Fully Secure Attribute-Based Encryption for t-CNF from LWE

Rotem Tsabary[*]

Abstract

Attribute-based Encryption (ABE), first introduced by [SW05, GPSW06], is a public key encryption system that can support multiple users with varying decryption permissions. One of the main properties of such schemes is the supported function class of policies. While there are fully secure constructions from bilinear maps for a fairly large class of policies, the situation with lattice-based constructions is less satisfactory and many efforts were made to close this gap. Prior to this work the only known fully secure lattice construction was for the class of point functions (also known as IBE).

In this work we construct for the first time a lattice-based (ciphertext-policy) ABE scheme for the function class t-CNF, which consists of CNF formulas where each clause depends on at most t bits of the input, for any constant t. This class includes NP-verification policies, bit-fixing policies and t-threshold policies. Towards this goal we also construct a fully secure single-key constrained PRF from OWF for the same function class, which might be of independent interest.

[*] Weizmann Institute of Science, `rotem.tsabary@weizmann.ac.il`. Supported by the Israel Science Foundation (Grant No. 468/14), Binational Science Foundation (Grants No. 2016726, 2014276), and by the European Union Horizon 2020 Research and Innovation Program via ERC Project REACT (Grant 756482) and via Project PROMETHEUS (Grant 780701).

1 Introduction

Atrribute-based Encryption (ABE), first introduced in [SW05, GPSW06], is a public key encryption system that can support multiple users with varying decryption permissions. In this work we focus on ciphertext-policy ABE schemes, where each ciphertext is associated with a public policy f and each decryption key is associated with a public attribute x, such that decryption succeeds conditioned on $f(x) = 1$. One of the main properties of an ABE scheme is the function class of policies that can be attached to ciphertexts. In fact, ABE was originally suggested as a generalization of identity-based encryption (IBE), in which each ciphertext is destined to a single attribute x (i.e. the policies are point functions).

Bilinear Maps Constructions. It was shown in a long line of works that bilinear maps prove to be useful for the task of constructing IBE and ABE under varying group assumptions. [BF03, Coc01] constructed the first IBE schemes in the random oracle model. [CHK03, BB04a] showed constructions in the standard model, however their security was proven under a weaker notion, called *selective security.*

A few approaches were suggested to go beyond selective security. [BB04b, Wat05] introduced the first constructions with full security in the standard model, using a *partitioning* technique. Their solutions were proved to be secure via a lossy reduction, where the simulator aborts with probability that grows exponentially with the number of keys owned by the adversary. [Gen06] introduced the *tagging* technique, with which he managed to construct a fully secure IBE scheme with a tight reduction, however the hardness assumption was still related to the number of keys. Finally, [Wat09] introduced the *dual system encryption* technique and achieved the first fully secure IBE scheme with a tight reduction to a fixed assumption.

The first ABE construction was suggested by [SW05] and supported threshold policies. Later, [GPSW06] constructed a key-policy[1] ABE scheme for policies that can be expressed as a linear secret-sharing (LSSS) access structure and [OSW07] constructed a key-policy ABE scheme for all formulas. [Wat11] showed a ciphertext-policy ABE construction for LSSS access structures. All of those works were proved to be secure in the weaker *selective* mode. The work of [LOS$^+$10, LW12] showed how to apply the dual system technique of [Wat09] to derived a fully secure ciphertext-policy[2] ABE for all LSSS access structures.

Lattice-Based Constructions. The emerging interest in hard problems over lattices, which are believed to be hard even at the presence of quantum machines, led to the development of a cryptographic toolbox [Ajt96, Ajt99, Reg05] that allows to base the security of various systems over random instances of such problems. This gave rise to a line of works about lattice-based IBE and ABE schemes. The first lattice-based IBE constructions were introduces by [**?**, CHKP12, ABB10a] and were secure in the *selective* model. Shortly after, [ABB10b] presented a construction with full security and [BL16] constructed a fully secure scheme with a tight reduction.

The first schemes to support richer classes of polices were [AFV11, ABV$^+$12], which constructed ABE for inner product policies and threshold policies respectively. [Boy13] showed key-policy ABE

[1] In *key-policy* ABE the policies are attached to the keys and the attributes are attached to the ciphertexts.

[2] [LW12] state that "Though we present only ciphertext-policy ABE schemes in this work, we expect that our techniques are equally applicable to the key-policy ABE setting", however to the best of our knowledge there are no explicit fully-secure key-policy ABE constructions up until today.

schemes for LSSS access structures. Lastly, the works of [GVW13, BGG^{+}14] constructed key-policy ABE for all policies that can be described by a bounded-depth polynomial-size circuit.

All of the aforementioned ABE constructions were proved to be selectively secure. The works of [BV16, GKW16] showed how to boost the security of [GVW13, BGG^{+}14] to an intermediate notion, named *semi-adaptive* security, however it is not clear how to further develop those techniques. The question of whether it is possible to construct fully-secure ABE schemes from lattices beyond point functions remained open.

Our Contribution. In this work we construct for the first time a lattice-based ciphertext-policy ABE scheme for the ensemble of function classes t-CNF, which consists of formulas in conjunctive normal form where each clause depends on at most t bits of the input, for any constant t. Our construction supports functions of unbounded size, that is, every function consisting of polynomial number of clauses. Those function classes includes NP-verification policies, bit-fixing policies and t-threshold policies. Towards this goal we also construct a fully secure single-key constrained PRF from OWF for the same function class, which might be of independent interest.

1.1 Technical Background

Let us first describe the difference between full security and selective security. The former is modeled as a game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ as follows. At the beginning of the game, $\mathcal{C}$ publishes the public parameters of the scheme. At any point of the game, $\mathcal{A}$ can query for multiple decryption keys to attributes x of its choice. In the challenge phase, $\mathcal{A}$ chooses a challenge policy f^* and $\mathcal{C}$ returns a ciphertext respective to f^*. The goal of $\mathcal{A}$ is to determine whether this is an encryption of 0 or 1, and the scheme is secure if it cannot do that as long as none of its queried keys x are authorized by f^*. The selective security game is identical, except that $\mathcal{A}$ has to announce the challenge policy f^* before the game begins.

In the latter game the security reduction has the opportunity to generate the public parameters according to f^*. Selective security proofs usually follow a similar structure, where f^* introduces a partitioning of the identity space. The public parameters are generated in the security reduction such that for all x for which $f^*(x) = 0$ (i.e. not authorized by f^*) it is possible to simulate a decryption key, and for all x for which $f^*(x) = 1$, a key for x would allow to break the hard problem. Since $\mathcal{A}$ can only query for keys of the first type, the reduction can still answer all of the queries appropriately.

Tagging. In [Gen06] Gentry presented an adaptively secure IBE scheme from bilinear maps, using a tagging technique as follows. In the real scheme, every ciphertext is associated with a random tag r_{ct} and every key is associated with a random tag r_{sk}. Decryption works as long as the IBE condition is satisfied *and* $r_{\mathsf{ct}} \neq r_{\mathsf{sk}}$. The probability that decryption fails is negligible since the tags are random. In the security proof, a random degree-Q polynomial P is embedded into the public parameters, such that it is possible to generate a challenge ciphertext respective to any x with the tag $r_{\mathsf{ct}} = P(x)$ and similarly it is possible to generate a key respective to any x with the tag $r_{\mathsf{sk}} = P(x)$. That is, the security reduction can answer any key query and can generate a challenge ciphertext respective to any x, however if it generates a ciphertext and a key for the same identity then the decryption fails because they both have the same tag. Recall that in the security game $\mathcal{A}$ is not allowed to query for a challenge and a key respective to the same attribute and therefore it cannot detect that case. Since P is a random polynomial, the values of P on up to Q points are distributed uniformly. For that reason security is guaranteed as long as $\mathcal{A}$ can only query for up to $Q-1$ keys. The evaluation of P has to be performed on a secret element in the exponent of a group. Since it is only possible to compute linear functions over the exponent, the reduction needs to get information that grows linearly with Q and makes the assumption stronger.

The BGG+ Lattice-Based Construction. A long sequence of works [ABB10b,MP12,GSW13, AP14,BGG$^+$14] led to a selectively secure key-policy ABE scheme with security based on LWE, for the function class of all policies that can be described as a bounded-depth polynomial-size circuit. We now give an overview of their technique.

The public parameters consist of a matrix $\mathbf{A}$, and for each attribute x (resp. policy f) there is a related efficiently computable matrix $\mathbf{A}_x \leftarrow \mathsf{EncodeX}(\mathbf{A}, x)$ (resp. $\mathbf{A}_f \leftarrow \mathsf{EncodeF}(\mathbf{A}, f)$). Encryption for an attribute x is a Dual-Regev encryption (see [GPV08]) respective to the public matrix $\mathbf{A}_x$, while a decryption key for f is a Dual-Regev key respective to the public matrix $\mathbf{A}_f$. The matrices $\mathbf{A}_x, \mathbf{A}_f$ are cleverly defined s.t., informally, for all x, f

$$f(x) = 1 \quad \longleftrightarrow \quad \begin{array}{l}\text{It is possible to convert a ciphertext respective to } \mathbf{A}_x \\ \text{to a ciphertext respective to } \mathbf{A}_f.\end{array}$$

Let $\mathsf{Convert}$ be the "ciphertext conversion algorithm" that satisfies the above condition, then we can informally say that

$$f(x) = 1 \quad \longleftrightarrow \quad \mathsf{Convert}(\mathbf{A}_x, x, f) = \mathbf{A}_f \ .$$

The property that is important to us, is that $\mathsf{Convert}$ works gate-by-gate and therefore respects function composition. That is, if $f = g_2 \circ g_1$, then for all x it holds that

$$\mathsf{Convert}(\mathbf{A}_x, x, f) = \mathsf{Convert}\left(\mathsf{Convert}(\mathbf{A}_x, x, g_1), g_1(x), g_2\right) \tag{1}$$

and therefore

$$f(x) = 1 \quad \longleftrightarrow \quad \mathsf{Convert}\left(\mathsf{Convert}(\mathbf{A}_x, x, g_1), g_1(x), g_2\right) = \mathbf{A}_f \ .$$

The security proof follows similar lines to other selectively-secure schemes as described at the beginning of this section. That is, the challenge attribute x^* is embedded into the public parameters $\mathbf{A}$ such that it is possible to create a challenge ciphertext only respective to $\mathbf{A}_{x^*} = \mathsf{EncodeX}(\mathbf{A}, x^*)$, and it is possible to generate keys only respective to $\mathbf{A}_f = \mathsf{EncodeF}(\mathbf{A}, f)$ for which $f(x^*) = 0$.

1.2 Our Techniques

Identity-Based Encryption. We first describe how to construct a fully secure IBE scheme with our approach. The main idea is to use the tagging technique of [Gen06], but with a PRF instead of a random polynomial. The rich function class supported by [BGG$^+$14] allows us to compute a PRF over a seed that is secretly embedded into the public parameters in the security proof. The tag of a key for an attribute x is the value of the PRF on the input x, i.e. r_x. That is, a key for x can decrypt any ciphertext respective to x unless the ciphertext tag is equivalent to r_x. In the real scheme the tags of ciphertexts are sampled uniformly, while in the security reduction they are determined by the PRF seed that is embedded into the public parameters. Details follow.

For all x we let U_x denote the circuit that on inuput σ evaluates the PRF on the point x with the seed σ. For all r we let $\bar{I}_r$ denote the circuit that on input r' returns 1 if and only if $r' \neq r$.

The public parameters of the IBE scheme are identical to [BGG+14] and the master secret key includes a PRF seed σ. To encrypt respective to x, one samples a fresh PRF seed σ' and computes the Dual-Regev encryption with the public matrix $\mathbf{A}'_x = \mathsf{Convert}(\mathbf{A}_{\sigma'}, \sigma', U_x)$ where $\mathbf{A}_{\sigma'} = \mathsf{EncodeX}(\mathbf{A}, \sigma')$. To generate a key respective to x, one first computes $r_x = U_x(\sigma)$ and then generates the Dual-Regev key respective to the matrix $\mathbf{A}_{f_x} = \mathsf{EncodeF}(\mathbf{A}, f_x)$, where $f_x = \bar{I}_{r_x} \circ U_x$. Note that $f_x(\sigma') = \bar{I}_{r_x}(U_x(\sigma'))$ where $r_x = U_x(\sigma)$. Therefore, if $\sigma = \sigma'$ then $f_x(\sigma) = 0$, but for any uniformly sampled σ', $U_x(\sigma') \neq U_x(\sigma)$ with high probability and therefore $f_x(\sigma') = 1$. That is, with high probability over a uniform σ' it holds that

$$f_x(\sigma') = 1 \quad \longleftrightarrow \quad \sigma' \neq \sigma$$

i.e.

$$\bar{I}_{r_x} \circ U_x(\sigma') = 1 \quad \longleftrightarrow \quad \sigma' \neq \sigma \ .$$

By the properties of [BGG+14] described above, it holds that

$$\bar{I}_{r_x} \circ U_x(\sigma') = 1 \quad \longleftrightarrow \quad \mathsf{Convert}\left(\mathsf{Convert}(\mathbf{A}_{\sigma'}, \sigma', U_x), U_x(\sigma'), \bar{I}_{r_x}\right) = \mathbf{A}_{f_x}$$

and therefore

$$\sigma' \neq \sigma \quad \longleftrightarrow \quad \mathsf{Convert}\left(\mathbf{A}'_x, U_x(\sigma'), \bar{I}_{r_x}\right) = \mathbf{A}_{f_x} \ .$$

That is, whenever $\sigma' \neq \sigma$ it is possible to convert a ciphertext respective to $\mathbf{A}'_x$ to a ciphertext respective to $\mathbf{A}_{f_x}$ and thus to decrypt. However, when $\sigma' = \sigma$ there is no such conversion algorithm.

In the security proof we encode σ in the public parameters, such that it is only possible to simulate Dual-Regev encryptions respective to matrices of the form $\mathbf{A}_x = \mathsf{Convert}(\mathbf{A}_\sigma, \sigma, U_x)$ (where $\mathbf{A}_\sigma = \mathsf{EncodeX}(\mathbf{A}, \sigma)$) but not respective to any other σ'. The indistinguishability relies on the pseudorandomness of the PRF and the properties of [BGG+14].

Expanding the Function Class. The main idea here is to replace the PRF with a constrained PRF. A constrained PRF, first defined in [BW13,KPTZ13,BGI14], allows the key owner to generate *constrained keys* σ_f respective to functions f, with which it is possible to compute the value of the PRF only on points x where $f(x) = 1$. More formally, there are two additional algorithms ($\mathsf{Constrain}$, $\mathsf{ConstrainEval}$) such that if $\sigma_f = \mathsf{Constrain}(\sigma, f)$, then for all x for which $f(x) = 1$ it holds that $\mathsf{ConstrainEval}(\sigma_f, f, x) = \mathsf{Eval}(\sigma, x)$, while for all x for which $f(x) = 0$, σ_f does not reveal information about $\mathsf{Eval}(\sigma, x)$.

Our construction uses a cPRF for policies in a function class $\mathcal{F}$ in order to construct an ABE scheme for policies in $\mathcal{F}$. The cPRF has to be single-key adaptively secure, and in addition it has to satisfy two properties as follows.

- *Gradual Evaluation* requires that for any f, x for which $f(x) = 1$, the circuit descriptions of the algorithms $\mathsf{Eval}(\cdot, x)$ and $\mathsf{ConstrainEval}(\mathsf{Constrain}(\cdot, f), f, x)$ are identical.
- *Key Simulation* requires an additional public algorithm $\mathsf{KeySim}(f) \rightarrow \sigma'_f$ that allows to simulate constrained keys. The keys should be indistinguishable from real constrained keys to a distinguisher with no access to evaluations on points x where $f(x) = 1$.

We call a cPRF that satisfies all of those properties a *conforming* cPRF. The ABE construction from a cPRF is a generalization of the IBE construction from a PRF. Details follow.

In the encryption algorithm, in order to encrypt respective to a policy f we compute a Dual-Regev encryption with the public matrix $\mathbf{A}'_f = \mathsf{Convert}(\mathbf{A}_{\sigma'}, \sigma', U_f)$, where $\mathbf{A}_{\sigma'} = \mathsf{EncodeX}(\mathbf{A}, \sigma')$ (as before) and U_f is the circuit description of $\mathsf{Constrain}(\cdot, f)$. The key generation algorithm remains the same. To decrypt with a key respective to x, one has to first convert the ciphertext to be respective to the matrix $\mathbf{A}'_x$. This is done by computing $\mathsf{Convert}(\mathbf{A}'_f, U_f(\sigma'), U_{f\to x})$, where $U_{f\to x}$ is the circuit description of $\mathsf{ConstrainEval}(\cdot, f, x)$. Note that

$$\begin{aligned}\mathsf{Convert}(\mathbf{A}'_f, U_f(\sigma'), U_{f\to x}) &= \mathsf{Convert}(\mathsf{Convert}(\mathbf{A}_{\sigma'}, \sigma', U_f), U_f(\sigma'), U_{f\to x}) \\ &= \mathsf{Convert}(\mathbf{A}_{\sigma'}, \sigma', U_x) \qquad (2) \\ &= \mathbf{A}'_x\end{aligned}$$

where the last equation holds by definition, and Equation (2) holds since $U_{f\to x} \circ U_f = U_x$ by the gradual evaluation property of the cPRF, and since $\mathsf{Convert}$ respects function composition as described in Equation (1).

The rest of the analysis is very similar to the IBE case. The key-simulation property guarantees that the adversary cannot tell whether the challenge ciphertext f^* is generated respective to σ or to a random σ', as long as it cannot query for evaluations of σ on points x where $f^*(x) = 1$ (which is indeed guaranteed by the ABE security game).

Constructing a Conforming cPRF. We construct a conforming cPRF for the function class t-CNF for any constant t. A policy f is in the class t-CNF if it can be described by a conjunctive normal form (CNF) formula, where each clause depends on t bits of the input. Our construction is inspired by the [DKNY18] construction of bit-fixing cPRF for a constant number of keys. In fact, their technique can be generalized to instantiate a family of cPRF schemes with a tradeoff between the "CNF locality" of the supported policies and the number of keys. They instantiate it with CNF locality 1 (i.e. bit-fixing) and t keys, while we instantiate it with CNF locality t and a single key. Details follow.

Let ℓ be the input length of the cPRF. We consider the set $S = \{(T, v)\}$ of all pairs (T, v) such that $T \subseteq [\ell]$, $|T| = t$, $v \in \{0,1\}^t$. For any input $x \in \{0,1\}^\ell$ we define the set $S_x = \{(T, x_T)\}_T$ where x_T is the substring of x on indices T. For all f we define the set $S_f \subseteq S$ of all of the pairs (T, v) that do not violate any of the clauses of f. It is easy to verify that for all x and f,

$$f(x) = 1 \quad \longleftrightarrow S_x \subseteq S_f \ . \qquad (3)$$

The master secret key is a key σ of a standard PRF. Evaluation on a point x returns the value r_x, computed as

$$r_x = \bigoplus_{(T,v)\in S_x} \mathsf{Eval}(\sigma_{(T,v)}, x) \qquad \text{where } \sigma_{(T,v)} = \mathsf{Eval}(\sigma, (T, v)) \ .$$

A constrained key for f consists of the values $\{\sigma_{(T,v)}\}_{(T,v)\in S_f}$. Correctness holds by Equation (3), security and key simulation holds by the pseudorandomness of the underlying PRF and gradual evaluation holds since the circuit $\mathsf{CPRF.Eval}(\cdot, x)$ is a sub-circuit of $\mathsf{CPRF.ConstrainEval}(\mathsf{Constrain}(\cdot, f), f, x)$.

1.3 Related Work

The idea to embed a PRF seed in a [BGG$^+$14]-like construction was previously suggested by [BV16, BL16].

Comparison with BV16. The work of [BV16] focuses on key-policy ABE with unbounded attribute length. In their scheme, the evaluation of the PRF allows to dynamically increase the width of the $\mathbf{A}$ matrix, so that $\mathbf{A}_x \leftarrow \mathsf{EncodeX}(\mathbf{A}, x)$ can be computed for x of varying length. In particular, the PRF is evaluated over values that only depend on the length of the attribute, where in our scheme the PRF is evaluated over the attribute value itself. Their ciphertexts contain two "pieces" for every bit of the attribute and they use an additional ABE scheme in a black-box manner in order to control the access that keys have to those pieces.

Their construction achieves *semi-adaptive* security, which means that the challenge attribute x^* has to be announced before the first key query, but possibly after seeing the public parameters. This property is due to the fact that in their cihpertexts the attribute value is implicitly XORed with a hidden random string Δ, that can be chosen in the security reduction at the first key generation. We note that if one desires a semi-adaptive scheme for a fixed attribute length ℓ, their technique can be instantiated with a PRG with $\text{poly}(\ell)$ stretch instead of a PRF. That is, the incentives for using a PRF are different in their work and ours.

Comparison with BL16. The work of [BL16] focuses on fully-secure signatures and IBE schemes with tight reductions. Their usage of a PRF in the IBE scheme has some similarities to an IBE instantiation of our approach, however the technicalities are different and the cPRF expansion is not applicable to their approach. They use a PRF with tight security that on input x outputs a single bit b_x. A ciphertext for an identity x contains two independent Dual-Regev encryptions of the message under two matrices $\mathbf{A}_{x,0}, \mathbf{A}_{x,1}$, and a key for x can only decrypt one of them $\mathbf{A}_{x,b_x}$. In the security proof the PRF seed is encoded into the public parameters such that it is possible to simulate keys for $\mathbf{A}_{x,b_x}$ without the master secret key, while it is only possible to simulate the "undecryptable" ciphertext part respective to $\mathbf{A}_{x,1-b_x}$.

1.4 Paper Organization

In Section 2 we go over the definitions of ABE and cPRF, and summarize lattice techniques from previous works. In section 3 we define the conforming cPRF and provide a construction for policies in t-CNF. In Section 4 we construct a fully secure ABE scheme that can be instantiated with any conforming cPRF.

2 Preliminaries

2.1 Constrained PRF, Attribute-Based Encryption, t-CNF Policies

Definition 2.1 ((Standard) PRF)**.** *A pseudo-random function family (PRF) is a pair of* PPT *algorithms* $(\mathsf{Setup}, \mathsf{Eval})$ *with the following syntax.* $\mathsf{Setup}(1^\lambda) \to \mathsf{sk}$ *takes as input a security parameter* λ *and outputs a secret key* sk*.* $\mathsf{Eval}_{\mathsf{sk}}(x) \to r_x$ *takes as input a secret key* sk *and a bit-string* $x \in \{0,1\}^\ell$*, and outputs a bit-sting* $r_x \in \{0,1\}^k$*.*

Pseudorandomness. *A PRF family is secure if for any* PPT *adversary* $\mathcal{A}$ *it holds that*

$$\left| Pr[\mathcal{A}^{\mathsf{Eval}_{\mathsf{sk}}(\cdot)}(1^\lambda) = 1] - Pr[\mathcal{A}^{\mathcal{O}(\cdot)}(1^\lambda) = 1] \right| = \text{negl}(\lambda)$$

where $\mathsf{sk} \leftarrow \mathsf{Setup}(1^\lambda)$ *and* $\mathcal{O}$ *is a random oracle.*

Definition 2.2 (Constrained PRF)**.** *Let $\mathcal{F}$ be a function class such that $\mathcal{F} \subseteq \{0,1\}^\ell \to \{0,1\}$. A constrained pseudo-random function (cPRF) for policies in $\mathcal{F}$ is a tuple of* PPT *algorithms with the following syntax.*

- $\mathsf{Setup}(1^\lambda) \to \mathsf{pp}, \mathsf{msk}$ *takes as input a security parameter λ and outputs public parameters* pp *along with a master secret key* msk.
- $\mathsf{Eval}_{\mathsf{msk}}(x) \to r_x$ *is a deterministic algorithm that takes as input a master secret key* msk *and a bit-string $x \in \{0,1\}^\ell$, and outputs a bit-sting $r_x \in \{0,1\}^k$.*
- $\mathsf{Constrain}_{\mathsf{msk}}(f) \to \mathsf{sk}_f$ *takes as input a master secret key* msk *and a function $f \in \mathcal{F}$, and outputs a constrained key* sk_f.
- $\mathsf{ConstrainEval}_{\mathsf{sk}_f}(x)$ *is a deterministic algorithm that takes as input a constrained key* sk_f *and a bit-string $x \in \{0,1\}^\ell$, and outputs a bit-string $r'_x \in \{0,1\}^k$.*

Correctenss. *A cPRF scheme is* correct *if for all $x \in \{0,1\}^\ell$ and $f \in \mathcal{F}$ for which $f(x) = 1$, it holds that* $\mathsf{Eval}_{\mathsf{msk}}(x) = \mathsf{ConstrainEval}_{\mathsf{sk}_f}(x)$ *where* $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ *and* $\mathsf{sk}_f \leftarrow \mathsf{Constrain}_{\mathsf{msk}}(f)$.

Pseudorandomness. *The adaptive security game of a cPRF scheme between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ is as follows.*

1. Initialization: *$\mathcal{C}$ generates* $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ *and sends* pp *to $\mathcal{A}$.*
2. Queries Phase I: *$\mathcal{A}$ makes (possibly many) queries in an arbitrary order:*
   - Evaluation Queries: *$\mathcal{A}$ sends a bit-string $x \in \{0,1\}^\ell$, $\mathcal{C}$ returns $r_x \leftarrow$* $\mathsf{Eval}_{\mathsf{msk}}(x)$.
   - Key Queries: *$\mathcal{A}$ sends a function $f \in \mathcal{F}$, $\mathcal{C}$ returns* $\mathsf{sk}_f \leftarrow \mathsf{Constrain}_{\mathsf{msk}}(f)$.
3. Challenge Phase: *$\mathcal{A}$ sends the challenge bit-string $x^* \in \{0,1\}^\ell$. $\mathcal{C}$ uniformly samples $b \overset{\$}{\leftarrow} \{0,1\}$. If $b = 0$ then $\mathcal{C}$ returns $r^* \overset{\$}{\leftarrow} \{0,1\}^k$. Otherwise it returns $r^* \leftarrow$* $\mathsf{Eval}_{\mathsf{msk}}(x^*)$.
4. Queries Phase II: *same as the first queries phase.*
5. End of Game: *$\mathcal{A}$ outputs a bit b'.*

$\mathcal{A}$ wins the game if (1) $b' = b$, (2) all of the evaluation queries are not for x^* and (3) all of the key queries f are such that $f(x^*) = 0$. The single-key *adaptive security game is as described above, except that $\mathcal{A}$ can only make a single key query throughout the entire game. A cPRF scheme is* secure *(resp.* single-key secure*) if for any* PPT *adversary $\mathcal{A}$, the probability that $\mathcal{A}$ wins in the adaptive (resp. single-key adaptive) security game is at most $1/2 + \mathrm{negl}(\lambda)$.*

Definition 2.3 (Attribute-Based Encryption)**.** *Let $\mathcal{F}$ be a function class such that $\mathcal{F} \subseteq \{0,1\}^\ell \to \{0,1\}$. A (ciphertext-policy) atrribute-based encryption (ctpABE) for policies in $\mathcal{F}$ is a tuple of* PPT *algorithms with the following syntax.*

- $\mathsf{Setup}(1^\lambda) \to \mathsf{pp}, \mathsf{msk}$ *takes as input a security parameter λ and outputs public parameters* pp *along with a master secret key* msk.

- $\mathsf{KeyGen}_{\mathsf{msk}}(x) \to \mathsf{sk}_x$ *takes as input a master secret key* msk *and a bit-string* $x \in \{0,1\}^\ell$*, and outputs a key* sk_x*.*
- $\mathsf{Enc}(f, \mu) \to \mathsf{ct}$ *takes as input a function* $f \in \mathcal{F}$ *and plaintext* $\mu \in \{0,1\}$*, and outputs a ciphertext* ct*.*
- $\mathsf{Dec}_{\mathsf{sk}_x}(\mathsf{ct}, f)$ *takes as input a key* sk_x*, a ciphertext* ct *and a function* $f \in \mathcal{F}$*, and outputs a bit* $\mu' \in \{0,1\}$*.*

Correctenss. *A ctpABE scheme is* correct *if for all* $x \in \{0,1\}^\ell$ *and* $f \in \mathcal{F}$ *for which* $f(x) = 1$*, and for all* $\mu \in \{0,1\}$*, it holds that*

$$Pr[\mathsf{Dec}_{\mathsf{sk}_x}(\mathsf{Enc}(f, \mu), f) \neq \mu] = \mathrm{negl}(\lambda)$$

where $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ *and* $\mathsf{sk}_x \leftarrow \mathsf{KeyGen}_{\mathsf{msk}}(x)$*.*

Security. *The adaptive security game of a ctpABE scheme between an adversary* $\mathcal{A}$ *and a challenger* $\mathcal{C}$ *is as follows.*

1. Initialization: $\mathcal{C}$ *generates* $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ *and sends* pp *to* $\mathcal{A}$*.*
2. Queries Phase I: $\mathcal{A}$ *makes (possibly many) key queries. For each query,* $\mathcal{A}$ *sends a string* $x \in \{0,1\}^\ell$ *and* $\mathcal{C}$ *returns* $\mathsf{sk}_x \leftarrow \mathsf{KeyGen}_{\mathsf{msk}}(x)$*.*
3. Challenge Phase: $\mathcal{A}$ *sends the challenge function* $f^* \in \mathcal{F}$*.* $\mathcal{C}$ *uniformly samples* $b \overset{\$}{\leftarrow} \{0,1\}$ *and returns* $\mathsf{ct}^* \leftarrow \mathsf{Enc}(f^*, b)$*.*
4. Queries Phase II: *same as the first queries phase.*
5. End of Game: $\mathcal{A}$ *outputs a bit* b'*.*

$\mathcal{A}$ *wins the game if (1)* $b' = b$ *and (2) all of the key queries* x *are such that* $f^*(x) = 0$*. A ctpABE scheme is* secure *if for any* PPT *adversary* $\mathcal{A}$*, the probability that* $\mathcal{A}$ *wins in the adaptive security game is at most* $1/2 + \mathrm{negl}(\lambda)$*.*

In this work we focus on the class of functions that can be described in a conjunctive normal form (CNF), where each clause is of constant locality. We give now a definition.

Definition 2.4 (t-CNF)**.** *A t-CNF policy* $f : \{0,1\}^\ell \to \{0,1\}$ *is a set of clauses* $f = \{(T_i, f_i)\}_i$*, where for all* i*,* $T_i \subseteq [\ell]$*,* $|T_i| = t$ *and* $f_i : \{0,1\}^t \to \{0,1\}$*. For all* $x \in \{0,1\}^\ell$ *the value of* $f(x)$ *is computed as*

$$f(x) = \bigwedge_i f_i(x_{T_i})$$

where x_T *is the length-t bit-string consisting of the bits of* x *in the indices* T*. A function class* $\mathcal{F}$ *is t-CNF if it consists only of t-CNF policies for some fixed* $\ell \in \mathbb{N}$ *and a* constant $t \leq \ell$*. If* $\mathcal{F}$ *is a t-CNF function class, we say that* t *is the* CNF locality *of* $\mathcal{F}$*.*

2.2 Lattice Trapdoors, Bounded Distributions, LWE

Lattice Trapdoors. Let $n, q \in \mathbb{Z}$, $\mathbf{g} = (1, 2, 4, \ldots, 2^{\lceil \log q \rceil - 1}) \in \mathbb{Z}_q^{\lceil \log q \rceil}$ and $m = n \lceil \log q \rceil$. The *gadget matrix* $\mathbf{G}$ is defined as the diagonal concatenation of $\mathbf{g}$ n times. Formally, $\mathbf{G} = \mathbf{g} \otimes \mathbf{I}_n \in \mathbb{Z}_q^{n \times m}$. For any $t \in \mathbb{Z}$, the function $\mathbf{G}^{-1} : \mathbb{Z}_q^{n \times t} \to \{0,1\}^{m \times t}$ expands each entry $a \in \mathbb{Z}_q$ of the input matrix into a column of size $\lceil \log q \rceil$ consisting of the bits representation of a. For any matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times t}$, it holds that $\mathbf{G} \cdot \mathbf{G}^{-1}(\mathbf{A}) = \mathbf{A}$.

The (centered) discrete Gaussian distribution over $\mathbb{Z}^m$ with parameter τ, denoted $D_{\mathbb{Z}^m, \tau}$, is the distribution over $\mathbb{Z}^m$ where for all $\mathbf{x}$, $\Pr[\mathbf{x}] \propto e^{-\pi \|\mathbf{x}\|^2 / \tau^2}$. Let $n, m, q \in \mathbb{N}$ and consider a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$. For all $\mathbf{v} \in \mathbb{Z}_q^n$ we let $\mathbf{A}_\tau^{-1}(\mathbf{v})$ denote the random variable whose distribution is the Discrete Gaussian $D_{\mathbb{Z}^m, \tau}$ conditioned on $\mathbf{A} \cdot \mathbf{A}_\tau^{-1}(\mathbf{v}) = \mathbf{v}$.

A τ-trapdoor for $\mathbf{A}$ is a procedure that can sample from a distribution within 2^{-n} statistical distance of $\mathbf{A}_\tau^{-1}(\mathbf{v})$ in time $\mathrm{poly}(n, m, \log q)$, for any $\mathbf{v} \in \mathbb{Z}_q^n$. We slightly overload notation and denote a τ-trapdoor for $\mathbf{A}$ by $\mathbf{A}_\tau^{-1}$. The following properties had been established in a long sequence of works.

Corollary 2.1 (Trapdoor Generation [Ajt96, MP12])**.** *There exists an efficiently computable value $m_0 = O(n \log q)$ and an efficient procedure $\mathsf{TrapGen}(1^n, q, m)$ such that for all $m \geq m_0$ outputs $(\mathbf{A}, \mathbf{A}_{\tau_0}^{-1})$, where $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ is 2^{-n}-uniform and $\tau_0 = O(\sqrt{n \log q \log n})$.*

We use the most general form of trapdoor extension as formalized in [MP12].

Theorem 2.2 (Trapdoor Extension [ABB10b, MP12])**.** *Given $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ with a trapdoor $\mathbf{A}_\tau^{-1}$, and letting $\mathbf{B} \in \mathbb{Z}_q^{n \times m'}$ be s.t. $\mathbf{A} = \mathbf{B}\mathbf{S} \pmod{q}$ where $\mathbf{S} \in \mathbb{Z}^{m' \times m}$ with largest singular value $s_1(\mathbf{S})$, then $(\mathbf{A}_\tau^{-1}, \mathbf{S})$ can be used to sample from $\mathbf{B}_{\tau'}^{-1}$ for any $\tau' \geq \tau \cdot s_1(\mathbf{S})$.*

A few additional important corollaries are derived from this theorem. We recall that $s_1(\mathbf{S}) \leq \sqrt{m'm}\, \|\mathbf{S}\|_\infty$ and that a trapdoor $\mathbf{G}_{O(1)}^{-1}$ is trivial. The first is a trapdoor extension that follows by taking $\mathbf{S} = [\mathbf{I}_{m'} \| \mathbf{0}_m]^T$.

Corollary 2.3. *Given $\mathbf{A} \in \mathbb{Z}_q^{n \times m'}$, with a trapdoor $\mathbf{A}_\tau^{-1}$, it is efficient to generate a trapdoor $[\mathbf{A} \| \mathbf{B}]_{\tau'}^{-1}$ for all $\mathbf{B} \in \mathbb{Z}_q^{n \times m}$, for any $m \in \mathbb{N}$ and any $\tau' \geq \tau$.*

Next is a trapdoor extension that had been used extensively in prior work. It follows from Theorem 2.2 with $\mathbf{S} = [-\mathbf{R}^T \| \mathbf{I}_m]^T$.

Corollary 2.4. *Given $\mathbf{A} \in \mathbb{Z}_q^{n \times m'}$, and $\mathbf{R} \in \mathbb{Z}^{m' \times m}$ with $m = n \lceil \log q \rceil$, it is efficient to compute $[\mathbf{A} \| \mathbf{A}\mathbf{R} + \mathbf{G}]_\tau^{-1}$ for $\tau = O(\sqrt{mm'}\, \|\mathbf{R}\|_\infty)$.*

Note that by taking $\mathbf{A}$ uniformly and $\mathbf{R}$ to be a high entropy small matrix, e.g. uniform in $\{-1, 0, 1\}$, and relying on the leftover hash lemma, Corollary 2.1 is in fact a special case of this one.

Lattice Evaluation. The following is an abstraction of the evaluation procedure in previous LWE based FHE and ABE schemes, that developed in a long sequence of works [ABB10b, MP12, GSW13, AP14, BGG$^+$14, GVW15].

Theorem 2.5. *There exist efficient deterministic algorithms* EvalF *and* EvalFX *such that for all* $n, q, \ell \in \mathbb{N}$ *and* $m = n\lceil \log q \rceil$, *for any depth* d *boolean circuit* $f : \{0,1\}^\ell \to \{0,1\}^k$ *and for every* $x \in \{0,1\}^\ell$, *for any matrix* $\mathbf{A} \in \mathbb{Z}_q^{n \times m \cdot \ell}$, *the outputs* $\mathbf{H} \leftarrow \mathsf{EvalF}(f, \mathbf{A})$ *and* $\widehat{\mathbf{H}} \leftarrow \mathsf{EvalFX}(f, x, \mathbf{A})$ *are both in* $\mathbb{Z}^{m \cdot \ell \times m \cdot k}$ *and it holds that* $\|\mathbf{H}\|_\infty , \left\|\widehat{\mathbf{H}}\right\|_\infty \leq (2m)^d$ *and*

$$[\mathbf{A} - x \otimes \mathbf{G}]\widehat{\mathbf{H}} = \mathbf{A}\mathbf{H} - f(x) \otimes \mathbf{G} \pmod q^{3} .$$

Moreover, for any pair of circuits $f : \{0,1\}^\ell \to \{0,1\}^k$, $g : \{0,1\}^k \to \{0,1\}^t$ *and for any matrix* $\mathbf{A} \in \mathbb{Z}_q^{n \times m \cdot \ell}$, *the outputs* $\mathbf{H}_f \leftarrow \mathsf{EvalF}(f, \mathbf{A})$, $\mathbf{H}_g \leftarrow \mathsf{EvalF}(g, \mathbf{A}\mathbf{H}_f)$ *and* $\mathbf{H}_{g \circ f} \leftarrow \mathsf{EvalF}(g \circ f, \mathbf{A})$ *satisfy* $\mathbf{H}_f \mathbf{H}_g = \mathbf{H}_{g \circ f}$.

Bounded Distributions. The following definitions and corollaries, taken from [BV16], will allow us to properly set the parameters of our scheme.

Definition 2.5. *A distribution* χ *supported over* $\mathbb{Z}$ *is* (B, ϵ)*-bounded if* $Pr_{x \overset{\$}{\leftarrow} \chi}[|x| > B] < \epsilon$.

Definition 2.6. *A distribution* $\tilde{\chi}$ *supported over* $\mathbb{Z}$ *is* (B, ϵ)*-swallowing if for all* $y \in [-B, B] \cap \mathbb{Z}$ *it holds that* $\tilde{\chi}$ *and* $y + \tilde{\chi}$ *are within* ϵ *statistical distance.*

Corollary 2.6. *For every* B, ϵ, δ *there exists an efficiently sampleable distribution that is both* (B, ϵ)*-swallowing and* $(B \cdot \sqrt{\log(1/\delta)}/\epsilon, O(\delta))$*-bounded.*

Definition 2.7. *A distribution* $\tilde{\chi}$ *supported over* $\mathbb{Z}$ *is* (χ, ϵ)*-swallowing, for a distribution* χ, *if it holds that* $\tilde{\chi}$ *and* $\chi + \tilde{\chi}$ *are within* ϵ *statistical distance. We omit the* ϵ *when it indicates a negligible function in a security parameter that is clear from the context.*

Corollary 2.7. *Let* $B(\lambda)$ *be some function and let* $\tilde{B}(\lambda) = B(\lambda) \cdot \lambda^{\omega(1)}$, *then there exists an efficiently sampleable ensemble* $\{\tilde{\chi}_\lambda\}_\lambda$ *such that* $\tilde{\chi}$ *is* χ*-swallowing for any* $B(\lambda)$*-bounded* $\{\chi_\lambda\}_\lambda$, *and also* $\tilde{B}(\lambda)$*-bounded.*

Learning With Errors. The *Learning with Errors* (LWE) problem was introduced by Regev [Reg05]. In this work we will use its decisional version.

Definition 2.8 (Decisional LWE (DLWE) [Reg05] and its HNF [ACPS09])**.** *Let* λ *be the security parameter,* $n = n(\lambda)$ *and* $q = q(\lambda)$ *be integers and let* $\chi = \chi(\lambda)$ *be a probability distribution over* $\mathbb{Z}$. *The* $\mathrm{DLWE}_{n,q,\chi}$ *problem states that for all* $m = \mathrm{poly}(n)$, *letting* $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$, $\mathbf{s} \leftarrow \mathbb{Z}_q^n$, $\mathbf{e} \leftarrow \chi^m$, *and* $\mathbf{u} \leftarrow \mathbb{Z}_q^m$, *it holds that* $\big(\mathbf{A}, \mathbf{s}\mathbf{A} + \mathbf{e}\big)$ *and* $\big(\mathbf{A}, \mathbf{u}\big)$ *are computationally indistinguishable.*

In this work we only consider the case where $q \leq 2^n$. Recall that GapSVP_γ is the (promise) problem of distinguishing, given a basis for a lattice and a parameter d, between the case where the lattice has a vector shorter than d, and the case where the lattice doesn't have any vector shorter than $\gamma \cdot d$. SIVP is the search problem of finding a set of "short" vectors. The best known algorithms for GapSVP_γ ([Sch87]) require at least $2^{\tilde{\Omega}(n/\log \gamma)}$ time. We refer the reader to [Reg05, Pei09] for more information. The following corollary allows us to appropriately choose the LWE parameters for our scheme according to known reductions from GapSVP_γ and SIVP_γ to $\mathrm{DLWE}_{n,q,\chi}$.

[3] For all $n \in \mathbb{Z}$ and $v \in \{0,1\}^n$ the term $v \otimes \mathbf{G}$ denotes a tensor product of the binary row-vector $v = (v_1, \ldots, v_n)$ and the matrix $\mathbf{G}$. That is, $v \otimes \mathbf{G} = [v_1 \cdot \mathbf{G} \| \ldots \| v_n \cdot \mathbf{G}]$.

Corollary 2.8 ([Reg05, Pei09, MM11, MP12, BLP+13]). *For all $\epsilon > 0$ there exists functions $q = q(n) \leq 2^n$, $\chi = \chi(n)$ such that χ is B-bounded for some $B = B(n)$, $q/B \geq 2^{n^\epsilon}$ and such that $\mathrm{DLWE}_{n,q,\chi}$ is at least as hard as the classical hardness of GapSVP_γ and the quantum hardness of SIVP_γ for $\gamma = 2^{\Omega(n^\epsilon)}$.*

3 Conforming cPRF

Our ABE construction in the next section instantiates a constrained PRF that has to satisfy some special properties, gathered under the following definition.

Definition 3.1 (Conforming cPRF). *A cPRF scheme is* conforming *if, in addition to the correctness and single-key addaptive security properties (see Definition 2.2), the following holds.*

Gradual Evaluation. *The algorithm $\mathsf{Constrain}$ (in addition to $\mathsf{Eval}, \mathsf{ConstrainEval}$) is deterministic and the following holds. For any fixing of $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda)$, $f \in \mathcal{F}$ and $x \in \{0,1\}^\ell$ for which $f(x) = 1$, define the following circuits:*

- *$U_{\sigma \to x} : \{0,1\}^\lambda \to \{0,1\}^k$ takes as input msk and computes $\mathsf{Eval}_{\mathsf{msk}}(x)$.*
- *$U_{\sigma \to f} : \{0,1\}^\lambda \to \{0,1\}^{\ell_f}$ takes as input msk and computes $\mathsf{Constrain}_{\mathsf{msk}}(f)$.*
- *$U_{f \to x} : \{0,1\}^{\ell_f} \to \{0,1\}^k$ takes as input sk_f and computes $\mathsf{ConstrainEval}_{\mathsf{sk}_f}(x)$.*

We require that for all pp, f, x as defined above, the circuit $U_{\sigma \to x}$ and the effective sub-circuit of $U_{f \to x} \circ U_{\sigma \to f}$ are the same. That is, the description of $U_{\sigma \to x}$ as a sequence of gates is identical to the sequence of gates that go from the input wires to the output wires of the circuit $U_{f \to x} \circ U_{\sigma \to f}$.

Key Simulation. *We require a PPT algorithm $\mathsf{KeySim}_{\mathsf{pp}}(f) \to \mathsf{sk}_f$ such that any PPT adversary $\mathcal{A}$ has at most $1/2 + \mathrm{negl}(\lambda)$ probability to win the following game against a challenger $\mathcal{C}$.*

- Initialization: *$\mathcal{C}$ generates $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and sends pp to $\mathcal{A}$.*
- Evaluation Queries I: *$\mathcal{A}$ makes (possibly multiple) queries. In each query it sends a bit-string $x \in \{0,1\}^\ell$ and $\mathcal{C}$ returns $r_x \leftarrow \mathsf{Eval}_{\mathsf{msk}}(x)$.*
- Challenge Phase: *$\mathcal{A}$ sends the challenge constraint $f^* \in \mathcal{F}$. $\mathcal{C}$ uniformly samples $b \overset{\$}{\leftarrow} \{0,1\}$. If $b = 0$ then $\mathcal{C}$ returns $\mathsf{sk}_{f^*} \leftarrow \mathsf{Constrain}_{\mathsf{msk}}(f)$, otherwise it returns $\mathsf{sk}_{f^*} \leftarrow \mathsf{KeySim}_{\mathsf{pp}}(f)$.*
- Evaluation Queries II: *same as the first queries phase.*
- End of Game: *$\mathcal{A}$ outputs a bit b'.*

$\mathcal{A}$ wins the game if (1) $b' = b$ and (2) all of the evaluation queries x are such that $f^*(x) = 0$.

Remark 3.1. *The requirement for a deterministic $\mathsf{Constrain}$ algorithm is for simplicity of exposition and since in our construction this requirement holds trivially. We note, however, that our ABE scheme can be extended to support a randomized $\mathsf{Constrain}$ algorithm. Alternatively, any cPRF scheme with a randomized $\mathsf{Constrain}$ algorithm can be converted to one with a deterministic algorithm by generating the randomness with an additional standard PRF.*

Example. The [GGM86] PRF is a conforming cPRF for prefix policies. Gradual evaluation holds since for any $x \in \{0,1\}^\ell$ and any length-t prefix $f \in \{0,1\}^t$, it holds that $U_{\sigma \to x}(\cdot) = G_{x_\ell}(\cdots G_{x_2}(G_{x_1}(\cdot)))$ while $U_{\sigma \to f}(\cdot) = G_{f_t}(\cdots G_{f_2}(G_{f_1}(\cdot)))$ and $U_{f \to x}(\cdot) = G_{x_\ell}(\cdots G_{x_{t+2}}(G_{x_{t+1}}(\cdot)))$. Key simulation holds since a constrained key for f is indistinguishable from uniform to any adversary that cannot query for evaluations on points accepted by f.

3.1 Construction for t-CNF Policies

We now describe our single key construction for the function class $\mathcal{F}$ consisting of CNF formulas where each clause depends on t bits of the input (see Definition 2.4). Our construction is inspired by the [DKNY18] construction of bit-fixing cPRF for a constant number of keys. In fact, their technique can be generalized to instantiate a family of cPRF schemes with a tradeoff between the *CNF locality* of the supported policies and the number of keys. They instantiate it with CNF locality 1 (i.e. bit-fixing) and t keys, while we instantiate it with CNF locality t and a single key.

Let $(\mathsf{P.Setup}, \mathsf{P.Eval})$ be a (standard) PRF (Definition 2.1), let $t \le \ell$ be a fixed constant and let S denote the set of all (T, v) pairs where $T \subseteq [\ell]$, $|T| = t$ and $v \in \{0,1\}^t$.

- $\mathsf{Setup}(1^\lambda)$: Sample and output $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{P.Setup}(1^\lambda)$.
- $\mathsf{Eval}(\mathsf{msk}, x)$: Let $S_x \subseteq S$ denote the set of all $(T, v) \in S$ pairs that "agree" with x, that is, $S_x = \{(T, x_T) \in S\}$ where x_T is the length-t bit-string consisting of the bits of x in the indices T. For all $(T, v) \in S_x$ compute $\mathsf{sk}_{T,v} \leftarrow \mathsf{P.Eval}_{\mathsf{msk}}(T \| v)$. Output
$$r_x = \bigoplus_{(T,v) \in S_x} \mathsf{P.Eval}_{\mathsf{sk}_{T,v}}(x) \ . \tag{4}$$
- $\mathsf{Constrain}_{\mathsf{msk}}(f)$: Parse f as a set of clauses $f = \{(T_i, f_i)\}$ and recall that for all i, $T_i \subseteq [\ell]$, $|T_i| = t$ and $f_i : \{0,1\}^t \to \{0,1\}$. For any clause $(T_i, f_i) \in f$ let $S_i^f \subseteq S$ be the set of all $(T, v) \in S$ pairs that "agree" with (T_i, f_i), that is,
$$S_i^f = \{(T_i, v) \in S : f_i(v) = 1\} \ .$$
Moreover, let $S_{rest}^f \subseteq S$ be the set of all $(T, v) \in S$ pairs such that f does not have a clause respective to T. That is,
$$S_{rest}^f = \{(T, v) \in S : \forall i \ T_i \neq T\} \ .$$
Finally let $S_f = S_{rest}^f \cup \bigcup_{(T_i, f_i) \in f} S_i^f$. For all $(T, v) \in S_f$ compute $\mathsf{sk}_{T,v} \leftarrow \mathsf{P.Eval}_{\mathsf{msk}}(T \| v)$. Output $\mathsf{sk}_f = \{\mathsf{sk}_{T,v}\}_{(T,v) \in S_f}$.
- $\mathsf{Eval}_{\mathsf{sk}_f}(x)$: If $f(x) = 0$ then abort, o.w. note that $S_x \subseteq S_f$ and compute r_x as in Eq. (4).

Correctness. Fix $x \in \{0,1\}^\ell$ and $f \in \mathcal{F}$ for which $f(x) = 1$. It is enough to prove that $S_x \subseteq S_f$. Note that $S_x = \{(T, x_T) \in S\}$ and parse $f = \{(T_i, f_i)\}$. For each $(T, x_T) \in S_x$ consider two options. If f has a clause respective to T, i.e. there exists i such that $T_i = T$, then since $f(x) = \bigwedge f_i(x_{T_i})$ and $f(x) = 1$, it also holds that $f_i(x_{T_i}) = 1$, and therefore $(T, x_T) = (T_i, x_{T_i}) \in S_i^f \subseteq S_f$. Otherwise, f does not have a clause respective to T, i.e. $\forall i \ T_i \neq T$, and therefore $(T, x_T) \in S_{rest}^f \subseteq S_f$.

Single-Key Adaptive Security. We sketch here the proof, which follows similar lines to [DKNY18]. Consider the single-key adaptive security game and let x^* and f be the challenge query and (single) key query respectively. It is guaranteed by the game that $f(x^*) = 0$, therefore there exists at least one clause $(T_i, f_i) \in f$ such that $f_i(x^*_{T_i}) = 0$ and therefore $(T_i, x^*_{T_i}) \notin S_f$.

In the simulated security game, the challenger guesses the value $(T_i, x^*_{T_i})$ at the beginning of the game by sampling a random pair $(T', v') \xleftarrow{\$} S$. When a key for f is queried, if there is no clause $(T_i, f_i) \in f$ such that $T_i = T'$ and $f_i(v') = 0$, then the challenger aborts. When a challenge for x^* is queried, if $x^*_{T'} \neq v'$ then the challenger aborts. Since there must exist an element $(T', v') \in S$ that does not cause an abort, and since (T', v') is chosen uniformly from S where $|S| = O((2\ell)^t)$, there is a significant probability $1/O((2\ell)^t)$ that the challenger does not abort when t is constant.

If the challenger does not abort, it replaces the element $\mathsf{Eval}_{\mathsf{sk}_{T',v'}}(x^*)$ in the challenge ciphertext with a uniform bit-string. This is indistinguishable by the pseudorandomness of P (respective to the key $\mathsf{sk}_{T',v'}$) and since the challenger does not have to provide $\mathsf{sk}_{T',v'}$ in the constrained key. At this point the challenge ciphertext is completely uniform, which completes the proof.

Gradual Evaluation. Fix $x \in \{0,1\}^\ell$ and $f \in \mathcal{F}$ for which $f(x) = 1$ and note that $S_x \subseteq S_f$. The circuit $U_{\sigma\to x}(\cdot)$ can be divided to two layers, where the first layer computes $\mathsf{sk}_x = \{sk_{T,x_T}\}_{(T,x_T)\in S_x}$ and the second layer computes r_x from sk_x. Moreover, letting $U^*_{f\to x} \circ U^*_{\sigma\to f}$ denote the effective sub-circuit of $U_{f\to x} \circ U_{\sigma\to f}$ (see Definition 3.1), it holds that $U^*_{\sigma\to f}$ (resp. $U^*_{f\to x}$) is exactly the first (resp. second) layer of $U_{\sigma\to x}(\cdot)$.

Key Simulation. The simulator $\mathsf{KeySim}(f)$ simply samples all of the components $\mathsf{sk}_f = \{\mathsf{sk}_{T,v}\}_{(T,v)\in S_f}$ uniformly. We sketch now the indistinguishability proof, which goes via a sequence of hybrids $\mathcal{H}_0, \ldots, \mathcal{H}_Q, \mathcal{H}_{Q+1}$ where Q is the number of evaluation queries made by $\mathcal{A}$. For $i = 0 \ldots Q$, in hybrid $\mathcal{H}_i$ the challenger answers the first i evaluation queries with uniformly sampled values and answers the challenge key query as in the real game. In hybrid $\mathcal{H}_{Q+1}$, the challenger answers all of the evaluation queries uniformly and answers the challenge key query with $\mathsf{KeySim}(f^*)$ regardless of the value of b. Note that hybrid $\mathcal{H}_0$ is identical to the key simulation game and that in hybrid $\mathcal{H}_{Q+1}$ the adversary wins the game with probability $1/2$. For all $i = 1 \ldots Q$, the indistinguishability of $\mathcal{H}_i$ and $\mathcal{H}_{i-1}$ follows from the single-key adaptive security of the scheme. Lastly, in hybrid $\mathcal{H}_Q$ the components of the key challenge $\mathsf{sk}_f = \{\mathsf{sk}_{T,v}\}_{(T,v)\in S_f}$ are either uniform (if $b = 1$) or from the distribution $\{\mathsf{sk}_{(T,v)} \leftarrow \mathsf{P.Eval}_{\mathsf{msk}}(T, v)\}_{(T,v)\in S_f}$ (if $b = 0$), while in $\mathcal{H}_{Q+1}$ they are always uniform. Those hybrids are indistinguishable by the pseudorandomness of P and since $|S_f| \in \mathrm{poly}(\lambda)$.

4 Fully Secure ABE from Conforming cPRF

4.1 The Construction

We now construct a ciphertext-policy ABE scheme for a function class $\mathcal{F}$ from a conforming cPRF (Definition 3.1) for $\mathcal{F}$. Our construction has adaptive security under the LWE assumption, and assuming that the underlying cPRF maintains single-key adaptive security.

Let $\mathsf{P} = (\mathsf{P.Setup}, \mathsf{P.Eval}, \mathsf{P.Constrain}, \mathsf{P.ConstrainEval})$ be a conforming cPRF for a class family $\mathcal{F}$ with input length ℓ and output length k. W.l.o.g. assume that the master secret key length of P is λ. For all $f \in \mathcal{F}$ let ℓ_f denote the size of a constrained key for the function f. Note that ℓ_f is constant and is efficiently computable given f and the description of $\mathsf{P.Constrain}$. Let $U_{\sigma\to x}$, $U_{\sigma\to f}$ and $U_{f\to x}$ be the circuits as in Definition 3.1. Define $\mathsf{ABE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{KeyGen}, \mathsf{Dec})$ as follows.

- $\mathsf{Setup}(1^\lambda)$: Sample $(\mathsf{P.msk}, \mathsf{P.pp}) \leftarrow \mathsf{P.Setup}(1^\lambda)$ and denote $\sigma = \mathsf{P.msk}$. Fix the parameters $n, q, m', \tau, \chi, \tilde{\chi}$ as explained below and let $m = n\lceil \log q \rceil$. Sample a matrix with its trapdoor $(\mathbf{B}, \mathbf{B}^{-1}_{\tau_0}) \leftarrow \mathsf{TrapGen}(1^n, m', q)$. Sample uniformly a matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m\cdot\lambda}$ and a vector $\mathbf{v} \xleftarrow{\$} \mathbb{Z}_q^n$. Output $\mathsf{pp} = (\mathbf{B}, \mathbf{A}, \mathbf{v}, \mathsf{P.pp})$ and $\mathsf{msk} = (\mathbf{B}^{-1}_{\tau_0}, \sigma)$.

- $\mathsf{Enc}_{\mathsf{pp}}(f, \mu)$: Sample $\mathsf{sk}_f \leftarrow \mathsf{P.KeySim}_{\mathsf{P.pp}}(f)$ and denote $s_f = \mathsf{sk}_f$. Sample $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$, $\mathbf{e}_0 \xleftarrow{\$} \chi^m$, $\mathbf{e}_1 \xleftarrow{\$} \tilde{\chi}^{m\cdot\ell_f}$, $e_2 \xleftarrow{\$} \chi$, and output $\mathsf{ct} = (s_f, \mathbf{u}_0, \mathbf{u}_1, u_2)$ such that

$$\mathbf{u}_0 = \mathbf{s}^T\mathbf{B} + \mathbf{e}_0^T\ , \qquad \mathbf{u}_1 = \mathbf{s}^T[\mathbf{A}_f - s_f \otimes \mathbf{G}] + \mathbf{e}_1^T\ , \qquad u_2 = \mathbf{s}^T\mathbf{v} + e_2 + \mu\lfloor q/2 \rceil\ ,$$

  where $\mathbf{A}_f = \mathbf{A}\mathbf{H}_{\sigma\to f}$ for $\mathbf{H}_{\sigma\to f} \leftarrow \mathsf{EvalF}(U_{\sigma\to f}, \mathbf{A})$.

- $\mathsf{KeyGen}_{\mathsf{msk}}(x)$: Compute the matrix $\mathbf{H}_{\sigma\to x} \leftarrow \mathsf{EvalF}(U_{\sigma\to x}, \mathbf{A})$ and denote $\mathbf{A}_x = \mathbf{A}\mathbf{H}_{\sigma\to x}$. Compute $r \leftarrow \mathsf{P.Eval}_\sigma(x)$ and let $I_r : \{0,1\}^k \to \{0,1\}$ be the function that on input r' returns 1 if and only if $r = r'$[4]. Compute $\mathbf{H}_r \leftarrow \mathsf{EvalF}(I_r, \mathbf{A}_x)$, denote $\mathbf{A}_{x,r} = \mathbf{A}_x\mathbf{H}_r$ and use $\mathbf{B}^{-1}_{\tau_0}$ to compute $[\mathbf{B}\|\mathbf{A}_{x,r}]^{-1}_\tau$. Sample $\mathbf{k} \leftarrow [\mathbf{B}\|\mathbf{A}_{x,r}]^{-1}_\tau(\mathbf{v})$ and output $\mathsf{sk}_x = (r, \mathbf{k})$.

- $\mathsf{Dec}_{\mathsf{sk}_x}(\mathsf{ct}, f)$: Parse $\mathsf{sk}_x = (r, \mathbf{k})$ and $\mathsf{ct} = (s_f, \mathbf{u}_0, \mathbf{u}_1, u_2)$. Compute $r' \leftarrow U_{f\to x}(s_f)$ and if $r = r'$ then abort. Otherwise, compute $\mathbf{A}_f$ and $\mathbf{A}_x$ as in $\mathsf{Enc}, \mathsf{KeyGen}$ respectively, then compute

$$\widehat{\mathbf{H}}_{s_f\to r'} \leftarrow \mathsf{EvalFX}(U_{f\to x}, s_f, \mathbf{A}_f) \qquad \text{and} \qquad \widehat{\mathbf{H}}_{r,r'} \leftarrow \mathsf{EvalFX}(I_r, r', \mathbf{A}_x)\ .$$

  Lastly, compute $u = u_2 - [\mathbf{u}_0\|\mathbf{u}_1\widehat{\mathbf{H}}_{s_f\to r'}\widehat{\mathbf{H}}_{r,r'}]\mathbf{k}$ and output 1 if and only if $|u| \geq q/4$.

Choice of Parameters. We set the parameters according to constraints that rise up in the security and correctness analysis. Choose $k = \lambda$, let $d = \mathrm{poly}(\lambda)$ denote the depth of $U_{\sigma\to x}$ and note that since P is gradual the depths of $U_{\sigma\to f}$, $U_{f\to x}$ are bounded by d. Choose $n \geq \lambda$ such that $(2n^2)^{2d+4} \leq 2^{n^\epsilon}$, where $\epsilon \in (0,1)$ is a security/efficiency tradeoff parameter. Note that $n \leq d^{O(1/\epsilon)}$ which is polynomial in λ for any constant ϵ. Moreover, $E' \leq 2^{n^\epsilon}$ where E' is as defined in Eq. (5). Choose q, B, χ according to Corollary 2.8 and note that $q/B \geq 2^{n^\epsilon}$ and that χ is B-bounded. Choose $m' = (n+1)\lceil \log q \rceil + 2\lambda$ and $\tau = \max\{\tau_0, \tau'\}$, where τ_0 is as in Corollary 2.1 and τ' is as in Eq. (6). Set $\tilde{\chi}$ to be a B'-swallowing distribution, where $B' = (m'+m)\lambda B(2m)^d$. By Corollary 2.7, $\tilde{\chi}$ can be chosen such that it is $\tilde{B}$-bounded for some $\tilde{B} \in O(B', \lambda)$.

4.2 Correctness

Lemma 4.1. *If* P *be a conforming cPRF for a class family* $\mathcal{F}$ *as per Definition 3.1, then* ABE *is a* correct *ciphertext policy attribute-based encryption scheme as per Definition 2.3 for the class family* $\mathcal{F}$.

[4] Previous works used an ABE definition where the decryption succeeds conditioned on $f(x) = 0$, while we require that $f(x) = 1$. Note that in our scheme the decryption succeeds conditioned on $f(x) = 1 \wedge r \neq r'$, i.e. $f(x) = 1 \wedge I_r(r') = 0$.

Proof. Fix $\mu \in \{0,1\}$, $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$, $f \in \mathcal{F}$ and $x \in \{0,1\}^\ell$ such that $f(x) = 0$. Consider $\mathsf{ct} \leftarrow \mathsf{Enc}(f, \mu)$ and $\mathsf{sk}_x \leftarrow \mathsf{KeyGen}_{\mathsf{msk}}(x)$, and parse $\mathsf{sk}_x = (r, \mathbf{k})$ and $\mathsf{ct} = (s_f, \mathbf{u}_0, \mathbf{u}_1, u_2)$. Consider the execution of $\mathsf{Dec}_{\mathsf{sk}_x}(\mathsf{ct}, f)$.

We first prove that with all but negligible probability $r \neq r'$ via a reduction to the pseudorandomness game of P. Recall that r' is computed as $\mathsf{P.ConstrainEval}_{\mathsf{sk}'_f}(x)$ where $\mathsf{sk}'_f \leftarrow \mathsf{P.KeySim}(f)$, while r is computed as $\mathsf{P.Eval}_{\mathsf{msk}}(x)$. Consider an adversary $\mathcal{A}$ in the pseudorandomness game of P as follows. Upon receiving $\mathsf{P.pp}$, it computes $\mathsf{sk}'_f \leftarrow \mathsf{P.KeySim}(f)$ and then $r'_x \leftarrow \mathsf{P.ConstrainEval}_{\mathsf{sk}'_f}(x)$. It then requests for a challenge on x, and upon receiving the challenge r^*_x it outputs 1 if and only if $r^*_x = r'_x$. The advantage of $\mathcal{A}$ is at least $Pr[r = r']$ and therefore if P is pseudorandom then $Pr[r = r']$ is negligible.

We now prove that if $r \neq r'$ then the decryption succeeds with all but negligible probability. Denote $\mathbf{H}_{f \to x} = \mathsf{EvalF}(U_{f \to x}, \mathbf{A}_f)$. Since P has gradual evaluation (see Definition 3.1), the effective sub-circuit of $U_{f \to x} \circ U_{\sigma \to f}$ and the circuit $U_{\sigma \to x}$ are identical. By Theorem 2.5 it follows that $\mathbf{H}_{\sigma \to f}\mathbf{H}_{f \to x} = \mathbf{H}_{\sigma \to x}$, and therefore $\mathbf{A}_f\mathbf{H}_{f \to x} = \mathbf{A}\mathbf{H}_{\sigma \to f}\mathbf{H}_{f \to x} = \mathbf{A}\mathbf{H}_{\sigma \to x} = \mathbf{A}_x$.

By applying Theorem 2.5 on $(\mathbf{H}_{f \to x}, \widehat{\mathbf{H}}_{s_f \to r'})$ and $(\mathbf{H}_r, \widehat{\mathbf{H}}_{r,r'})$, we get respectively

$$[\mathbf{A}_f - s_f \otimes \mathbf{G}]\,\widehat{\mathbf{H}}_{s_f \to r'} = \mathbf{A}_f\mathbf{H}_{f \to x} - U_{f \to x}(s_f) \otimes \mathbf{G} = \mathbf{A}_x - r' \otimes \mathbf{G}$$

and

$$\left[\mathbf{A}_x - r' \otimes \mathbf{G}\right]\widehat{\mathbf{H}}_{r,r'} = \mathbf{A}_x\mathbf{H}_r - I_r(r')\mathbf{G} = \mathbf{A}_{x,r}$$

where the last equation holds since $r \neq r'$ and thus $I_r(r') = 0$. Therefore,

$$\begin{aligned}
\mathbf{u}_1\widehat{\mathbf{H}}_{s_f \to r'}\widehat{\mathbf{H}}_{r,r'} &= \left(\mathbf{s}^T[\mathbf{A}_f - s_f \otimes \mathbf{G}] + \mathbf{e}_1^T\right)\widehat{\mathbf{H}}_{s_f \to r'}\widehat{\mathbf{H}}_{r,r'} \\
&= \mathbf{s}^T\,[\mathbf{A}_f - s_f \otimes \mathbf{G}]\,\widehat{\mathbf{H}}_{s_f \to r'}\widehat{\mathbf{H}}_{r,r'} + \mathbf{e}'_1 \qquad\qquad \text{where } \mathbf{e}'_1 = \mathbf{e}_1^T\widehat{\mathbf{H}}_{s_f \to r'}\widehat{\mathbf{H}}_{r,r'} \\
&= \mathbf{s}^T\left[\mathbf{A}_x - r' \otimes \mathbf{G}\right]\widehat{\mathbf{H}}_{r,r'} + \mathbf{e}'_1 \\
&= \mathbf{s}^T\mathbf{A}_{x,r} + \mathbf{e}'_1 \ .
\end{aligned}$$

Hence,

$$\begin{aligned}
u_2 - [\mathbf{u}_0\|\mathbf{u}_1\widehat{\mathbf{H}}_{s_f \to r'}\widehat{\mathbf{H}}_{r,r'}]\mathbf{k} &= \mathbf{s}^T\mathbf{v} + e_2 + \mu\lfloor q/2\rceil - \mathbf{s}^T[\mathbf{B}\|\mathbf{A}_{x,r}]\mathbf{k} - [\mathbf{e}_0^T\|\mathbf{e}'_1]\mathbf{k} \\
&= \mu\lfloor q/2\rceil + e_2 - [\mathbf{e}_0^T\|\mathbf{e}'_1]\mathbf{k} \ .
\end{aligned}$$

Note that

$$\left\|\mathbf{e}'_1\right\|_\infty \le m^2\ell_f k\left\|\mathbf{e}_1^T\right\|_\infty\left\|\widehat{\mathbf{H}}_{s_f \to r'}\right\|_\infty\left\|\widehat{\mathbf{H}}_{r,r'}\right\|_\infty \le m^2\ell_f k\tilde{B}(2m)^{d_{\mathsf{ConEv}}+1}$$

and that by the properties of discrete Gaussians, $\|\mathbf{k}\|_\infty \le \tau\sqrt{m'+m}$ with all but $2^{-(m'+m)} = \mathrm{negl}(\lambda)$ probability.

Therefore, if $m', k, \ell_f \in O(n, \lceil\log q\rceil)$, $\tilde{B} \in O(B, n)$ and $\tau \in O\left(k, \lambda, (2m)^{d+3}\right)$, then with all but negligible probability

$$\begin{aligned}
\left|e_2 - [\mathbf{e}_0^T\|\mathbf{e}'_1]\mathbf{k}\right| &\le |e_2| + (m'\left\|\mathbf{e}_0^T\right\|_\infty + m\left\|\mathbf{e}'_1\right\|_\infty)\cdot\|\mathbf{k}\|_\infty \\
&\le B + (m'B + m^3\ell_f k\tilde{B}(2m)^{d_{\mathsf{ConEv}}+1})\tau\sqrt{m'+m} \\
&\le B \cdot \mathrm{poly}(n, \lceil\log q\rceil)\cdot(2m)^{d_{\mathsf{ConEv}}+d+4} \ .
\end{aligned}$$

Denoting

$$E = B \cdot \mathrm{poly}(n, \lceil \log q \rceil) \cdot (2m)^{d_{\mathsf{ConEv}}+d+4}$$

and

$$E' = 4E/B = 4 \cdot \mathrm{poly}(n, \lceil \log q \rceil) \cdot (2m)^{d_{\mathsf{ConEv}}+d+4} \ , \tag{5}$$

by our choice of parameters E' is bounded by q/B, and therefore $E = BE'/4$ is bounded by $q/4$. Therefore, if $\mu = 0$ then $|u| \leq q/4$ and if $\mu = 1$ then $|u| > q/4$. □

4.3 Security

Lemma 4.2. *If* P *be a conforming cPRF for a class family* $\mathcal{F}$ *as per Definition 3.1, then* ABE *is a* secure *ciphertext policy attribute-based encryption scheme as per Definition 2.3 for the class family* $\mathcal{F}$ *under the* $\mathrm{DLWE}_{n,q,\chi}$ *assumption.*

Proof. We prove via a sequence of hybrids.

Hybrid $\mathcal{H}_0$. This is the adaptive security game from Definition 2.3.

Hybrid $\mathcal{H}_1$. We change the way $\mathcal{C}$ answers the challenge query f^*. Instead of computing $s_f \leftarrow \mathsf{P.KeySim}_{\mathsf{P.pp}}(f^*)$, it computes $s_f \leftarrow \mathsf{P.Constrain}_\sigma(f^*)$. Note that now $s_f = U_{\sigma \to f}(\sigma)$.

We show computational indistinguishability via a reduction to the key simulation game of P (see Definition 3.1). Let $\mathcal{A}_{\mathsf{P}}$ be an adversary in the key simulation game. It operates as the challenger in the ABE security game as follows. For every key query x sent by $\mathcal{A}$, $\mathcal{A}_{\mathsf{P}}$ queries the P challenger for an evaluation over the input x and proceeds with computing the ABE key for x as in the scheme. Note that it is guaranteed by the ABE game that $f^*(x) = 0$ and therefore this query is valid in the P game. When $\mathcal{A}$ asks for the challenge ciphertext, $\mathcal{A}_{\mathsf{P}}$ asks for the challenge constrained key sk'_f and proceeds with the encryption algorithm as in the scheme. Any advantage of $\mathcal{A}$ at distinguishing between those hybrids translates to identical advantage of $\mathcal{A}_{\mathsf{P}}$ in the key simulation game.

Hybrid $\mathcal{H}_2$. We change the way $\mathcal{C}$ generates the matrix $\mathbf{A}$ as follows. It samples uniformly a matrix $\mathbf{R} \xleftarrow{\$} \{0,1\}^{m' \times m \cdot \lambda}$ and sets $\mathbf{A} = \mathbf{BR} + \sigma \otimes \mathbf{G}$. Indistinguishability follows from the extended leftover hash lemma, since $m' \geq (n+1)\lceil \log q \rceil + 2\lambda$ and $\mathbf{B}$ is statistically-close to uniform by Corollary 2.1.

Hybrid $\mathcal{H}_3$. We change again the way $\mathcal{C}$ answers the challenge query f^*, specifically the way it generates $\mathbf{u}_1$. Note that now

$$\begin{aligned} \mathbf{A}_f - s_f \otimes \mathbf{G} &= \mathbf{A}\mathbf{H}_{\sigma \to f} - U_{\sigma \to f}(\sigma) \otimes \mathbf{G} \\ &= [\mathbf{A} - \sigma \otimes \mathbf{G}]\widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} \qquad \text{where } \widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} \leftarrow \mathsf{EvalFX}(U_{\sigma \to f}, \sigma, \mathbf{A}) \\ &= \mathbf{BR}\widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} \ . \end{aligned}$$

The values $\mathbf{u}_0$ and u_2 will be generated as before, by sampling $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$, $\mathbf{e_0} \xleftarrow{\$} \chi^m$, $e_2 \xleftarrow{\$} \chi$ and computing $\mathbf{u}_0 = \mathbf{s}^T\mathbf{B} + \mathbf{e}_0^T$ and $u_2 = \mathbf{s}^T\mathbf{v} + e_2 + \mu\lfloor q/2 \rceil$.

Recall that previously $\mathbf{u}_1$ was computed as $\mathbf{u}_1 = \mathbf{s}^T[\mathbf{A}_f - s_f \otimes \mathbf{G}] + \mathbf{e}_1^T$, where $\mathbf{e_1} \xleftarrow{\$} \tilde{\chi}^{m \cdot \ell_f}$. In this hybrid, it will be computed as $\mathbf{u}_1 = \mathbf{u}_0 \mathbf{R} \widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} + \mathbf{e}_1^T$. Note that now

$$\begin{aligned}
\mathbf{u}_1 &= \mathbf{u}_0 \mathbf{R} \widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} + \mathbf{e}_1^T \\
&= (\mathbf{s}^T \mathbf{B} + \mathbf{e}_0^T) \mathbf{R} \widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} + \mathbf{e}_1^T \\
&= \mathbf{s}^T[\mathbf{A}_f - s_f \otimes \mathbf{G}] + \mathbf{e}_0^T \mathbf{R} \widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} + \mathbf{e}_1^T
\end{aligned}$$

and that $B' = \left\| \mathbf{e}_0^T \mathbf{R} \widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} \right\|_\infty \le (m' + m)\lambda \left\| \mathbf{e}_0^T \right\|_\infty \|\mathbf{R}\|_\infty \left\| \widehat{\mathbf{H}}_{\mathsf{msk} \to s_f} \right\|_\infty \le (m' + m)\lambda B (2m)^{d_{\mathsf{Con}}}$, where d_{Con} is the depth of $U_{\sigma \to f}$. Therefore, if $\tilde{\chi}$ is B'-swallowing then this change is statistically indistinguishable.

Hybrid $\mathcal{H}_4$. We change the way $\mathcal{C}$ answers key queries. Let x be a query and fix $r \leftarrow \mathsf{P.Eval}_\sigma(x)$ and $\widehat{\mathbf{H}}_{\mathsf{msk} \to r} \leftarrow \mathsf{EvalFX}(U_{\sigma \to x}, \sigma, \mathbf{A})$. Note that

$$\begin{aligned}
[\mathbf{A} - \sigma \otimes \mathbf{G}] \widehat{\mathbf{H}}_{\mathsf{msk} \to r} &= \mathbf{A} \mathbf{H}_{\sigma \to x} - r \otimes \mathbf{G} \\
&= \mathbf{A}_x - r \otimes \mathbf{G} \qquad \text{where } \widehat{\mathbf{H}}_{\mathsf{msk} \to r} \leftarrow \mathsf{EvalFX}(U_{\sigma \to x}, \sigma, \mathbf{A}) \ ,
\end{aligned}$$

and since $I_r(r) = 1$,

$$[\mathbf{A}_x - r \otimes \mathbf{G}] \widehat{\mathbf{H}}_{r,r} = \mathbf{A}_x \mathbf{H}_r - I_r(r) \mathbf{G} = \mathbf{A}_{x,r} - \mathbf{G} \qquad \text{where } \widehat{\mathbf{H}}_{r,r} \leftarrow \mathsf{EvalFX}(I_r, r, \mathbf{A}_x) \ .$$

Therefore, since $\mathbf{A} - \sigma \otimes \mathbf{G} = \mathbf{B}\mathbf{R}$ it holds that $\mathbf{B}\mathbf{R}\widehat{\mathbf{H}}_{\mathsf{msk} \to r} \widehat{\mathbf{H}}_{r,r} = \mathbf{A}_{x,r} - \mathbf{G}$ and hence

$$[\mathbf{B} \| \mathbf{A}_{x,r}] = [\mathbf{B} \| \mathbf{B}\mathbf{R}\widehat{\mathbf{H}}_{\mathsf{msk} \to r} \widehat{\mathbf{H}}_{r,r} + \mathbf{G}] \ .$$

Note that

$$\begin{aligned}
\left\| \mathbf{R} \widehat{\mathbf{H}}_{\mathsf{msk} \to r} \widehat{\mathbf{H}}_{r,r} \right\|_\infty &\le m^2 k \lambda \|\mathbf{R}\|_\infty \left\| \widehat{\mathbf{H}}_{\mathsf{msk} \to r} \right\|_\infty \left\| \widehat{\mathbf{H}}_{r,r} \right\|_\infty \\
&\le m^2 k \lambda (2m)^{d+1} \ ,
\end{aligned}$$

and that Corollary 2.4, given $\mathbf{B}$ and $\mathbf{R}\widehat{\mathbf{H}}_{\mathsf{msk} \to r} \widehat{\mathbf{H}}_{r,r}$ it is efficient to compute $[\mathbf{B} \| \mathbf{A}_{x,r}]_{\tau'}^{-1}$ for some

$$\tau' = O\left(\left\| \mathbf{R} \widehat{\mathbf{H}}_{\mathsf{msk} \to r} \widehat{\mathbf{H}}_{r,r} \right\|_\infty \right) = O\left(k, \lambda, (2m)^{d+3} \right) \ . \tag{6}$$

Therefore, if $\tau \ge \tau'$ then $\mathcal{C}$ can now sample from $[\mathbf{B} \| \mathbf{A}_{x,r}]_\tau^{-1}(\mathbf{v})$ without $\mathbf{B}_{\tau_0}^{-1}$. The distribution remains identical to the previous hybrid.

Hybrid $\mathcal{H}_5$. We change the way $\mathbf{B}$ is generated. Instead of sampling it via $\mathsf{TrapGen}$, sample uniformly $\mathbf{B} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$. By Corollary 2.1 this change is statistically indistinguishable.

Hybrid $\mathcal{H}_6$. We change again the way $\mathcal{C}$ answers the challenge query. It now samples uniformly $\mathbf{u}_0 \xleftarrow{\$} \mathbb{Z}_q^{m'}$ and $u_2 \xleftarrow{\$} \mathbb{Z}_q$. This change is computationally indistinguishable under the $\mathrm{DLWE}_{n,q,\chi}$ assumption. At this step the challenge completely hides b and so $\mathcal{A}$ has no advantage. □

References

[ABB10a] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (H)IBE in the standard model. In *EUROCRYPT*, pages 553–572, 2010.

[ABB10b] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In *CRYPTO*, pages 98–115, 2010.

[ABV^{+}12] Shweta Agrawal, Xavier Boyen, Vinod Vaikuntanathan, Panagiotis Voulgaris, and Hoeteck Wee. Functional encryption for threshold functions (or fuzzy IBE) from lattices. In *Public Key Cryptography - PKC 2012 - 15th International Conference on Practice and Theory in Public Key Cryptography, Darmstadt, Germany, May 21-23, 2012. Proceedings*, pages 280–297, 2012.

[ACPS09] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *CRYPTO*, pages 595–618, 2009.

[AFV11] Shweta Agrawal, David Mandell Freeman, and Vinod Vaikuntanathan. Functional encryption for inner product predicates from learning with errors. In *Advances in Cryptology - ASIACRYPT 2011 - 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings*, pages 21–40, 2011.

[Ajt96] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *STOC*, pages 99–108, 1996.

[Ajt99] M. Ajtai. Generating hard instances of the short basis problem. In *ICALP*, 1999.

[AP14] Jacob Alperin-Sheriff and Chris Peikert. Faster bootstrapping with polynomial error. In *Advances in Cryptology - CRYPTO 2014*, pages 297–314, 2014.

[BB04a] Dan Boneh and Xavier Boyen. Efficient selective-id secure identity-based encryption without random oracles. In *Advances in Cryptology - EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004, Proceedings*, pages 223–238, 2004.

[BB04b] Dan Boneh and Xavier Boyen. Secure identity based encryption without random oracles. In *Advances in Cryptology - CRYPTO 2004, 24th Annual International CryptologyConference, Santa Barbara, California, USA, August 15-19, 2004, Proceedings*, pages 443–459, 2004.

[BF03] D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. *SIAM Journal on Computing*, 32(3):586–615, 2003. Preliminary version in *CRYPTO* '01.

[BGG^{+}14] Dan Boneh, Craig Gentry, Sergey Gorbunov, Shai Halevi, Valeria Nikolaenko, Gil Segev, Vinod Vaikuntanathan, and Dhinakaran Vinayagamurthy. Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In *EUROCRYPT*, pages 533–556, 2014.

[BGI14] Elette Boyle, Shafi Goldwasser, and Ioana Ivan. Functional signatures and pseudorandom functions. In *Public-Key Cryptography - PKC 2014 - 17th International Conference on Practice and Theory in Public-Key Cryptography, Buenos Aires, Argentina, March 26-28, 2014. Proceedings*, pages 501–519, 2014.

[BL16] Xavier Boyen and Qinyi Li. Towards tightly secure lattice short signature and id-based encryption. In *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part II*, pages 404–434, 2016.

[BLP$^+$13] Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In Boneh et al. [BRF13], pages 575–584.

[Boy13] X. Boyen. Attribute-based functional encryption on lattices. In *TCC*, 2013.

[BRF13] Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors. *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*. ACM, 2013.

[BV16] Zvika Brakerski and Vinod Vaikuntanathan. Circuit-abe from LWE: unbounded attributes and semi-adaptive security. In *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part III*, pages 363–384, 2016.

[BW13] D. Boneh and B. Waters. Constrained pseudorandom functions and their applications. In *ASIACRYPT*, 2013.

[CHK03] Ran Canetti, Shai Halevi, and Jonathan Katz. A forward-secure public-key encryption scheme. In *Advances in Cryptology - EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4-8, 2003, Proceedings*, pages 255–271, 2003.

[CHKP12] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. *J. Cryptology*, 25(4):601–639, 2012.

[Coc01] C. Cocks. An identity based encryption scheme based on quadratic residues. In *IMA Int. Conf.*, 2001.

[DKNY18] Alex Davidson, Shuichi Katsumata, Ryo Nishimaki, and Shota Yamada. Constrained prfs for bit-fixing from owfs with constant collusion resistance. *IACR Cryptology ePrint Archive*, 2018:982, 2018.

[Dwo08] Cynthia Dwork, editor. *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*. ACM, 2008.

[Gen06] Craig Gentry. Practical identity-based encryption without random oracles. In *Advances in Cryptology – EUROCRYPT '06*, pages 445–464, 2006.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986. Extended abstract in FOCS 84.

[GKW16] Rishab Goyal, Venkata Koppula, and Brent Waters. Semi-adaptive security and bundling functionalities made generic and easy. In *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part II*, pages 361–388, 2016.

[GPSW06] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In Ari Juels, Rebecca N. Wright, and Sabrina De Capitani di Vimercati, editors, *Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006, Alexandria, VA, USA, Ioctober 30 - November 3, 2006*, pages 89–98. ACM, 2006.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Dwork [Dwo08], pages 197–206.

[GSW13] Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In *Advances in Cryptology - CRYPTO 2013*, pages 75–92, 2013.

[GVW13] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Attribute-based encryption for circuits. In Boneh et al. [BRF13], pages 545–554.

[GVW15] Sergey Gorbunov, Vinod Vaikuntanathan, and Daniel Wichs. Leveled fully homomorphic signatures from standard lattices. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 469–477. ACM, 2015.

[KPTZ13] Aggelos Kiayias, Stavros Papadopoulos, Nikos Triandopoulos, and Thomas Zacharias. Delegatable pseudorandom functions and applications. In Ahmad-Reza Sadeghi, Virgil D. Gligor, and Moti Yung, editors, *2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13, Berlin, Germany, November 4-8, 2013*, pages 669–684. ACM, 2013.

[LOS$^+$10] A. B. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *EUROCRYPT*, pages 62–91, 2010.

[LW12] A. B. Lewko and B. Waters. New proof methods for attribute-based encryption: Achieving full security through selective techniques. In *CRYPTO*, 2012.

[MM11] Daniele Micciancio and Petros Mol. Pseudorandom knapsacks and the sample complexity of LWE search-to-decision reductions. In *Advances in Cryptology - CRYPTO 2011*, pages 465–484, 2011.

[MP12] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*, pages 700–718, 2012.

[OSW07] Rafail Ostrovsky, Amit Sahai, and Brent Waters. Attribute-based encryption with non-monotonic access structures. In *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA, October 28-31, 2007*, pages 195–203, 2007.

[Pei09] Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 333–342, 2009.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 84–93, 2005.

[Sch87] Claus-Peter Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theor. Comput. Sci.*, 53:201–224, 1987.

[SW05] A. Sahai and B. Waters. Fuzzy identity-based encryption. In *EUROCRYPT*, 2005.

[Wat05] Brent Waters. Efficient identity-based encryption without random oracles. In *Advances in Cryptology – EUROCRYPT '05*, pages 114–127, 2005.

[Wat09] Brent Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings*, volume 5677 of *Lecture Notes in Computer Science*, pages 619–636. Springer, 2009.

[Wat11] Brent Waters. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In Dario Catalano, Nelly Fazio, Rosario Gennaro, and Antonio Nicolosi, editors, *Public Key Cryptography - PKC 2011 - 14th International Conference on Practice and Theory in Public Key Cryptography, Taormina, Italy, March 6-9, 2011. Proceedings*, volume 6571 of *Lecture Notes in Computer Science*, pages 53–70. Springer, 2011.