   The functional encryption scheme $\mathcal{FE}$ is then said to be simulation-secure for one message against adaptive adversaries (1-AD-SIM-secure, for short) *if there is an* admissible *stateful p.p.t. simulator* Sim *such that for every p.p.t. adversary $A = (A_1, A_2)$, the following two distributions are computationally indistinguishable:*

$$\left\{\mathsf{Exp}^{\mathsf{real}}_{\mathcal{FE},A}(1^\kappa)\right\}_{\kappa\in\mathbb{N}} \overset{c}{\approx} \left\{\mathsf{Exp}^{\mathsf{ideal}}_{\mathcal{FE},\mathrm{Sim}}(1^\kappa)\right\}_{\kappa\in\mathbb{N}}$$

2. The non-adaptive experiment*, where the oracles $\mathcal{O}(\mathsf{MSK}, \cdot)$ and $\mathcal{O}'(\cdot)$ are both the "empty oracles" that return nothing.*

   The functional encryption scheme $\mathcal{FE}$ is then said to be simulation-secure for one message against non-adaptive adversaries (1-NA-SIM-secure, for short) *if there is an* admissible *stateful p.p.t. simulator* Sim *such that for every p.p.t. adversary $A = (A_1, A_2)$, the two distributions above are computationally indistinguishable.*

Remarks on the Definition. Our definition is stronger than that in [BSW11] but weaker than that in [GVW12]; our lower bound in Section 4 holds for all three definitions. Amongst the three, the one in [GVW12] is the only for which we know a composition theorem where security for one message implies security for many messages, in the non-adaptive setting. Note that composition in the non-adaptive setting is the "best" we can hope for; composition in the adaptive setting is essentially impossible by many-AD-SIM lower bound for IBE [BSW11]. In more detail:

[7] One can replace a stateful simulator can be replaced by a regular (stateless) simulator that outputs a state st_s upon each invocation which is carried over to its next invocation.

- In [BSW11], the simulator is given oracle access to A_2, which it can call on any ciphertext. Therefore, it can "rewind" the adversary A_2 and adaptively reconstruct the view, which is problematic for composition [PRS02, Lin04, BMQU07]. We call this a "rewinding" definition. In our "straight-line" definition, the simulator must commit to a ciphertext once and for all, which makes it stronger.

- Unlike our definition, the [GVW12] definition does not allow the simulator to fake or "program" the setup parameters and the secret keys. The difficulty in proving a composition theorem for our definition lies in that the simulator may use "trapdoor" information from faking the setup parameters and secret keys while simulating the ciphertext.

We note that in the equivalence of NA-IND and NA-SIM under pre-image sampleability in [O'N10, Section 4], the NA-SIM-simulator actually satisfies the stronger definition in [GVW12].

The Indistinguishability-based Definition of Security. We define the non-adaptive NA-IND and the adaptive AD-IND notions of security in Appendix A.

3 Preliminaries

Notations. Let $\mathcal{D}$ denote a distribution over some finite set S. Then, $x \leftarrow \mathcal{D}$ is used to denote the fact that x is chosen from the distribution $\mathcal{D}$. When we say $x \leftarrow S$, we simply mean that x is chosen from the uniform distribution over S. Unless explicitly mentioned, all logarithms are to base 2. For $n \in \mathbb{N}$, let $[n]$ denote the set of numbers $1, \ldots, n$. Let κ denote the security parameter.

Definition 3.1 (wPRF)**.** *Let* $\mathsf{wPRF} = \{\mathsf{wPRF}_\kappa\}_{\kappa\in\mathbb{N}}$ *denote a family of efficiently computable functions where* $\mathsf{wPRF}_\kappa : \{0,1\}^{n(\kappa)} \times \{0,1\}^{m(\kappa)} \to \{0,1\}^{k(\kappa)}$*, the first argument of which is called the seed to the wPRF and the second argument is the input.*

For every probabilistic polynomial time oracle distinguisher Dist*, consider the following two experiments:*

- $\mathsf{Real}_{\mathrm{Dist}}(1^\kappa)$: *Choose* $x \xleftarrow{\$} \{0,1\}^{n(\kappa)}$ *and run* Dist *with access to a probabilistic oracle* $\mathcal{O}_{real}(x)$ *which, when invoked, chooses a uniformly random* $d \leftarrow \{0,1\}^{m(\kappa)}$ *and returns the pair* $(d, \mathsf{wPRF}_\kappa(x,d))$*. This experiment outputs whatever* Dist *outputs.*

- $\mathsf{Rand}_{\mathrm{Dist}}(1^\kappa)$: *Choose a uniformly random function* $R : \{0,1\}^{m(\kappa)} \to \{0,1\}^{k(\kappa)}$ *and run* Dist *with access to a probabilistic oracle* $\mathcal{O}_{rand}(R)$ *which, when invoked, chooses a uniformly random* $d \leftarrow \{0,1\}^{m(\kappa)}$ *and returns the pair* $(d, R(d))$*. This experiment outputs whatever* Dist *outputs.*

We say wPRF *is a weak pseudo-random function if for all p.p.t. distinguishers* Dist*,*

$$\left|\Pr[\mathsf{Real}_{\mathrm{Dist}}(1^\kappa) = 1] - \Pr[\mathsf{Rand}_{\mathrm{Dist}}(1^\kappa) = 1]\right| = \mathrm{negl}(\kappa)$$

where the probabilities are over the choice of x *and* R*, as well as the coin-tosses of* Dist *and the oracles* $\mathcal{O}_{real}$ *and* $\mathcal{O}_{rand}$*.*

In our impossibility result, we will use a weak pseudo-random function with seed length $n(\kappa) = \kappa$ and output length $k(\kappa) = 1$.

4 Impossibility Results for Functional Encryption

In this section, we present our main lower bound for 1-NA-SIM-secure functional encryption. We begin with a notion of "incompressible" circuits. Then, we show that (1) weak pseudo-random functions are "incompressible", and (2) 1-NA-SIM-secure functional encryption only exists for "compressible" circuits. Putting the two together yields our lower bound.

4.1 Incompressible Circuits

We first define a family of compressible circuits. Informally, we say that a family of circuits $\{\mathcal{G}_\kappa\}$ is (ℓ, t)-compressible if for a list of uniformly random circuit descriptions $G_1, \ldots, G_\ell \in \mathcal{G}_\kappa$ and a uniformly chosen input x, there is some efficiently computable description of $G_1(x), \ldots, G_\ell(x)$ of size t. Note that if there is no efficiency requirement, then any family is $(\ell, |s|)$-compressible.

Definition 4.1 (Incompressible Circuits)**.** *Let $\ell = \ell(\kappa)$ and $t = t(\kappa)$ be functions of the security parameter κ. A family of circuits $\mathcal{G} = \{\mathcal{G}_\kappa\}_{\kappa\in\mathbb{N}}$ is* (ℓ, t)-compressible *if there exist a family of (deterministic) compressor circuits $\{\mathbf{C}_\kappa\}_{\kappa\in\mathbb{N}}$ and a family of decompressor circuits $\{\mathbf{D}_\kappa\}_{\kappa\in\mathbb{N}}$ such that:*

- *(polynomial size) the circuits $\mathbf{C}_\kappa$ and $\mathbf{D}_\kappa$ have size $\mathsf{poly}(\kappa, \ell)$.*
- *(mild compression) for sufficiently large κ, $\left|\mathbf{C}_\kappa(G_1, \ldots, G_\ell, y_1, \ldots, y_\ell)\right| = t$, where $y_i = G_i(x)$.*
- *(correctness) there is a polynomial $p = p(\kappa)$ such that*

$$\Pr[x \xleftarrow{\$} \{0,1\}^\kappa, G_1, \ldots, G_\ell \xleftarrow{\$} \mathcal{G}_\kappa, y_i = G_i(x) : \mathbf{D}_\kappa(G_1, \ldots, G_\ell, \mathbf{C}_\kappa(G_1, \ldots, G_\ell, y_1, \ldots, y_\ell)) = (y_1, \ldots, y_\ell)] \geq 1/p(\kappa)$$

where the probability is taken over the choice of x as well as the circuits $G_1, \ldots, G_\ell$.

The family $\mathcal{G}$ is (ℓ, t)-incompressible *if it is not (ℓ, t)-compressible.*

We now give examples of (in)compressible circuits. First, consider the notion of pre-image samplable family of circuits introduced by O'Neill [O'N10] which requires that given $G_1(x), \ldots, G_\ell(x)$, there is a polynomial-time algorithm that returns an arbitrary x' such that $G_i(x') = G_i(x)$ for all i. In our language, this says that the family $\mathcal{G}$ is $(\ell, |x'|)$-compressible; the compression algorithm simply outputs x'.

Next, consider an arbitrary public-index circuit family parametrized by predicates P and given by:

$$G_P(\mathsf{ind}, \mu) = \begin{cases} (\mathsf{ind}, \mu) & \text{if } P(\mathsf{ind}) = 1 \\ (\mathsf{ind}, \bot) & \text{otherwise} \end{cases}$$

It is easy to see that this circuit family is $(\ell, |(\mathsf{ind}, \mu)|)$-compressible. On input

$$G_{P_1}(\mathsf{ind}, \mu), \ldots, G_{P_\ell}(\mathsf{ind}, \mu)$$

If $P_i(\mathsf{ind}) = 1$ for some i, then the compression algorithm outputs (ind, μ). If $P_i(\mathsf{ind}) = 0$ for all i, then the algorithm outputs $(\mathsf{ind}, \bot)$.

On the other hand, as we show below (see Lemma 4.1), any family of (weak) pseudo-random functions is incompressible in a strong sense. More precisely, consider a family of

circuits $\mathcal{G} = \{G_{d_i}(\cdot) = \mathsf{wPRF}(\cdot, d_i)\}$ where d_i serves as the input to the pseudo-random function. Informally, the incompressibility is due to the fact that a sequence $(G_{d_1}(x), \ldots, G_{d_\ell}(x)) = (\mathsf{wPRF}(x, d_1), \ldots, \mathsf{wPRF}(x, d_\ell))$ is indistinguishable from a sequence of uniformly random bits, which are clearly incompressible.

Lemma 4.1 (weak PRFs are $(\ell, \ell - \kappa)$-incompressible)**.** *Let* $\mathsf{wPRF} = \{\mathsf{wPRF}_\kappa : \{0,1\}^\kappa \times \{0,1\}^{m(\kappa)} \to \{0,1\}\}_{\kappa \in \mathbb{N}}$ *be a family of weak pseudo-random functions, where* $m(\kappa) = \omega(\log \kappa)$*. Define* $G_d(x) = \mathsf{wPRF}(x, d)$*. Consider a family* $\mathcal{G} = \{\mathcal{G}_\kappa\}_{\kappa \in \mathbb{N}}$ *defined as*

$$\mathcal{G}_\kappa = \left\{G_d(\cdot) : |d| = m(\kappa)\right\}$$

Then, $\mathcal{G}$ *is* $(\ell, \ell - \kappa)$*-incompressible.*

Proof. Assume, for the sake of contradiction, that $\mathcal{G}$ is $(\ell, \ell - \kappa)$-compressible. Namely, there are families of compressor and decompressor circuits $(\mathbf{C}, \mathbf{D})$ that satisfy Definition 4.1. We show how to construct a distinguisher $\mathrm{Dist}^{\mathcal{O}}$ that distinguishes between the case where $\mathcal{O} = \mathsf{wPRF}(x, \cdot)$ is a pseudo-random oracle that outputs pairs $(d_i, y_i = \mathsf{wPRF}_\kappa(x, d_i))$ where d_i are uniformly random, and the case where $\mathcal{O}$ outputs strings $(d_i, y_i = R(d_i))$ where d_i and R are uniformly random strings and function, respectively. $\mathrm{Dist}^{\mathcal{O}}$ proceeds as follows.

- Choose a sufficiently large κ such that

$$\left|\mathbf{C}_\kappa(G_1, \ldots, G_\ell, y_1, \ldots, y_\ell)\right| = \ell - \kappa$$

- Query the oracle $\mathcal{O}$ to obtain pairs of strings of the form $(d_i \xleftarrow{\$} \{0,1\}^{m(\kappa)}, y_i)$. Define the circuit $G_{d_i}(\cdot) := \mathsf{wPRF}(\cdot, d_i)$.

- Run the compressor $\mathbf{C}_\kappa$ to get a string

$$\gamma \leftarrow \mathbf{C}_\kappa(G_{d_1}, \ldots, G_{d_\ell}, y_1, \ldots, y_\ell)$$

- Outputs 1 if and only if

$$\mathbf{D}_\kappa(G_{d_1}, \ldots, G_{d_\ell}, \gamma) = (y_1, \ldots, y_\ell)$$

We now show that the distinguisher succeeds with non-negligible advantage $1/p(\kappa) - 2^{-\kappa}$ in breaking the weak pseudo-random function family wPRF.

If $\mathcal{O}$ is the pseudo-random oracle, then the samples $\mathrm{Dist}^{\mathcal{O}}$ gets are of the form $(d_i, y_i \leftarrow \mathsf{wPRF}(x, d_i))$. Hence, by correctness of $\mathbf{C}$ and $\mathbf{D}$,

$$\mathbf{D}_\kappa(G_{d_1}, \ldots, G_{d_\ell}, \mathbf{C}_\kappa(G_{d_1}, \ldots, G_{d_\ell}, y_1, \ldots, y_\ell)) = (y_1, \ldots, y_\ell)$$

with probability at least $1/p(\kappa)$. Thus, the distinguisher in this case outputs 1 with probability at least $1/p(\kappa)$ as well.

On the other hand, if $\mathcal{O}$ outputs pairs of strings of the form $(d_i, y_i \leftarrow R(d_i))$ for a randomly chosen function mapping R, we now show that the distinguisher above outputs 1 with probability at most $2^{-\kappa}$. In the analysis below, we assume that $d_1, \ldots, d_\ell$ are distinct, for which we need to pay a price of an additive $\ell^2 \cdot 2^{-m(\kappa)} = \mathrm{negl}(\kappa)$ term in the distinguishing error.

$$
\begin{aligned}
&\Pr[\mathrm{Dist}^{\mathcal{O}} \text{ outputs } 1] \\
&\leq \Pr_{\substack{d_1,\ldots,d_\ell \xleftarrow{\$} \{0,1\}^{m(\kappa)} \\ y_1,\ldots,y_\ell \xleftarrow{\$} \{0,1\}}} \left[\exists \gamma : |\gamma| = \ell - \kappa \text{ and } \mathbf{D}_\kappa(G_{d_1}, \ldots, G_{d_\ell}, \gamma) = (y_1, \ldots, y_\ell)\right] \\
&\leq \sum_{\gamma \in \{0,1\}^{\ell-\kappa}} \Pr_{\substack{d_1,\ldots,d_\ell \xleftarrow{\$} \{0,1\}^{m(\kappa)} \\ y_1,\ldots,y_\ell \xleftarrow{\$} \{0,1\}}} \left[\mathbf{D}_\kappa(G_{d_1}, \ldots, G_{d_\ell}, \gamma) = (y_1, \ldots, y_\ell)\right] \quad \text{(via a union bound)} \\
&= \sum_{\gamma \in \{0,1\}^{\ell-\kappa}} 2^{-\ell} \quad \text{(since } y_1, \ldots, y_\ell \text{ are random and independent of } d_1, \ldots, d_\ell, \gamma) \\
&\leq 2^{\ell-\kappa} \cdot 2^{-\ell} = 2^{-\kappa}
\end{aligned}
$$

This yields the required contradiction to the security of wPRF. □

4.2 The Impossibility Result

We are now ready to state and prove our main theorem.

Theorem 4.2. *There exists a family of circuits $\mathcal{G}$ for which there are no* 1-NA-SIM*-secure functional encryption schemes.*

Proof. We consider two cases.

Case 1: Assume there exists a circuit family of weak pseudo-random functions

$$\mathsf{wPRF} = \{\mathsf{wPRF}_\kappa : \{0,1\}^\kappa \times \{0,1\}^{m(\kappa)} \to \{0,1\}\}_{\kappa \in \mathbb{N}}$$

where $m(\kappa) = \omega(\log \kappa)$. Let $G_d(x) = \mathsf{wPRF}(x, d)$ and consider a family $\mathcal{G} = \{\mathcal{G}_\kappa\}_{\kappa \in \mathbb{N}}$ defined as

$$\mathcal{G}_\kappa = \left\{G_d(\cdot) : |d| = m(\kappa)\right\}$$

Assume, for the sake of contradiction, there exist a 1-NA-SIM-secure function encryption scheme $\mathcal{FE}$ for $\mathcal{G}$, and let $|\mathsf{CT}|$ denote the length of a ciphertext in the scheme. Let $\ell = \ell(\kappa) = |\mathsf{CT}| + \kappa$.

From Lemma 4.1, we know that $\mathcal{G}$ is $(|\mathsf{CT} + \kappa, |\mathsf{CT}|)$-incompressible. However, Lemma 4.3 below tells us that since there is a 1-NA-SIM secure scheme for $\mathcal{G}$, the family $\mathcal{G}$ is $(|\mathsf{CT}| + \kappa, |\mathsf{CT}|)$-compressible. This gives us the desired contradiction, and therefore, there cannot exist a 1-NA-SIM-secure functional encryption scheme for $\mathcal{G}$.

Case 2: Assume there does not exist a family of weak pseudo-random functions. Also, for the sake of contradiction, assume there exists a 1-NA-SIM-secure function encryption scheme for all families of circuits $\mathcal{G}$.

In particular, this means that there is a functional encryption scheme for the empty circuit family (namely, a family $\mathcal{G}$ that does not contain any circuits at all). A 1-NA-SIM-secure scheme $\mathcal{FE}$ for $\mathcal{G}$ is also a secure public-key encryption scheme. Since public-key encryption implies one-way functions, which in turn imply pseudo-random functions [GGM86, HILL99], we obtain the desired contradiction. □

Lemma 4.3 (1-NA-SIM $\Rightarrow$ $(\ell, |\mathsf{CT}|)$-compressibility)**.** *Let $\mathcal{G} = \{\mathcal{G}_\kappa\}_{\kappa \in \mathbb{N}}$ be a family of circuits. Suppose there exists a* 1-NA-SIM*-secure functional encryption scheme for the $\mathcal{G}$. Then, the family $\mathcal{G}$ is $(\ell, |\mathsf{CT}|)$-compressible for any polynomially bounded $\ell = \ell(\kappa)$, where $|\mathsf{CT}|$ denotes size of the encryption of input x.*

Informally, the compression algorithm works as follows: on input $G_1, \ldots, G_\ell$ and $G_1(x), \ldots, G_\ell(x)$, the output is the simulated ciphertext corresponding to an encryption of x. The decompression algorithm then evaluates the decryption algorithm, which is guaranteed to produce $G_1(x), \ldots, G_\ell(x)$.

Proof. Let $(\mathsf{FE.Setup}, \mathsf{FE.Keygen}, \mathsf{FE.Enc}, \mathsf{FE.Dec})$ denote the encryption scheme for the family $\mathcal{G}$. Consider the adversary $A = (A_1, A_2)$ in the 1-NA-SIM security experiment that acts as follows:

- A_1 chooses $G_1, \ldots, G_\ell \xleftarrow{\$} \mathcal{G}$ independently at random and requests for the corresponding secret keys $\mathsf{SK}_1, \ldots, \mathsf{SK}_\ell$. In addition, it chooses $x \xleftarrow{\$} \{0,1\}^{m(\kappa)}$ and outputs x as the challenge message, and $(G_1, \ldots, G_\ell, \mathsf{SK}_1, \ldots, \mathsf{SK}_\ell)$ as the state.
- A_2 outputs α composed of the challenge ciphertext and the state $(G_1, \ldots, G_\ell, \mathsf{SK}_1, \ldots, \mathsf{SK}_\ell)$.

Let Sim denote the (admissible) stateful p.p.t. simulator guaranteed by 1-NA-SIM security. We show how to use the simulator to construct a family of (deterministic) compressor and decompressor circuits $\mathbf{C}_\rho$ and $\mathbf{D}_\rho$, indexed by a random string ρ corresponding to the random tape for the simulator:

- The compressor $\mathbf{C}_\rho$, on input $G_1, \ldots, G_\ell$ and $y_1, \ldots, y_\ell$ works as follows: first, compute $\mathsf{MPK} \leftarrow \mathrm{Sim}(1^\kappa;\ \rho)$ and secret keys $\{\mathsf{SK}_i : \mathsf{SK}_i \leftarrow \mathrm{Sim}(G_i;\ \rho)\}_{i \in [\ell]}$. Then compute and output CT as the compressed string, where queries $G_i(x)$ are answered with y_i:

$$\mathsf{CT} \leftarrow \mathrm{Sim}^{U_x(\cdot)}(1^{|m(\kappa)|})$$

- The decompressor $\mathbf{D}_\rho$, on input $G_1, \ldots, G_\ell$ and CT first reconstructs the master public key $\mathsf{MPK} \leftarrow \mathrm{Sim}(1^\kappa;\ \rho)$ and the set of secret keys:

$$\{\mathsf{SK}_i : \mathsf{SK}_i \leftarrow \mathrm{Sim}(G_i;\ \rho)\}_{i \in [\ell]}$$

  Note that $\mathbf{D}_\rho$ has the same randomness ρ hard-wired, and so the secret keys SK_i are exactly the same as those used by $\mathbf{C}_\rho$. Finally, it computes and outputs:

$$\left\{y_i \leftarrow \mathsf{FE.Dec}(\mathsf{SK}_i, \mathsf{CT})\right\}_{i \in [\ell]}$$

Formally, we output $(\mathbf{C}_\rho, \mathbf{D}_\rho)$ for a random ρ, which is a pair of polynomial-size circuits. Clearly, we achieve $(\ell, |\mathsf{CT}|)$-compressibility, since the size of CT is determined by the functional encryption scheme and independent of ℓ. To establish correctness, it suffices to show that:

$$\Pr_{\rho, x, G_1, \ldots, G_\ell}[\mathbf{D}_\rho(G_1, \ldots, G_\ell, \mathbf{C}_\rho(G_1, \ldots, G_\ell, G_1(x), \ldots, G_\ell(x))) = (G_1(x), \ldots, G_\ell(x))] \geq 1 - \mathrm{negl}(\kappa)$$

Here, we will rely on the correctness of the functional encryption scheme as well as 1-NA-SIM-security. First, consider the distinguisher Dist that given the output $(x, \mathsf{CT}, G_1, \ldots, G_\ell, \mathsf{SK}_1, \ldots, \mathsf{SK}_\ell)$ of the adversary A_2 proceeds as follows:

> Output 1 iff for all $i \in [\ell]$, $\mathsf{FE.Dec}(\mathsf{SK}_i, \mathsf{CT}) = G_i(x)$.

Observe that by correctness of the encryption scheme, Dist outputs 1 with probability $1 - \mathrm{negl}(\kappa)$ given the output of the adversary A_2 in the 1-NA-SIM experiment. Therefore, by 1-NA-SIM-security, Dist also outputs 1 with probability $1 - \mathrm{negl}(\kappa)$ given the output of the (admissible) simulator, where the randomness is taken over the coin tosses ρ of the simulator, along with the random choices of $x, G_1, \ldots, G_\ell$.

This shows that the pair of circuits $(\mathbf{C}_\rho, \mathbf{D}_\rho)$ for a uniformly random ρ is a correct compressor-decompressor pair, establishing the lemma. □

We point out here that our lower bound extends to the setting where the simulator is not required to be admissible, by using a family of (standard) pseudo-random functions.

Finally, the argument here generalizes to showing that functional encryption secure against an a-priori bounded number $q = q(\kappa)$ of collusions is impossible if one insists on small ciphertexts (namely, ciphertexts with much fewer than q bits). This matches the recent result of [GVW12] who construct such functional encryption schemes with ciphertexts of size polynomial in q.

Corollary 4.4. *There exists a family of circuits $\mathcal{G}$ such that for every $q = q(\kappa)$, there are no q-collusion resistant 1-*NA-SIM*-secure functional encryption schemes. with ciphertexts of size $o(q)$.*

4.3 Extensions: Impossibility of Weaker Simulation-based Definitions

The idea behind our impossibility result is robust enough to apply to various relaxations of the simulation-based security definition. In this section, we describe a number of such extensions of our result.

Impossibility for the selective and random-input definitions. In the selective model, the adversary is required to commit to the secret key queries $G_1, \ldots, G_q$ as well as the challenge input x before the setup phase. In particular, this means that the adversary will not be able to pick up the circuits or the challenge input depending on the system parameters. Variants of the selective security model are frequently considered in the literature as a relaxations of regular security notions (see, e.g., [BB11, GPSW06, AFV11]). Another relaxation one can consider is one where the adversary is not allowed to choose the circuits or the challenge, but instead, they are chosen uniformly at random.

Our lower bound easily extends to these weaker notions, simply because the adversary we consider in the proof of Lemma 4.3 chooses the circuits and the challenge uniformly at random, and independent of the system parameters.

Impossibility for the non-adaptive BSW Definition (the "Rewinding Definition"). The main difference between the definition proposed by [BSW11] and our definition in Section 2 is that whereas our definition restricts the simulator to be "straight-line", the BSW definition allows the simulator to "rewind" the adversary and interact with it in order to generate the view. For more details, we direct the reader to the discussion after Definition 2.2.

The proof of Lemma 4.3 transparently extends to the BSW definition. The adversary A is the same as in the proof. The compressor $\mathbf{C}$ runs the simulator, executing the code of the designated adversary A to compute the response whenever the simulator queries ("rewinds") A. Also, since the simulator is admissible, the queries it makes are exactly the ones that the compressor knows the answer to. As before, we can make the impossibility result work for even non-admissible simulators by appealing to regular (rather than weak) PRFs.

Impossibility for Secret-key Functional Encryption. In the setting of secret-key functional encryption (first considered by Shi, Shen and Waters [SSW09] in its predicate encryption variant), the encryption algorithm relies on the master secret key to produce the ciphertext for an input x.

All our impossibility results carry over to the setting of secret key functional encryption since in the proof of Lemma 4.3, neither the compressor nor the decompressor needs to run the encryption algorithm and generate ciphertexts.

Acknowledgments

We thank Shafi Goldwasser, Yael Kalai, Raluca Ada Popa and Charles Rackoff for a number of insightful conversations that helped improve the presentation of the impossibility result. We would also like to thank Adam O'Neill and Amit Sahai for helpful pointers and discussions.

References

[ABV+12] Shweta Agrawal, Xavier Boyen, Vinod Vaikuntanathan, Panagiotis Voulgaris, and Hoeteck Wee. Functional encryption for threshold functions (or fuzzy IBE) from lattices. In *PKC*, 2012.

[AFV11] Shweta Agrawal, David Mandell Freeman, and Vinod Vaikuntanathan. Functional encryption for inner product predicates from learning with errors. In *Asiacrypt*, 2011.

[BB11] Dan Boneh and Xavier Boyen. Efficient selective identity-based encryption without random oracles. *J. Cryptology*, 24(4):659–693, 2011.

[BF01] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. In *CRYPTO*, pages 213–229, 2001.

[BGW88] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, STOC '88, pages 1–10, New York, NY, USA, 1988. ACM.

[BMQU07] Michael Backes, Jörn Müler-Quade, and Dominique Unruh. On the necessity of rewinding in secure multiparty computation. In *Proceedings of the 4th conference on Theory of cryptography*, TCC'07, pages 157–173, Berlin, Heidelberg, 2007. Springer-Verlag.

[BMR90] Donald Beaver, Silvio Micali, and Phillip Rogaway. The round complexity of secure protocols (extended abstract). In *STOC*, pages 503–513, 1990.

[BO12] Mihir Bellare and Adam O'Neill. Semantically-secure functional encryption: Possibility results, impossibility results and the quest for a general definition. Cryptology ePrint Archive, Report 2012/515, 2012.

[BS05] Boaz Barak and Amit Sahai. How to play almost any mental game over the net - concurrent composition via super-polynomial simulation. In *FOCS*, pages 543–552, 2005.

[BSW11] Dan Boneh, Amit Sahai, and Brent Waters. Functional encryption: Definitions and challenges. In *TCC*, pages 253–273, 2011.

[BW06] Xavier Boyen and Brent Waters. Anonymous hierarchical identity-based encryption (without random oracles). In *CRYPTO*, pages 290–307, 2006.

[CHK03] Ran Canetti, Shai Halevi, and Jonathan Katz. A forward-secure public-key encryption scheme. In *EUROCRYPT*, pages 255–271, 2003.

[Coc01] Clifford Cocks. An identity based encryption scheme based on quadratic residues. In *IMA Int. Conf.*, pages 360–363, 2001.

[DKXY02] Yevgeniy Dodis, Jonathan Katz, Shouhuai Xu, and Moti Yung. Key-insulated public key cryptosystems. In *In EUROCRYPT*, pages 65–82. Springer-Verlag, 2002.

[FS90] Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *STOC*, pages 416–426, 1990.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986.

[GK05] Shafi Goldwasser and Yael Tauman Kalai. On the impossibility of obfuscation with auxiliary input. In *FOCS*, pages 553–562, 2005.

[GLW12] Shafi Goldwasser, Allison B. Lewko, and David A. Wilson. Bounded-collusion IBE from key homomorphism. In *TCC*, pages 564–581, 2012.

[GPSW06] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *ACM Conference on Computer and Communications Security*, pages 89–98, 2006.

[GVW12] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Functional encryption with bounded collusions from multiparty computation. In *CRYPTO*, 2012. To appear.

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

[KSW08] Jonathan Katz, Amit Sahai, and Brent Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In *EUROCRYPT*, pages 146–162, 2008.

[Lin04] Yehuda Lindell. Lower bounds and impossibility results for concurrent self composition. *the Journal of Cryptology*, 2004.

[LOS+10] Allison B. Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, and Brent Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *EUROCRYPT*, pages 62–91, 2010.

[MPR06] Silvio Micali, Rafael Pass, and Alon Rosen. Input-indistinguishable computation. In *FOCS*, pages 367–378, 2006.

[Nie02] Jesper Buus Nielsen. Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case. In *CRYPTO*, pages 111–126, 2002.

[O'N10] Adam O'Neill. Definitional issues in functional encryption. Cryptology ePrint Archive, Report 2010/556, 2010. http://eprint.iacr.org/.

[Pas03] Rafael Pass. Simulation in quasi-polynomial time and its application to protocol composition. In *EUROCRYPT*, pages 160–176, 2003.

[PRS02] Manoj Prabhakaran, Alon Rosen, and Amit Sahai. Concurrent zero knowledge with logarithmic round-complexity. In *In 43rd FOCS*, pages 366–375, 2002.

[PRV12] Bryan Parno, Mariana Raykova, and Vinod Vaikuntanathan. How to delegate and verify in public: Verifiable computation from attribute-based encryption. In Ronald Cramer, editor, *TCC*, volume 7194 of *Lecture Notes in Computer Science*, pages 422–439. Springer, 2012.

[PS04] Manoj Prabhakaran and Amit Sahai. New notions of security: achieving universal composability without trusted setup. In *STOC*, pages 242–251, 2004.

[Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO*, pages 47–53, 1984.

[SS10] Amit Sahai and Hakan Seyalioglu. Worry-free encryption: functional encryption with public keys. In *ACM Conference on Computer and Communications Security*, pages 463–472, 2010.

[SSW09] Emily Shen, Elaine Shi, and Brent Waters. Predicate privacy in encryption systems. In *TCC*, pages 457–473, 2009.

[SW] Amit Sahai and Brent Waters. Functional encryption:beyond public key cryptography. Power Point Presentation, 2008. http://userweb.cs.utexas.edu/bwaters/presentations/files/functional.ppt.

[SW05] Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *EUROCRYPT*, pages 457–473, 2005.

[Yao86] Andrew Chi-Chih Yao. How to generate and exchange secrets (extended abstract). In *FOCS*, pages 162–167, 1986.

A Indistinguishability-based Definition of Security

Definition A.1 (NA-IND- and AD-IND-Security)**.** *Let $\mathcal{FE}$ be a functional encryption scheme for a family of circuits $\mathcal{C}$. For every p.p.t. adversary $A = (A_1, A_2)$, consider the following two experiments:*

$\underline{\mathsf{Exp}^{(0)}_{\mathcal{FE},A}(1^\kappa)\textbf{:}}$

1: $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{FE.Setup}(1^\kappa)$
2: $(x_0, x_1, st) \leftarrow A_1^{\mathsf{FE.Keygen}(\mathsf{MSK},\cdot)}(\mathsf{MPK})$
3: $\mathsf{CT} \leftarrow \mathsf{FE.Enc}(\mathsf{MPK}, x_0)$
4: $b \leftarrow A_2^{\mathcal{O}(\mathsf{MSK},\cdot)}(\mathsf{MPK}, \mathsf{CT}, st)$
5: *Output* b

$\underline{\mathsf{Exp}^{(1)}_{\mathcal{FE},A}(1^\kappa)\textbf{:}}$

1: $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{FE.Setup}(1^\kappa)$
2: $(x_0, x_1, st) \leftarrow A_1^{\mathsf{FE.Keygen}(\mathsf{MSK},\cdot)}(\mathsf{MPK})$
3: $\mathsf{CT} \leftarrow \mathsf{FE.Enc}(\mathsf{MPK}, x_1)$
4: $b \leftarrow A_2^{\mathcal{O}(\mathsf{MSK},\cdot)}(\mathsf{MPK}, \mathsf{CT}, st)$
5: *Output* b

Define an *admissible adversary* $A = (A_1, A_2)$ as one such that for each oracle query C of A, $C(x_0) = C(x_1)$. *We distinguish between two cases of the above experiment:*

1. The adaptive experiment*, where the oracle* $\mathcal{O}(\mathsf{MSK}, \cdot) = \mathsf{FE.Keygen}(\mathsf{MSK}, \cdot)$*: the functional encryption scheme $\mathcal{FE}$ is said to be* indistinguishable-secure for one message against adaptive

adversaries (1-AD-IND-secure, for short) *if for every polynomial function* $\ell = \ell(\kappa)$ *and every* admissible *p.p.t. admissible adversary* $A = (A_1, A_2)$*, the advantage of* A *defined as below is negligible in the security parameter* κ*:*

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

where the probability is over the random coins of the algorithms of the scheme $\mathcal{FE}$ *and that of* A*.*

2. The non-adaptive experiment*, where the oracle* $\mathcal{O}(\mathsf{MSK}, \cdot)$ *is the "empty oracle" that returns nothing: the functional encryption scheme* $\mathcal{FE}$ *is said to be* indistinguishable-secure for one message against non-adaptive adversaries (1-NA-IND-secure, for short) *if for every* admissible *p.p.t. adversary* $A = (A_1, A_2)$*, the advantage of* A *defined as above is negligible in the security parameter* κ*.*

We do not distinguish between one and many message security since this definition composes [GVW12].

B Many-Message Simulation-based Definition of Security

Definition B.1 (many-NA-SIM- and many-AD-SIM- Security)**.** *Let* $\mathcal{FE}$ *be a functional encryption scheme for a family of circuit* $\mathcal{C}$*. Let* $U_x(\cdot)$ *denote the universal oracle that on input* C *returns* $U_x(C) = C(x)$*. Consider a p.p.t. adversary* $A = (A_1, A_2)$ *and a stateful p.p.t. simulator* Sim*. Consider the following two experiments:*

$\underline{\mathsf{Exp}^{\mathsf{real}}_{\mathcal{FE},A}(1^\kappa)}$**:**

1: $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{FE.Setup}(1^\kappa)$
2: $(\{x_i\}_{i\in[\ell]}, st) \leftarrow A_1^{\mathsf{FE.Keygen}(\mathsf{MSK},\cdot)}(\mathsf{MPK})$
3: $\mathsf{CT}_i \leftarrow \mathsf{FE.Enc}(\mathsf{MPK}, x_i)$ *for all* $i \in [\ell]$
4: $\alpha \leftarrow A_2^{\mathcal{O}(\mathsf{MSK},\cdot)}(\mathsf{MPK}, \{\mathsf{CT}_i\}_{i\in[\ell]}, st)$
5: *Output* $(\{x_i\}_{i\in[\ell]}, \alpha)$

$\underline{\mathsf{Exp}^{\mathsf{ideal}}_{\mathcal{FE},\mathrm{Sim}}(1^\kappa)}$**:**

1: $\mathsf{MPK} \leftarrow \mathrm{Sim}(1^\kappa)$
2: $(\{x_i\}_{i\in[\ell]}, st) \leftarrow A_1^{\mathrm{Sim}(\cdot)}(\mathsf{MPK})$
3: $\{\mathsf{CT}_i\}_{i\in[\ell]} \leftarrow \mathrm{Sim}^{\{U_{x_i}(\cdot)\}_{i\in[\ell]}}(\{1^{|x_i|}\}_{i\in[\ell]})$
4: $\alpha \leftarrow A_2^{\mathcal{O}'(\cdot)}(\mathsf{MPK}, \{\mathsf{CT}_i\}_{i\in[\ell]}, st)$
5: *Output* $(\{x_i\}_{i\in[\ell]}, \alpha)$

We distinguish between two cases of the above experiment:

1. The adaptive experiment*, where:*
   - *the oracle* $\mathcal{O}(\mathsf{MSK}, \cdot) = \mathsf{FE.Keygen}(\mathsf{MSK}, \cdot)$*, and*
   - *the oracle* $\mathcal{O}'(\cdot)$ *is the simulator, namely* $\mathrm{Sim}^{\{U_{x_i}(\cdot)\}_{i\in[\ell]}}(\cdot)$*.*

   We call a stateful simulator algorithm Sim admissible *if, on each input* C*,* Sim *makes just a single query to its oracle* $U_x(\cdot)$ *on* C *itself.*

   The functional encryption scheme $\mathcal{FE}$ *is then said to be* simulation-secure for many messages against adaptive adversaries (many-AD-SIM-secure, for short) *if there is an* admissible *stateful*

p.p.t. simulator Sim *such that for every polynomial function* $\ell = \ell(\kappa)$*, every p.p.t. adversary* $A = (A_1, A_2)$*, the following two distributions are computationally indistinguishable:*

$$\left\{ \mathsf{Exp}^{\mathsf{real}}_{\mathcal{FE},A}(1^\kappa) \right\}_{\kappa\in\mathbb{N}} \overset{c}{\approx} \left\{ \mathsf{Exp}^{\mathsf{ideal}}_{\mathcal{FE},\mathrm{Sim}}(1^\kappa) \right\}_{\kappa\in\mathbb{N}}$$

2. The non-adaptive experiment*, where the oracles* $\mathcal{O}(\mathsf{MSK},\cdot)$ *and* $\mathcal{O}'(\cdot)$ *are both the "empty oracles" that return nothing.*

   The functional encryption scheme $\mathcal{FE}$ *is then said to be simulation-secure for many messages against non-adaptive adversaries (many-*NA-SIM*-secure, for short) if there is an* admissible *stateful p.p.t. simulator* Sim *such that for every polynomial function* $\ell = \ell(\kappa)$*, every p.p.t. adversary* $A = (A_1, A_2)$*, the two distributions above are computationally indistinguishable.*

We define many-AD-USIM and many-NA-USIM identically to the above definition, except with computationally unbounded simulator.

C Indistinguishability and Unbounded Simulation

In this section, we put forth the notion of *unbounded-simulation secure* functional encryption. We argue that this is a very natural notion of security, with counterparts in the worlds of multiparty computation and zero knowledge [Pas03, PS04, BS05]. Unbounded simulation security further elucidates the power and limitations of indistinguishability-based definitions, and captures the spirit of some known separations (see below).

The definition of unbounded-simulation security (in both the adaptive and non-adaptive settings) is the same as Definition 2.2, except that the simulator is not restricted to run in polynomial time.

C.1 Relationship between SIM, USIM and IND

Theorem C.1. *Let* $\mathcal{FE}$ *be a functional encryption scheme for a family of circuits* $\mathcal{C}$*. Then, if* FE *is* AD-SIM *(resp.* NA-SIM*) secure then it is also* AD-USIM *(resp.* NA-USIM*) secure. In addition, if* FE *is* AD-USIM*(resp.* NA-USIM*) secure then it is* AD-IND *(resp.* NA-IND*) secure.*

Proof. AD-SIM(NA-SIM) $\Rightarrow$ AD-USIM(NA-USIM): This claim is trivial since the unbounded simulator can just run the poly-time simulator, in both the adaptive and non-adaptive cases.

AD-USIM(NA-USIM) $\Rightarrow$ AD-IND(NA-IND): We prove this claim by showing the contrapositive. Let $A = (A_1, A_2)$ be the adversary that breaks AD-IND(resp. NA-IND) security of the $\mathcal{FE}$ scheme. We construct an adversary $B = (B_1, B_2)$ from A. In our construction, if A is a AD-IND (resp. NA-IND) adversary, then B is a AD-USIM (resp. NA-USIM) adversary.

- $B_1^{\mathsf{FE.Keygen}(\mathsf{MSK},\cdot)}(\mathsf{MPK})$: Run $A_1^{\mathsf{FE.Keygen}(\mathsf{MSK},\cdot)}(\mathsf{MPK})$. Answer A_1's key queries using FE.Keygen oracle. Finally, A_1 outputs two messages x_0, x_1 and a state st_a. Now, choose a random bit b and output $(x, st) = (x_b, [st_a, x_0, x_1])$.
- $B_2^{\mathcal{O}(\mathsf{MSK},\cdot)}(\mathsf{MPK}, \mathsf{CT}, st)$: Invoke $A_2^{\mathcal{O}(\mathsf{MSK},\cdot)}(\mathsf{MPK}, \mathsf{CT}, st_a)$ and output whatever bit b' it outputs. In the adaptive case, if A_2 makes any oracle queries, answer them using own oracle.

The output of the real experiment is $(x_b, st = [st_a, x_0, x_1], \alpha = b', C_1, \ldots, C_q)$. Hence, with probability significantly greater than a half $b = b'$ and the distinguisher can verify this. Now we claim that there is no unbounded simulator Sim_u for the adversary (B_1, B_2) in the adaptive or non-adaptive case. In particular, we argue that the simulator Sim_u cannot guess bit b' with probability better than a half. We observe:

- The IND adversary $A = (A_1, A_2)$ is admissible, hence, for all queries $\{C'_i\}_{i\in[q']}$ that A makes to B, and hence B makes to Sim_u, we have that $C'_i(x_0) = C'_i(x_1)$. The view of Sim_u is statistically independent of the challenge bit b.
- Since the queries $C_1, \ldots, C_q$ are part of the output of the experiment, Sim_u is restricted to make admissible queries only to the function oracle, otherwise a distinguisher can easily distinguish between the real and ideal worlds. Thus, it must be the case that $C_i(x_0) = C_i(x_1)$, $\forall i \in [q]$. Hence, the view of Sim_u is also statistically independent of the challenge bit b.

Putting these together, we deduce that the view of the simulator is statistically independent of the challenge bit b, thus the probability it guesses b correctly is at most $1/2$. Therefore, the simulator Sim_u cannot produce the output indistinguishable from the real experiment.

□

C.2 Impossibility of AD-USIM IBE

We observe that the impossibility of realizing the IBE functionality under many-AD-SIM as shown in [BSW11, Section 5.1] (which is in turn similar to the impossibility result for non-committing encryption given in [Nie02]) extends to many-AD-USIM. For self containment, we provide a recap of the argument.

Let κ denote the security parameter and ℓ be an upper bound on the secret key length produced by FE.Keygen for security parameter κ. Assume that both the identity and the payload message are bits. Then, the real world adversary behaves as follows (refer to Definition 2.2) (1) A_1 makes no secret key queries and outputs $\mathbf{x} = \{(0, x_0, 0), \ldots, (0, x_{\ell+\kappa})\}$ where each x_i, $i \in [\ell + \kappa]$ is a uniformly random bit, and all payload messages correspond to identity 0. (2) The adaptive adversary A_2 requests for the SK_0.

Note that the admissible simulator Sim can only query the function oracle $U_{x_i}(\cdot)$ after A_2 requests the secret key SK_0, and thus cannot learn the payload bits until such point. However, Sim has to simulate the tuple of $\ell + \kappa$ ciphertexts before it sees the actual payload bits. Thus, the simulator has to produce a fixed string of CTs which, upon decryption with an ℓ bit secret key, need to produce an *arbitrary* $\ell + \kappa$ bits – which is impossible. This argument holds even if the simulator is computationally unbounded, hence we conclude that many-AD-USIM is impossible, even for IBE.

C.3 Illustrative Examples and Discussion

In this section we discuss some examples to separate the various notions of security.

AD-SIM and AD-USIM : We observe the following example from [BSW11, Section 4.2] separating AD-SIM and AD-IND also separates AD-SIM and AD-USIM.[8] Let π be a one-way permutation and consider the circuit family $\mathcal{C}$ consisting of a single circuit C defined as follows:

$$C(x) = \pi(x)$$

[8] We need to modify the construction to account for the fact that we do not admit the "empty" key.

Note that this circuit satisfies our "litmus test" in 1.2 for checking if IND security is inadequate — $\pi(x)$ hides x from a computationally bounded adversary, but reveals x to an unbounded adversary.

Now, consider the FE.Enc algorithm that outputs a public-key encryption of x on input x; FE.Keygen algorithm that outputs the secret key for the public-key encryption on input C; FE.Dec recovers x and outputs $\pi(x)$. It was shown in [BSW11] that this scheme AD-IND-secure but not AD-SIM-secure. We observe that this scheme is also AD-USIM-secure, provided the underlying encryption scheme is "non-committing" (c.f. [GVW12, Section 4.1]). The simulator proceeds as follows:

- If the adversary makes a secret key query before seeing the ciphertext, then the simulator learns $\pi(x)$ by querying the oracle on C, and then computes x via "brute force" and encrypts it.
- Otherwise, the simulator generates a "non-committing" simulated ciphertext. If the adversary makes a secret key query after seeing the ciphertext, then the simulator learns $\pi(x)$ by querying the oracle on C, computes x via "brute force", and generates a consistent secret key.

AD-IND and AD-USIM : We do not have a natural example of a scheme which is AD-IND-secure but not AD-USIM-secure, which we leave as an open problem. We point out that the naive approach of replacing the one-way permutation in the preceding example with a collision resistant hash function does not seem to work; if the encryptor picks the hash function, then the adversary could potentially pick a "bad" hash function for which it knows a pair of collisions.

Summary. We summarize some of the properties of USIM security: (1) It clarifies the inadequacy of indistinguishability-based security definitions as pointed out in [BSW11, Section 4.2]; specifically, the statement therein that "game-based formulation essentially ignores any computational hiding properties of the circuits, and therefore offers no security guarantees that could be meaningfully combined with such computational considerations", is really a statement about USIM security. (2) It potentially admits realizations for a large class of circuits; it allows us to circumvent our 1-NA-SIM lower bound, but is nonetheless ultimately limited by many-AD-SIM lower bound for IBE. (3) Unbounded simulation is a natural notion, with analogues in zero knowledge and secure computation [Pas03, PS04, BS05], and comparisons with these can aid our understanding of functional encryption. (4) For certain circuit families, USIM may be "good enough"; understanding when this happens could further clarify existing constructions.