FABEO: Fast Attribute-Based Encryption with Optimal Security

Doreen Riepel[1] and Hoeteck Wee[2]

[1] Ruhr-Universität Bochum, Germany
doreen.riepel@rub.de
[2] NTT Research, Sunnyvale, USA
wee@di.ens.fr

Abstract. Attribute-based encryption (ABE) enables fine-grained access control on encrypted data and has a large number of practical applications. This paper presents FABEO: faster pairing-based ciphertext-policy and key-policy ABE schemes that support expressive policies and put no restriction on policy type or attributes, and the first to achieve optimal, adaptive security with multiple challenge ciphertexts. We implement our schemes and demonstrate that they perform better than the state-of-the-art (Bethencourt et al. S&P 2007, Agrawal et al., CCS 2017 and Ambrona et al., CCS 2017) on all parameters of practical interest.

Keywords: attribute-based encryption; generic group model; tightness

1 Introduction

Attribute-based encryption (ABE) [45,29] extends classical public-key encryption to support fine-grained access control on encrypted data. ABE has applications in a variety of settings including electronic medical records [5], messaging systems [40], online social networks [9] and information-centric networking [34]. Companies like Cloudflare already use ABE to distribute private key storage across data centers [48].

ABE comes in two variants: ciphertext-policy (CP-ABE) and key-policy (KP-ABE), depending on whether access policies are attached to ciphertexts or to keys [12,29]. In CP-ABE, keys are associated with sets of attributes, and a key is able to recover the message hidden in a ciphertext if and only if the set of attributes satisfy the access policy attached to the ciphertext. For instance, a policy P could say '(Zipcode:90210 OR City:BeverlyHills) AND (AgeGroup:18-25)' and an individual A could have a key for Zipcode:90210, AgeGroup:Over65, in which case A would not be able to decrypt any message encrypted under P. A KP-ABE is the dual of CP-ABE with ciphertexts attached to attribute sets and keys associated with access policies.

There is by now a vast body of research on ABE realizing a broad spectrum of trade-offs between efficiency, expressiveness, security and hardness assumptions. The state of the art for practical ABE schemes are encapsulated by the following pairing-based schemes: (i) BSW CP-ABE scheme (Bethencourt, Sahai and Waters [12]), (ii) FAME CP-ABE and KP-ABE schemes (Agrawal and Chase [2]), and (iii) ABGW CP-ABE and KP-ABE schemes (Ambrona, Barthe, Gay and Wee [6]). These schemes simultaneously achieve the following properties that are highly desirable in practice:

(1) support expressive policies described by boolean formula and monotone span programs (MSP);
(2) put no restriction on size of policies or attribute sets;
(3) allow any arbitrary string such as street addresses to be used as an attribute;
(4) achieve the strong and natural notion of adaptive security, the defacto standard for ABE.

However, these schemes achieve incomparable efficiency guarantees, and deciding which one to deploy requires making complex performance trade-offs that depend on the policies that arise in the specific context.

Scheme	*Unrestricted policies*	*Arbitrary attributes*	*Fast decryption*	*Attribute multi-use*	*Security bounds*
CP-ABE					
BSW [12, §4.2] [2, §D]	✓	✓	×	✓	t^3/p
Waters [51, §3] [2, §E]	✓	×	×	✓	–
ABGW [6, §5.3]	✓	✓	×	✓	t^4/p
FAME [2, §3]	✓	✓	✓	×	t^4/p
Ours FABEO [Fig 1]	✓	✓	✓	✓	t^2/p
KP-ABE					
GPSW [29, §A.1] [2, §F]	✓	×	×	✓	–
ABGW [6, §5.3]	✓	✓	×	✓	t^4/p
FAME [2, §B]	✓	✓	✓	×	t^4/p
Ours FABEO [Fig 1]	✓	✓	✓	✓	t^2/p

Table 1: A property-wise comparison of the various ABE schemes we consider. The BSW, Waters and GPSW schemes were specified using symmetric pairings in the original works; throughout, we refer to the asymmetric variants from [2]. The last column shows the security bounds on the adversary's advantage in the multi-ciphertext setting for the adaptively secure schemes, with dashes indicating selectively secure schemes.

1.1 Our Contributions

We present FABEO, new pairing-based KP-ABE and CP-ABE schemes achieving properties (1) – (4), with improved efficiency and quantitatively stronger security guarantees. FABEO uses asymmetric (Type-III) prime-order bilinear groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ which support efficient hashing to $\mathbb{G}_1$ [46,23]. Ciphertexts and secret keys in FABEO comprise mostly of elements in the smaller and faster group $\mathbb{G}_1$, plus 1 or 2 elements in $\mathbb{G}_2$. Computation for key generation, encryption and decryption are mostly carried out in $\mathbb{G}_1$, with 2 to 3 pairings for decryption. We prove *optimal* security bounds for FABEO against adversaries that get an arbitrary number of ciphertexts and keys: in particular, when instantiated over the popular BLS12-381 curve, FABEO achieves close to 128-bit security.

FABEO subsumes BSW, FAME and ABGW on all parameters of practical interest. We improve upon the ciphertext and key sizes of all three schemes, as well as the running times. In particular, our ciphertexts are 66% smaller; encryption is (at least) 33% faster; and decryption uses fewer pairings. FABEO also supports multi-use of attributes like in BSW and ABGW (without an a-prior bound during set-up), with a small additive overhead in the multi-use parameter. See Table 1 for a property-wise comparison of our schemes against BSW, FAME, ABGW and other prominent schemes in the literature, as well as Tables 3 and 5 for a theoretical analysis and comparison for efficiency.

FABEO achieves properties (2) and (3) by hashing attributes to $\mathbb{G}_1$; smaller ciphertext/key sizes and fast decryption via randomness reuse (in CP-ABE ciphertexts and KP-ABE keys); and adaptive security without efficiency penalties by considering "generic" adversaries, a widely accepted model that captures all known attacks. While each of these techniques is already present in BSW, FAME, ABGW and prior works, FABEO is the first to combine them in a single design, along with a novel analysis establishing optimal security.

Optimal security. We prove security of our schemes in the generic bilinear group model (GGM) [14,47,42] (as with BSW and ABGW), where we model the underlying hash function as a random oracle [11] (as with BSW and FAME). We show that any generic, adaptive adversary running in time t and sees at most t ciphertexts and keys breaks our schemes with probability at most $O(t^2/p)$, where p is the order of the underlying group. This bound is optimal, since an adversary can break discrete log with the same probability. Prior ABE schemes, including BSW, FAME, and ABGW, achieve a bound of $O(t^3/p)$ or worse, since the security proofs only consider a single challenge ciphertext, and a hybrid argument is needed to achieve multi-ciphertext security.

Proof framework and application. In both our CP-ABE and KP-ABE schemes, the ciphertext ct for x and secret key sk for y are of the form:

$$\mathsf{ct} = \left(g_1^{c_x^1(\mathbf{s},\mathbf{b})}, g_2^{c_x^2(\mathbf{s})}, e(g_1,g_2)^{\alpha s_1} \cdot M \right), \mathsf{sk} = \left(g_1^{k_y^1(\alpha,\mathbf{b},\mathbf{r})}, g_2^{k_y^2(\mathbf{r})} \right).$$

Here, $\mathbf{s} = (s_1, \ldots)$ and $\mathbf{r}$ are fresh randomness; $g_1^{\mathbf{b}}$ contains the hash of every attribute in the universe[3], and $c_x^1, c_x^2, k_y^1, k_y^2$ are simple functions of degree 1 or 2. Roughly speaking, we show that for schemes of this form[4], security for a single ciphertext-key pair implies optimal, adaptive security against generic adversaries with an arbitrary number of ciphertexts and keys. Our modular proof framework extends and generalizes an analogous statement shown in ABGW in several ways: (i) we allow $c_x^2(\mathbf{s})$ to have arbitrary length instead of length 1, as is necessary to capture our CP-ABE scheme and the one below, (ii) we consider security with multiple ciphertexts, and (iii) we achieve optimal security.

Next, we describe an additional application of our proof framework that pertains to property (1). A limitation of boolean formula and monotone span programs is they do not capture computation over data of arbitrary, unbounded size, which arise settings such as genome sequencing, processing network and event logs, tax returns and virus scanners; such computation are better captured by regular languages, or deterministic finite automata (DFA). As a secondary contribution, we prove that Waters' KP-ABE scheme for DFA [52] achieves optimal, adaptive security.[5] In this scheme, $c_x^2(\mathbf{s})$ has arbitrary length that grows with x. Compared to prior adaptively secure KP-ABE for DFA [7,8,3,41,28], we obtain (at least) a 50% improvement in ciphertext and key sizes as well as running times.

Implementation and evaluation. We implement FABEO in the Charm framework [4]. Our experiments validate our theoretical analysis in Table 3 showing that FABEO improves on the performance of BSW, FAME and ABGW, for all of key generation, encryption and decryption. FABEO compares favorably even against the Waters CP-ABE [51] and GPSW KP-ABE [29], even though these schemes do not achieve property (3). See Figure 3 in Section 7 for the performance of the algorithms of each scheme under various test cases. Our code is available on GitHub [44].

All computations are performed on an ordinary laptop and we achieve practical results, even for large attribute sets and policies. Specifically for our CP-ABE with the MNT224 curve, set-up takes less than 0.02s, and it takes around 0.09s to generate a key for 100 attributes, and 0.18s to encrypt data under a policy that requires all 100 attributes. Decryption then takes only 0.02s. As a comparison, the ABGW CP-ABE scheme takes 0.63s to generate a key for the same number of attributes, 0.33s to encrypt and 0.48s to decrypt. In FAME, decryption takes 0.03s, and key generation and encryption are slower than ABGW.

Summary of Contributions. To summarize, our contributions are as follows:

- We present new KP-ABE and CP-ABE schemes for MSP with improved efficiency guarantees and the first to achieve optimal, adaptive security with multiple challenge ciphertexts.
- We provide a more general and modular framework for proving optimal ABE security in the GGM.
- We implement our KP-ABE and CP-ABE schemes for MSP and evaluate their performance for various parameters.
- We present and implement a new KP-ABE for DFA with optimal, adaptive security in the GGM.

1.2 Discussion and Related Work

We discuss additional context and related works.

Choosing curve parameters. When choosing curve parameters for a pairing-based scheme, practitioners often base the decisions on the hardness of the discrete log problem, and ignore the security bounds provided in security proof for the scheme. This is in part due to the limited number of pairing-friendly curves that are available in practice [46], and the possibly prohibitive performance penalty from using a curve with larger bit security. In particular, there is an implicit expectation that a scheme instantiated over a curve with 128-bit security should also achieve close to 128-bit security. Our work takes a step towards rigorously justifying this expectation in the context of pairing-based ABE.

[3] Ignoring for now the fact that $\mathbf{b}$ has exponential length.

[4] The ABGW CP-ABE and KP-ABE schemes we compare with are not of this form since the k_y^1 computes a rational function.

[5] Waters only proved weaker, selective security for his scheme. More precisely, we consider a variant of Waters' scheme with smaller keys from [27].

$\underline{\mathsf{Setup}(1^\lambda)}$. Run $\mathsf{GroupGen}(1^\lambda)$ to obtain $\mathcal{G} \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$. Pick $\alpha \xleftarrow{\$} \mathbb{Z}_p$ and a hash function $\mathsf{H} : [|\mathcal{U}|+1] \to \mathbb{G}_1$. Compute the master public key as

$$\mathsf{mpk} \coloneqq (\mathcal{G}, \mathsf{H}, e(g_1, g_2)^\alpha)$$

Let $\mathsf{msk} \coloneqq \alpha$ be the master secret key.

$\underline{\mathsf{KeyGen}(\mathsf{msk}, \mathcal{S} \subseteq \mathcal{U})}$. Pick $r \xleftarrow{\$} \mathbb{Z}_p$. Compute

$$\mathsf{sk}_1 \coloneqq g_1^\alpha \cdot \mathsf{H}(|\mathcal{U}|+1)^r \qquad \mathsf{sk}_{2,u} \coloneqq \mathsf{H}(u)^r \qquad \mathsf{sk}_3 \coloneqq g_2^r$$

for each $u \in \mathcal{S}$. Output $\mathsf{sk} \coloneqq (\mathsf{sk}_1, \{\mathsf{sk}_{2,u}\}_{u \in \mathcal{S}}, \mathsf{sk}_3)$.

$\underline{\mathsf{Enc}(\mathsf{mpk}, (\mathbf{M}, \pi))}$. Pick $s_1 \xleftarrow{\$} \mathbb{Z}_p$, $\mathbf{v} \xleftarrow{\$} \mathbb{Z}_p^{n_2-1}$, $\mathbf{s}' \xleftarrow{\$} \mathbb{Z}_p^\tau$. Compute

$$\mathsf{ct}_1 \coloneqq g_2^{s_1} \qquad \mathsf{ct}_{2,j} \coloneqq g_2^{\mathbf{s}'[j]}$$

for $j \in [\tau]$, as well as

$$\mathsf{ct}_{3,i} \coloneqq \mathsf{H}(|\mathcal{U}|+1)^{\mathbf{M}_i(s_1 \| \mathbf{v})^\top} \cdot \mathsf{H}(\pi(i))^{\mathbf{s}'[\rho(i)]}$$

for each row $i \in [n_1]$. Output $\mathsf{ct} \coloneqq (\mathsf{ct}_1, (\mathsf{ct}_{2,j})_{j \in [\tau]}, (\mathsf{ct}_{3,i})_{i \in [n_1]})$ and $d \coloneqq e(g_1, g_2)^{\alpha s_1}$.

$\underline{\mathsf{Dec}(\mathsf{mpk}, (\mathbf{M}, \pi), \mathcal{S}, \mathsf{ct}, \mathsf{sk})}$. If $\mathcal{S}$ satisfies $(\mathbf{M}, \pi)$, there exist constants $\{\gamma_i\}_{i \in I}$ s.t. $\sum_{i \in I} \gamma_i \mathbf{M}_i = (1, 0, \ldots, 0)$. Reconstruct d by computing

$$e(\mathsf{sk}_1, \mathsf{ct}_1) \cdot \frac{\prod_{j \in [\tau]} e(\prod_{i \in I, \rho(i)=j} (\mathsf{sk}_{2,\pi(i)})^{\gamma_i}, \mathsf{ct}_{2,j})}{e(\prod_{i \in I} (\mathsf{ct}_{3,i})^{\gamma_i}, \mathsf{sk}_3)}$$

and output the result.

$\underline{\mathsf{Setup}(1^\lambda)}$. Run $\mathsf{GroupGen}(1^\lambda)$ to obtain $\mathcal{G} \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$. Pick $\alpha \xleftarrow{\$} \mathbb{Z}_p$ and a hash function $\mathsf{H} : \mathcal{U} \to \mathbb{G}_1$. Compute the master public key as

$$\mathsf{mpk} \coloneqq (\mathcal{G}, \mathsf{H}, e(g_1, g_2)^\alpha)$$

Let $\mathsf{msk} \coloneqq \alpha$ be the master secret key.

$\underline{\mathsf{KeyGen}(\mathsf{msk}, (\mathbf{M}, \pi))}$. Pick $\mathbf{r}' \xleftarrow{\$} \mathbb{Z}_p^\tau$, $\mathbf{v} \xleftarrow{\$} \mathbb{Z}_p^{n_2-1}$. Compute

$$\mathsf{sk}_{1,j} \coloneqq g_2^{\mathbf{r}'[j]}$$

for $j \in [\tau]$, as well as

$$\mathsf{sk}_{2,i} \coloneqq g_1^{\mathbf{M}_i(\alpha \| \mathbf{v})^\top} \cdot \mathsf{H}(\pi(i))^{\mathbf{r}'[\rho(i)]}$$

for each row $i \in [n_1]$. Output $\mathsf{sk} \coloneqq ((\mathsf{sk}_{1,j})_{j \in [\tau]}, (\mathsf{sk}_{2,i})_{i \in [n_1]})$.

$\underline{\mathsf{Enc}(\mathsf{mpk}, \mathcal{S} \subseteq \mathcal{U})}$. Pick $s \xleftarrow{\$} \mathbb{Z}_p$. For each $u \in \mathcal{S}$ compute

$$\mathsf{ct}_{1,u} \coloneqq \mathsf{H}(u)^s \qquad \mathsf{ct}_2 \coloneqq g_2^s$$

Output $\mathsf{ct} \coloneqq ((\mathsf{ct}_{1,u})_{u \in \mathcal{S}}, \mathsf{ct}_2)$ and $d \coloneqq e(g_1, g_2)^{\alpha s}$.

$\underline{\mathsf{Dec}(\mathsf{mpk}, \mathcal{S}, (\mathbf{M}, \pi), \mathsf{ct}, \mathsf{sk})}$. If $\mathcal{S}$ satisfies $(\mathbf{M}, \pi)$, there exist constants $\{\gamma_i\}_{i \in I}$ s.t. $\sum_{i \in I} \gamma_i \mathbf{M}_i = (1, 0, \ldots, 0)$. Reconstruct d by computing

$$\frac{e(\prod_{i \in I} (\mathsf{sk}_{2,i})^{\gamma_i}, \mathsf{ct}_2)}{\prod_{j \in [\tau]} e(\prod_{i \in I, \rho(i)=j} (\mathsf{ct}_{1,\pi(i)})^{\gamma_i}, \mathsf{sk}_{1,j})}$$

and output the result.

Fig. 1: Our CP-ABE (left) and KP-ABE (right) scheme for monotone span programs ($\mathbf{M} \in \mathbb{Z}_p^{n_1 \times n_2}, \pi : [n_1] \to \mathcal{U}$). We define $\rho(i) \coloneqq |\{z \mid \pi(z) = \pi(i), z \leq i\}|$ and $\tau = \max_{i \in [n_1]} \rho(i)$ corresponding to maximum number of times an attribute is used in $\mathbf{M}$.

Achieving adaptive security. There are two main approaches for realizing adaptive security for ABE schemes in the literature: (1) prove security against generic adversaries as was done in BSW, ABGW and this work, and (2) adopt the dual system encryption framework [50,39,53,7] as used in FAME, which allows us to base security on SXDH and DLIN (and in some settings, with the additional use of q-type assumptions). While the latter yields theoretically stronger results, it incurs a huge penalty in efficiency: for security from k-LIN ($k = 1$ corresponds to SXDH and $k = 2$ to DLIN), it requires (at least) a factor $k + 1$ blow-up in ciphertext and key sizes as well as running times for encryption, key generation and decryption [16,8,3]. Moreover, the schemes have a more complex structure, and the security proofs are also substantially more complex. Another drawback is that the proofs typically require a hybrid argument over the keys and the ciphertexts, so we cannot hope for a security bound better than $O(t^4/p)$.

GGM security. We argue that GGM security is sufficient for most practical applications. The reasoning is two-fold: First, our understanding of pairing curves has advanced substantially over the past two decades, with increasing adoption (e.g. Cloudflare and ZCash) as well as on-going standardization [46]. The known attacks fall broadly into two categories: (1) attacks on discrete log, most notably the exTNFS in [35], rendering the curves unsuitable for any applications, (2) attacks that are captured by the GGM [19]. In short, there is in practice no discernible distinction between the standard assumptions like SXDH and GGM security. Second, it is much easier to break a real-world system via side channel attacks or poor security practices (e.g. phishing attacks or weak passwords) than to come up with an attack outside of the GGM. Indeed, a large number of recent works also use the GGM to analyze practical cryptosystems, e.g. [30,10,31].

Optimal and tight security. This work falls under a broader cryptographic research agenda of achieving optimal security and tight security reductions, for instance, recent works on symmetric-key encryption [32], signature schemes [21] and TLS 1.3 [25,22]. In the context of ABE, optimal security was only

previously known in very limited settings, namely identity-based encryption and its hierachical variant [24,18,13,33,26,37,15,17]; we clarify that these works focus on the more challenging goal of basing security on static assumptions such as DLIN. In particular, these are the only settings where we know how to carry out a dual system encryption proof with security bound better than $O(t^4/p)$.

Benchmarking ABE schemes. Two very recent works [43,20] looked into benchmarking pairing-based ABE schemes, focusing on low-level optimizations (whereas our work focuses on high-level design as well as new security guarantees and proof techniques): the first for CP-ABE covering BSW, Waters, and FAME but not ABGW, and the second for broadcast encryption. Both works highlight the complexity of effective benchmarking due to incomparable trade-offs between efficiency, expressiveness, security and hardness assumptions, which we alluded to at the beginning of the paper. Our results, together with those in ABGW and the preceding discussion, support the thesis that one should consider GGM-based schemes for benchmarking, since we can achieve the strongest notion of adaptive security without efficiency penalties.

1.3 Technical Overview

Let $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ be an (asymmetric) bilinear group of prime order p, along with a pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ and generators g_1, g_2 for $\mathbb{G}_1, \mathbb{G}_2$ respectively. In general, the bit sizes of group elements in $\mathbb{G}_2$ are 2-3 times that of $\mathbb{G}_1$ and group operations in $\mathbb{G}_2$ take (at least) twice as much time. In addition, we can securely hash into $\mathbb{G}_1$ at the cost of roughly one exponentiation in $\mathbb{G}_1$.

High-level design. We begin with a high-level overview of our KP-ABE scheme described in Figure 1. An MSP is given by a matrix $\mathbf{M}$ and a function π that maps each row of $\mathbf{M}$ to an attribute (for this overview, assume π is injective, i.e., no attribute multi-use). Following [29], we design the ciphertexts and secret keys so that for each row i in $\mathbf{M}$ such that $\pi(i)$ appears in the attribute set, decryption will compute

$$e(g_1, g_2)^{s_1 \alpha_i} \tag{1}$$

where α_i is a share of the master secret key α and $s_1 \leftarrow \mathbb{Z}_p$ is the encryption randomness. The values in (1) can then be combined to recover the blinding factor $e(g_1, g_2)^{s_1 \alpha}$.

To realize the above invariant, we have

$$(g_2^{s_1}, \mathsf{H}(\pi(i))^{s_1}) \in \mathsf{ct}, \quad (g_2^r, g_1^{\alpha_i} \mathsf{H}(\pi(i))^r) \in \mathsf{sk}$$

so that we can compute (1) using $e(g_1^{\alpha_i} \mathsf{H}(\pi(i))^r, g_2^{s_1}) / e(\mathsf{H}(\pi(i))^{s_1}, g_2^r)$. In addition, we use the same r across all the rows in $\mathbf{M}$, to keep the key size small. This way, we can also carry out decryption using two pairings[6]. and with most of the computation in the faster group $\mathbb{G}_1$. In contrast,

- BSW uses a different r_i for each share, namely $(g_2^{r_i},\ g_1^{\alpha_i} \mathsf{H}(\pi(i))^{r_i}) \in \mathsf{sk}$ (here, we are describing the KP-ABE analogue of the BSW CP-ABE). This incurs a factor 2 blow-up in key size, and decryption requires computing a pairing for each row of $\mathbf{M}$.
- ABGW uses
$$\left(g_1^{s - s_i}, g_1^{s_i (b_1 + \pi(i) b_2)}\right) \in \mathsf{ct}, \quad \left(g_2^{\frac{\alpha_i}{b_1 + \pi(i) b_2}}, g_2^{\alpha_i}\right) \in \mathsf{sk}$$
where $g_1^{b_1}, g_1^{b_2}$ comes from mpk. This incurs (at least) a factor 2 blow-up in ciphertext and key sizes, and an extra exponentation per attribute during encryption. Decryption requires computing a pairing for each attribute.
- FAME replaces $g_2^{s_1}, g_2^r$ with DLIN-tuples in $\mathbb{G}_2^3$ in order to achieve security under the DLIN assumption using the dual system encryption framework as described in Section 1.2. Overall, this incurs a factor 3 blow-up in ciphertext and key sizes, as well as a factor 3-6 blow-up in running time for encryption and key generation.

[6] by writing $\prod_i (e(g_1, g_2)^{s_1 \alpha_i})^{\gamma_i}$ as $e(\prod_i (g_1^{\alpha_i} \mathsf{H}(\pi(i))^r)^{\gamma_i}, g_2^{s_1}) \cdot e(\prod_i (\mathsf{H}(\pi(i))^{s_1})^{\gamma_i}, g_2^r)^{-1}$

Our CP-ABE scheme is conceptually the dual of our KP-ABE, though algebraically more intricate and less intuitive (the same holds for BSW, FAME, and ABGW). Briefly, instead of (1), decryption computes $e(g_1, g_2)^{\mu_i b' r}$ where μ_i is a share of the encryption randomness s_1; $g_1^{b'}$ is specified in the public key; and r comes from key generation randomness. These values can then be combined to compute $e(g_1, g_2)^{s_1 b' r}$, which is in turn used to recover the blinding factor $e(g_1, g_2)^{s_1 \alpha}$. Our CP-ABE scheme is the same as the AC17-LU-OK and AC17-LU-CP schemes in the independent work [43], which asserts selective security under q-type assumptions without a formal security proof.

Proof strategy. We provide a unified proof security of our KP-ABE and CP-ABE schemes in the GGM, where we model H as a random oracle. At a high level, we follow the framework in [6]. Both our KP-ABE and CP-ABE schemes have the following structure where the ciphertext is associated with a label x and the secret key with a label y (x is an attribute set for KP-ABE and a policy for CP-ABE, and vice-versa for y)

$$\mathsf{ct}_x = \left(g_1^{c_x^1(\mathbf{s} \otimes \mathbf{b})}, g_2^{c_x^2(\mathbf{s})}, e(g_1, g_2)^{\alpha s_1} \cdot M \right), \ \mathsf{sk}_y = \left(g_1^{k_y^1(\alpha, \mathbf{r}, \mathbf{b} \otimes \mathbf{r})}, g_2^{k_y^2(\mathbf{r})} \right),$$

where[7]

- $\mathbf{s} = (s_1, \ldots)$ and $\mathbf{r}$ are random vectors over $\mathbb{Z}_p$ corresponding to randomnness for encryption and key generation;
- $g_1^{\mathbf{b}}$ contains the hash of every attribute in the universe, along with $g_1^{b'}$ for our CP-ABE (note that the length of $\mathbf{b}$ is exponential, but the c_x^1, k_y^1 only depend on a polynomial number of entries of $\mathbf{b}$);
- $c_x^1, c_x^2, k_y^1, k_y^2$ are linear functions over $\mathbb{Z}_p$ (therefore c_x^1 and k_y^1 computes degree 2 functions of $\mathbf{s}, \mathbf{b}, \mathbf{r}, \alpha$);
- decryption uses the pairing to compute $e(g_1, g_2)^{c_x^1(\mathbf{s} \otimes \mathbf{b}) \otimes k_y^2(\mathbf{r})}$ and $e(g_1, g_2)^{k_y^1(\alpha, \mathbf{r}, \mathbf{b} \otimes \mathbf{r}) \otimes c_x^2(\mathbf{s})}$, followed by additional linear computation in the exponent to recover the blinding factor $e(g_1, g_2)^{\alpha s_1}$.

We refer to ABE schemes with the above structure as a PES-ABE (PES is short for pair encoding schemes [7]). Towards proving GGM security, we consider notions of symbolic security for PES-ABE, where an adversary sees abstract expressions for group elements in the form of polynomials. The proof of security of our KP-ABE and CP-ABE schemes follows the following modular framework:

Step 1. We show that our KP-ABE and CP-ABE schemes satisfy the syntax of a PES-ABE and (1,1) symbolic security, a relaxation of ABE security where the adversary is selective[8] and only receives a single ciphertext and single secret key.

Step 2. We prove that any PES-ABE satisfying (1,1) symbolic security also satisfies strong symbolic security, where the adversary is still selective but can see the public key as well as an arbitrary number of ciphertexts and secret keys.

Step 3. We prove that any PES-ABE satisfying strong symbolic security is adaptively secure in the GGM with optimal security.

We now describe the key differences between our framework and the one in ABGW:

- The syntax for PES-ABE is different: (i) both ct_x and sk_y contain elements from both $\mathbb{G}_1$ and $\mathbb{G}_2$, and (ii) we generate $g_1^{\mathbf{b}}$ using a random oracle.
- We introduce a strengthening of (1,1) symbolic security where we essentially require that all of $[\alpha c_x^1(\mathbf{s})]_T$ are pseudorandom, and not just $[\alpha s_1]_T$. Our notion is also weaker in that the proof only needs to reason about the terms $e(g_1, g_2)^{c_x^1(\mathbf{s} \otimes \mathbf{b}) \otimes k_y^2(\mathbf{r})}$ and $e(g_1, g_2)^{k_y^1(\alpha, \mathbf{r}, \mathbf{b} \otimes \mathbf{r}) \otimes c_x^2(\mathbf{s})}$.
- The ABGW KP-ABE and CP-ABE schemes for MSP do not satisfy the syntax of a PES-ABE since k_y^2 computes rational functions with linear functions $\mathbf{b}$ in the denominator and therefore proving security of these schemes require directly establishing strong symbolic security;

[7] The tensor product $\mathbf{u} \otimes \mathbf{v}$ of two vectors $\mathbf{u} = (u_1, u_2, \ldots)$ and $\mathbf{v} = (v_1, v_2, \ldots)$ is a vector $(u_1 v_1, u_1 v_2, \ldots)$ containing all pairwise products of the entries in u and v.

[8] In this overview, we use selective to refer to an adversary that specifies all of its ciphertext and key queries in advance.

– The analogue of strong symbolic securty in ABGW in Steps 2 and 3 considers only a single challenge ciphertext.

– We achieve a security bound of $O(t^2/p)$ in Step 3, whereas ABGW achieves $O(t^3/p)$. Our proof crucially relies on the fact that $c_x^1, c_x^2, k_y^1, k_y^2$ compute functions of degree at most 2 in the inputs so that we only need to apply Schwartz-Zippel to constant-degree polynomials. The proof in ABGW applies Schwartz-Zippel to polynomials of degree t in order to "clear the denominators" across t keys.

2 Preliminaries

We will first fix some notation that we will use throughout the paper. For integers m, n where $m < n$, $[m, n]$ denotes the set $m, m+1, ..., n$. For $m = 1$, we simply write $[n]$. For a prime p, let $\mathbb{Z}_p$ denote the set $[0, p-1]$, where addition and multiplication are computed modulo p. For a set $\mathcal{S}$, $s \xleftarrow{\$} \mathcal{S}$ denotes that s is sampled uniformly and independently at random from $\mathcal{S}$. $y \leftarrow \mathcal{A}(x_1, x_2, \ldots)$ denotes that on input $x_1, x_2, \ldots$ the probabilistic algorithm $\mathcal{A}$ returns y. $\mathcal{A}^{\mathcal{O}}$ denotes that algorithm $\mathcal{A}$ has access to oracle $\mathcal{O}$. An adversary is a probabilistic algorithm. A probabilistic algorithm is called *efficient* or PPT if its running time is bounded by some polynomial in the length of its input.

We use lower case bold-face letters for row vectors, where $\|$ denotes concatenation of row vectors. $\mathbf{v}[i]$ denotes the i-th coordinate of the vector $\mathbf{v}$. Given a vector $\mathbf{v}$ of polynomials of length m over $\mathbb{Z}_p$, we write $\mathrm{span}(\mathbf{v})$ to denote $\{\mathbf{v} \cdot \mathbf{e}^\top : \mathbf{e} \in \mathbb{Z}_p^m\}$. Formal variables are marked with a tilde. We write $\tilde{\mathbf{v}} \leftarrow \mathsf{Var}^n$ to pick n formal variables.

2.1 Pairing Groups

Let $\mathsf{GroupGen}$ be a PPT algorithm that takes a security parameter 1^λ as input and returns a group description $\mathcal{G} := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$, where p is a prime of $\Theta(\lambda)$ bits, $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ are cyclic groups of order p, $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is a non-degenerate bilinear map (also called pairing) and g_1 resp. g_2 or generators of $\mathbb{G}_1$ resp. $\mathbb{G}_2$. The generator g_T of $\mathbb{G}_T$ can be computed as $e(g_1, g_2)$. We require that the group operations in $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ and the bilinear map e are computable in deterministic polynomial time in λ. In this work, we only consider asymmetric (or Type-III) pairing groups where there exists no efficiently computable homomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$. In some cases we will use *implicit representation* of group elements: for a vector $\mathbf{v}$ over $\mathbb{Z}_p$, we define $[\mathbf{v}]_1 := g_s^{\mathbf{v}}$ for $s \in \{1, 2, T\}$, where exponentiation is carried out component-wise.

2.2 Attribute-based Encryption

Throughout the paper, we will use a KEM-style definition of ABE. However note that it is implied by the corresponding definition in the PKE setting.

Syntax. An attribute-based encryption (ABE) scheme for some class P consists of four algorithms:

$\mathsf{Setup}(1^\lambda, \mathsf{P}) \to (\mathsf{mpk}, \mathsf{msk})$. The setup algorithm gets as input the security parameter 1^λ and class description P. It outputs the master public key mpk and the master secret key msk. We assume mpk defines the key space $\mathcal{K}$.

$\mathsf{Enc}(\mathsf{mpk}, x) \to (\mathsf{ct}_x, d)$. The encryption algorithm gets as input mpk and an input x. It outputs a ciphertext ct_x and an encapsulated key $d \in \mathcal{K}$.

$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, y) \to \mathsf{sk}_y$. The key generation algorithm gets as input mpk, msk and $y \in \mathsf{P}$. It outputs a secret key sk_y.

$\mathsf{Dec}(\mathsf{mpk}, x, y, \mathsf{ct}_x, \mathsf{sk}_y) \to m$. The decryption algorithm gets as input sk_y and ct_x such that $\mathsf{P}(x, y) = 1$ along with mpk. It outputs a key d.

Correctness. For all input x and y with $\mathsf{P}(x) = 1$, we require

$$\Pr\left[\mathsf{Dec}(\mathsf{mpk}, x, y, \mathsf{ct}_x, \mathsf{sk}_y) = d : \begin{array}{c}(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{P}) \\ \mathsf{sk}_y \leftarrow \mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, y) \\ (\mathsf{ct}_x, d) \leftarrow \mathsf{Enc}(\mathsf{mpk}, x)\end{array}\right] = 1.$$

Many-Ciphertext CPA Security. We define security by a game between a challenger and an adversary $\mathcal{A}$. The challenger picks a random challenge bit β and provides the following oracles to $\mathcal{A}$.

- Setup oracle $\mathcal{O}_{\mathsf{mpk}}$: This oracle can only be queried once and it must be the first query. The challenger runs Setup to obtain $(\mathsf{msk}, \mathsf{mpk})$ and outputs mpk to $\mathcal{A}$.
- Ciphertext (or challenge) oracle $\mathcal{O}_{\mathsf{ct}}$: On the i-th query, $\mathcal{A}$ provides $x_i \in \mathcal{X}$. The challenger runs $(\mathsf{ct}_i, d_i^{(0)}) \leftarrow \mathsf{Enc}(\mathsf{mpk}, x_i)$, chooses a random key $d_i^{(1)} \xleftarrow{\$} \mathcal{K}$ and outputs $(\mathsf{ct}_i, d_i^{(\beta)})$.
- Secret key oracle $\mathcal{O}_{\mathsf{sk}}$: On the j-th query, $\mathcal{A}$ provides $y_j \in \mathcal{Y}$. The challenger runs $\mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, y_j)$ and outputs sk_j.

$\mathcal{O}_{\mathsf{ct}}$ and $\mathcal{O}_{\mathsf{sk}}$ can be queried adaptively and an arbitrary polynomial number of times. Finally, $\mathcal{A}$ outputs a bit β'. We say that $\mathcal{A}$ wins the game if $\beta = \beta'$ and $\mathsf{P}(x_i, y_j) = 0$ for all queries x_i and y_j.

Definition 1. *An* ABE *scheme is adaptively many-ciphertext secure if for all efficient* $\mathcal{A}$*,*

$$\mathsf{Adv}_{\mathsf{ABE},\mathcal{A}}(\lambda) \coloneqq \left|\Pr[\beta = \beta'] - \tfrac{1}{2}\right|$$

is negligible in λ*.*

Boolean formulae and MSP Boolean formulae are a common way to model access control. A (monotone) boolean formula consists of **and** and **or** gates, where each input is associated with an attribute in the universe of attributes denoted by $\mathcal{U}$. Monotone means that an authorized user who acquires more attributes will not lose any privileges. Let $\mathcal{S} \subseteq \mathcal{U}$ be a set of attributes. We say that $\mathcal{S}$ satisfies a boolean formula if we set all inputs of the formula that map to an attribute in $\mathcal{S}$ to true and the others to false and the formula evaluates to true.

Monotone span programs (MSP) are a more general class of functions and include boolean formulae. We encode an access structure by a policy $(\mathbf{M}, \pi)$, where $\mathbf{M} \in \mathbb{Z}_p^{n_1 \times n_2}$ and $\pi : [n_1] \to \mathcal{U}$. Note that we can compute $(\mathbf{M}, \pi)$ for any (monotone) boolean formula in polynomial time [38]. Then every row $\mathbf{M}_i$ corresponds to an input to the formula and the number of columns is the same as the number of **and** gates. If the mapping π is not injective, we use the notation $\rho(i) \coloneqq |\{z \mid \pi(z) = \pi(i), z \leq i\}|$ to denote the $\rho(i)$-th occurrence of attribute $\pi(i)$.

Let $\mathcal{S} \subseteq \mathcal{U}$ be a set of attributes and $I = \{i \mid i \in [n_1], \pi(i) \in \mathcal{S}\}$ be the indices of rows in $\mathbf{M}$ that are associated with $\mathcal{S}$. We say that $(\mathbf{M}, \pi)$ accepts $\mathcal{S}$ if the vector $(1, 0, \ldots, 0)$ lies in the span of rows associated with $\mathcal{S}$. This means, there exist constants $\gamma_i \in \mathbb{Z}_p$ for $i \in I$ such that $\sum_{i \in I} \gamma_i \mathbf{M}_i = (1, 0, \ldots, 0)$. These constants can be computed in time polynomial in the size of $\mathbf{M}$. On the contrary, $(\mathbf{M}, \pi)$ does not accept $\mathcal{S}$ if there exist a vector $\mathbf{w} \in \mathbb{Z}_p^{n_2}$ such that $\mathbf{w}$ is orthogonal to all rows $\mathbf{M}_i$ for $\pi(i) \in \mathcal{S}$, but not to $(1, 0, \ldots, 0)$. That means $\langle \mathbf{w}, \mathbf{M}_i \rangle = 0$. W.l.o.g. we can set $\mathbf{w}[1] = 1$.

Polynomials. Let p be a prime and $n \in \mathbb{N}$. We denote the set of multi-variate polynimals over $\mathbb{Z}_p$ with indeterminates $\tilde{x}_1, \ldots, \tilde{x}_n$ by $\mathbb{Z}_p[\tilde{x}_1, \ldots, \tilde{x}_n]$.

3 PES-ABE

We consider PES-ABE, which is a standard ABE scheme augmented with 3 deterministic algorithms Setup_0, Enc_0, KeyGen_0 used in Setup, Enc, KeyGen, Dec respectively, where:

- $\mathsf{Setup}_0(1^\lambda, \mathcal{X}, \mathcal{Y})$ outputs $n \in \mathbb{N}$,
- $\mathsf{Enc}_0(x)$ outputs linear functions $c^1 : \mathbb{Z}_p^{wn} \to \mathbb{Z}_p^{w_1}$, $c^2 : \mathbb{Z}_p^{w} \to \mathbb{Z}_p^{w_2}$,
- $\mathsf{KeyGen}_0(y)$ outputs linear functions $k^1 : \mathbb{Z}_p^{1+m+mn} \to \mathbb{Z}_p^{m_1}$, $k^2 : \mathbb{Z}_p^{m} \to \mathbb{Z}_p^{m_2}$,

and

- $\mathsf{Setup}(1^\lambda)$: Run $\mathcal{G} \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2) \leftarrow \mathsf{GroupGen}(1^\lambda)$, $n \leftarrow \mathsf{Setup}_0$. Pick $\alpha \xleftarrow{\$} \mathbb{Z}_p$ and a hash function $\mathsf{H} : [n] \to \mathbb{G}_1$ Output
$$\mathsf{mpk} \coloneqq (\mathcal{G}, \mathsf{H}, [\alpha]_T), \ \mathsf{msk} \coloneqq \alpha$$
Using H, we implictly define $\mathbf{b} \in \mathbb{Z}_p^n$ via $[\mathbf{b}[i]]_1 = \mathsf{H}(i)$.

– Enc: Run $(c^1, c^2) \leftarrow \mathsf{Enc}_0(x)$. Pick $\mathbf{s} \leftarrow \mathbb{Z}_p^w$. Compute $[\mathbf{c}^1]_1 := c^1([\mathbf{s} \otimes \mathbf{b}]_1), [\mathbf{c}^2]_2 := c^2([\mathbf{s}]_2)$ where $\mathbf{c}^2[1] = \mathbf{s}[1]$. Output

$$\mathsf{ct} := ([\mathbf{c}^1]_1, [\mathbf{c}^2]_2),\ \mathsf{kem} := [\alpha \mathbf{s}[1]]_T$$

– KeyGen: Run $(k^1, k^2) \leftarrow \mathsf{KeyGen}_0(y)$. Pick $\mathbf{r} \leftarrow \mathbb{Z}_p^m$. Compute $[\mathbf{k}^1]_1 := k^1([\alpha]_1, [\mathbf{r}]_1, [\mathbf{b} \otimes \mathbf{r}]_1), [\mathbf{k}^2]_2 := k^2([\mathbf{r}]_2)$. Output

$$\mathsf{sk} := ([\mathbf{k}^1]_1, [\mathbf{k}^2]_2)$$

Note that Enc and KeyGen compute the linear functions c^1, k^1 "in the exponent" since it only knows $[\mathbf{b}]_1$ and not $\mathbf{b}$. We also require that c^1, k^1 depend only on a polynomial number of entries in $\mathbf{b}$, so that $\mathsf{Enc}, \mathsf{KeyGen}$ only need to make a polynomial number of calls to H to compute $[c^1(\mathbf{s} \otimes \mathbf{b})]_1$ and $[k^1(\alpha, \mathbf{r}, \mathbf{b} \otimes \mathbf{r})]_1$ respectively. Depending on the application, some of these calls to H can also be pre-computed.

Remark 1 (Decryption). Note that we can augment PES-ABE with an additional deterministic algorithm Dec_0 used in Dec where

– $\mathsf{Dec}_0(x, y)$ outputs $\mathbf{e} \in \mathbb{Z}_p^{w_1 m_2}, \mathbf{e}' \in \mathbb{Z}_p^{w_2 m_1}$;

– $\mathsf{Dec}(\mathsf{mpk}, x, y, \mathsf{ct} = ([\mathbf{c}^1]_1, [\mathbf{c}^2]_2), \mathsf{sk} = ([\mathbf{k}^1]_1, [\mathbf{k}^2]_2)$: Run $(\mathbf{e}, \mathbf{e}') \leftarrow \mathsf{Dec}_0(x, y)$. Compute $[\mathbf{k}^1 \otimes \mathbf{c}^2]_T, [\mathbf{c}^1 \otimes \mathbf{k}^2]_T$ using e, and output $[(\mathbf{k}^1 \otimes \mathbf{c}^2) \cdot \mathbf{e}^\top + (\mathbf{c}^1 \otimes \mathbf{k}^2) \cdot \mathbf{e}'^\top]_T$.

It would then follow from ABE correctness that if $\mathsf{P}(x, y) = 1$, $(\mathbf{k}^1 \otimes \mathbf{c}^2) \cdot \mathbf{e}^\top + (\mathbf{c}^1 \otimes \mathbf{k}^2) \cdot \mathbf{e}'^\top = \alpha \mathbf{s}[1]$. We omit Dec_0 in our presentation and instead, specify and analyze Dec for correctness directly. This does not affect our security notions and proofs which only refer to $\mathsf{Enc}, \mathsf{KeyGen}, \mathsf{Enc}_0, \mathsf{KeyGen}_0$.

4 Symbolic Security of PES-ABE

Following previous work [3,6], we define symbolic security for PES-ABE, where we replace the inputs $(\alpha, \mathbf{b}, \mathbf{s}, \mathbf{r}) \leftarrow \mathbb{Z}_p \times \mathbb{Z}_p^n \times \mathbb{Z}_p^w \times \mathbb{Z}_p^m$ to the linear functions (c^1, c^2, k^1, k^2) with vectors of formal variables

$$(\tilde{\alpha}, \tilde{\mathbf{b}}, \tilde{\mathbf{s}}, \tilde{\mathbf{r}}) \leftarrow \mathsf{Var} \times \mathsf{Var}^n \times \mathsf{Var}^w \times \mathsf{Var}^m$$

In particular, $c^1(\tilde{\mathbf{s}} \otimes \tilde{\mathbf{b}}), c^2(\tilde{\mathbf{s}}), k^1(\tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}), k^2(\tilde{\mathbf{r}})$ are now (vectors of) polynomials in $\mathbb{Z}_p[\tilde{\alpha}, \tilde{\mathbf{b}}, \tilde{\mathbf{s}}, \tilde{\mathbf{r}}]$.

4.1 Definitions

Fix $x \in \mathcal{X}, y \in \mathcal{Y}$. ABE correctness tells us that if $\mathsf{P}(x, y) = 1$, then

$$\tilde{\alpha}\tilde{\mathbf{s}}[1] \in \mathrm{span}\big(c^1(\tilde{\mathbf{s}} \otimes \tilde{\mathbf{b}}) \otimes k^2(\tilde{\mathbf{r}}) \| k^1(\tilde{\alpha}, \tilde{\mathbf{r}}, \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}) \otimes c^2(\tilde{\mathbf{s}})\big)$$

On the other hand, if $\mathsf{P}(x, y) = 0$, it should be the case that

$$\tilde{\alpha}\tilde{\mathbf{s}}[1] \notin \mathrm{span}\big(c^1(\tilde{\mathbf{s}} \otimes \tilde{\mathbf{b}}) \otimes k^2(\tilde{\mathbf{r}}) \| k^1(\tilde{\alpha}, \tilde{\mathbf{r}}, \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}) \otimes c^2(\tilde{\mathbf{s}})\big)$$

Our basic formulation of symbolic security stipulates something stronger, where we basically replace $\tilde{\alpha}\tilde{\mathbf{s}}[1]$ with $\tilde{\alpha} \otimes c^2(\tilde{\mathbf{s}})$ and require $c^2(\tilde{\mathbf{s}})[1] = \tilde{\mathbf{s}}[1]$. In the special case where $c^2(\tilde{\mathbf{s}}) = \tilde{\mathbf{s}}[1]$ (as is the case when $w_2 = 1$), these two requirements are equivalent.

Definition 2 ($(1,1)$ Symbolic Security). *For all $x \in \mathcal{X}, y \in \mathcal{Y}$ such that $\mathsf{P}(x, y) = 0$: we have*

$$\mathrm{span}\big(\tilde{\alpha} \otimes \mathbf{c}^2\big) \cap \mathrm{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2 \parallel \mathbf{k}^1 \otimes \mathbf{c}^2\big) = \{0\}\ ,$$

where

$$(\tilde{\alpha}, \tilde{\mathbf{b}}) \leftarrow \mathsf{Var} \times \mathsf{Var}^n$$
$$(\mathbf{c}^1, \mathbf{c}^2) := (c^1(\tilde{\mathbf{s}} \otimes \tilde{\mathbf{b}}), c^2(\tilde{\mathbf{s}})),\ \tilde{\mathbf{s}} \leftarrow \mathsf{Var}^w,\ (c^1, c^2) \leftarrow \mathsf{Enc}_0(x)$$
$$(\mathbf{k}^1, \mathbf{k}^2) := (k^1(\tilde{\alpha}, \tilde{\mathbf{r}}, \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}), k^2(\tilde{\mathbf{r}})),\ \tilde{\mathbf{r}} \leftarrow \mathsf{Var}^m,\ (k^1, k^2) \leftarrow \mathsf{KeyGen}_0(y).$$

The symbolic property captured by this definition will be required to prove many-ciphertext CPA security of our ABE scheme. To capture the ABE security experiment more closely, we extend the definition such that it also include many secret keys, many ciphertexts as well as the public key. Also we consider that in the ABE security experiment the adversary may ask for the same x or y multiple times. In Lemma 1 below, we show that this stronger symbolic property is actually implied by the weaker one above.

Definition 3 (Strong Symbolic Security). *For all $Q_{\mathsf{ct}}, Q_{\mathsf{sk}} \in \mathbb{N}$, $X \in \mathcal{X}^{Q_{\mathsf{ct}}}, Y \in \mathcal{Y}^{Q_{\mathsf{sk}}}$ such that $\mathsf{P}(X[i], Y[j]) = 0$ for all $i \in [Q_{\mathsf{ct}}]$, $j \in [Q_{\mathsf{sk}}]$, we have*

$$\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}_X^2\big) \cap \operatorname{span}\big(\tilde{\alpha} \parallel (1 \| \tilde{\mathbf{b}} \| \mathbf{c}_X^1 \| \mathbf{k}_Y^1) \otimes (1 \| \mathbf{c}_X^2 \| \mathbf{k}_Y^2)\big) = \{0\} \ ,$$

where

$$\begin{aligned}
&(\tilde{\alpha}, \tilde{\mathbf{b}}) \leftarrow \mathsf{Var} \times \mathsf{Var}^n \\
&(\mathbf{c}_i^1, \mathbf{c}_i^2) := (c^1(\tilde{\mathbf{s}}_i \otimes \tilde{\mathbf{b}}), c^2(\tilde{\mathbf{s}}_i)),\ \tilde{\mathbf{s}}_i \leftarrow \mathsf{Var}^w, (c_i^1, c_i^2) \leftarrow \mathsf{Enc}_0(X[i]), \forall i \in [Q_{\mathsf{ct}}], \\
&(\mathbf{k}_j^1, \mathbf{k}_j^2) := (k^1(\tilde{\alpha}, \tilde{\mathbf{r}}_j, \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}_j), k^2(\tilde{\mathbf{r}}_j)),\ \tilde{\mathbf{r}}_j \leftarrow \mathsf{Var}^m, (k_j^1, k_j^2) \leftarrow \mathsf{KeyGen}_0(Y[j]), \forall j \in [Q_{\mathsf{sk}}], \\
&\mathbf{c}_X^1 := (\mathbf{c}_1^1 \| \cdots \| \mathbf{c}_{Q_{\mathsf{ct}}}^1), \mathbf{c}_X^2 := (\mathbf{c}_1^2 \| \cdots \| \mathbf{c}_{Q_{\mathsf{ct}}}^2) \\
&\mathbf{k}_Y^1 := (\mathbf{k}_1^1 \| \cdots \| \mathbf{k}_{Q_{\mathsf{sk}}}^1), \mathbf{k}_Y^2 := (\mathbf{k}_1^2 \| \cdots \| \mathbf{k}_{Q_{\mathsf{sk}}}^2) \ .
\end{aligned}$$

4.2 Relations

Now we can establish the desired implication in the following lemma.

Lemma 1. *If a PES-ABE scheme satisfies $(1,1)$ symbolic security (Definition 2), then it also satisfies strong symbolic security (Definition 3).*

The proof follows the high-level strategy laid out in [6, Theorem 4.1] with two main differences: (i) the proof of Claim 2 where we handle $w_2 > 1$ (see also Remark 2) and (ii) Step 2 where we handle many-ciphertext security.

Proof. Fix a PES-ABE satisfying $(1,1)$ symbolic security as well as $Q_{\mathsf{sk}}, Q_{\mathsf{ct}}, X, Y$ satisfying the conditions in Definition 3. We want to show that

$$\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}_X^2\big) \cap \operatorname{span}\big(\tilde{\alpha} \parallel (1 \| \tilde{\mathbf{b}} \| \mathbf{c}_X^1 \| \mathbf{k}_Y^1) \otimes (1 \| \mathbf{c}_X^2 \| \mathbf{k}_Y^2)\big) = \{0\} \ . \tag{2}$$

The proof proceeds in three steps.

Step 1. First, we show that for all $i \in [Q_{\mathsf{ct}}]$,

$$\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}_i^2\big) \cap \operatorname{span}\big(\mathbf{c}_i^1 \otimes \mathbf{k}_Y^2 \parallel \mathbf{k}_Y^1 \otimes \mathbf{c}_i^2\big) = \{0\} \ .$$

The proof proceeds by contradiction. Suppose on the contrary that there exist $i^* \in [Q_{\mathsf{ct}}]$, $\mathbf{e}^* \in \mathbb{Z}_p^{w_2}$, $\mathbf{e}_j \in \mathbb{Z}_p^{w_1 m_2}$, $\mathbf{e}'_j \in \mathbb{Z}_p^{m_1 w_2}$ for all $j \in [Q_{\mathsf{sk}}]$ such that $\mathbf{e}^* \neq \mathbf{0}$ and

$$(\tilde{\alpha} \otimes \mathbf{c}_{i^*}^2) \cdot \mathbf{e}^{*\top} = \sum_{j \in [Q_{\mathsf{sk}}]} (\mathbf{c}_{i^*}^1 \otimes \mathbf{k}_j^2) \cdot \mathbf{e}_j^\top + (\mathbf{k}_j^1 \otimes \mathbf{c}_{i^*}^2) \cdot \mathbf{e}_j'^\top \ . \tag{3}$$

We claim that $\{\mathbf{e}_j, \mathbf{e}'_j\}_{j \in [Q_{\mathsf{sk}}]}$ then satisfies

- Claim 1: $(\mathbf{c}_{i^*}^1 \otimes \mathbf{k}_j^2) \cdot \mathbf{e}_j^\top + (k_j^1(0, \tilde{\mathbf{r}}_j, \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}_j) \otimes \mathbf{c}_{i^*}^2) \cdot \mathbf{e}_j'^\top = 0$ for all $j \in [Q_{\mathsf{sk}}]$.
- Claim 2: there exists $j^* \in [Q_{\mathsf{sk}}], \boldsymbol{\mu} \in \mathbb{Z}_p^{w_2}$ such that $\boldsymbol{\mu} \neq \mathbf{0}$ and $(\tilde{\alpha} \otimes \mathbf{c}_{i^*}^2) \cdot \boldsymbol{\mu}^\top = (k_{j^*}^1(\tilde{\alpha}, \mathbf{0}, \mathbf{0}) \otimes \mathbf{c}_{i^*}^2) \cdot \mathbf{e}_{j^*}'^\top$.

Combining the two claims with the fact that $\mathbf{k}_{j^*}^1 = k_{j^*}^1(0, \tilde{\mathbf{r}}_{j^*}, \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}_{j^*}) + k_{j^*}^1(\tilde{\alpha}, \mathbf{0}, \mathbf{0})$, we have

$$(\tilde{\alpha} \otimes \mathbf{c}_{i^*}^2) \cdot \boldsymbol{\mu}^\top = (\mathbf{c}_{i^*}^1 \otimes \mathbf{k}_{j^*}^2) \cdot \mathbf{e}_{j^*}^\top + (\mathbf{k}_{j^*}^1 \otimes \mathbf{c}_{i^*}^2) \cdot \mathbf{e}_{j^*}'^\top \ .$$

which contradicts $(1,1)$ symbolic security since $\mathsf{P}(X[i^*], Y[j^*]) = 0$. It remains to establish Claims 1 and 2 to complete the proof:

– Fix $j \in [Q_{\mathsf{sk}}]$. Claim 1 follows from evaluating (3) on $\tilde{\alpha} = 0, \tilde{\mathbf{r}}_{j'} = \mathbf{0}\ \forall j' \in [Q_{\mathsf{sk}}] \setminus \{j\}$.

– Next, evaluating (3) on $\tilde{\mathbf{r}}_j = \mathbf{0}\ \forall j \in [Q_{\mathsf{sk}}]$ yields

$$0 \neq (\tilde{\alpha} \otimes \mathbf{c}^2_{i^*}) \cdot \mathbf{e}^{*\top} = \sum_{j \in [Q_{\mathsf{sk}}]} (k^1_j(\tilde{\alpha}, \mathbf{0}, \mathbf{0}) \otimes \mathbf{c}^2_{i^*}) \cdot \mathbf{e}'^{\top}_j \ .$$

Therefore, there exists $j^* \in Y$ such that $(k^1_{j^*}(\tilde{\alpha}, 0, 0) \otimes \mathbf{c}^2_{i^*}) \cdot \mathbf{e}'^{\top}_{j^*} \neq 0$. Moreover, since the polynomial $k^1_{j^*}(\tilde{\alpha}, \mathbf{0}, \mathbf{0})$ is linear in $\tilde{\alpha}$, there exists $\boldsymbol{\mu} \neq \mathbf{0}$ such that $(k^1_{j^*}(\tilde{\alpha}, 0, 0) \otimes \mathbf{c}^2_{i^*}) \cdot \mathbf{e}'^{\top}_{j^*} = (\tilde{\alpha} \otimes \mathbf{c}^2_{i^*}) \cdot \boldsymbol{\mu}^{\top}$ and Claim 2 follows.

Step 2. We show that

$$\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}^2_X\big) \cap \operatorname{span}\big(\mathbf{c}^1_X \otimes \mathbf{k}^2_Y \parallel \mathbf{k}^1_Y \otimes \mathbf{c}^2_X\big) = \{0\} \ .$$

As in the previous step, the proof proceeds by contradiction. Suppose the above statement is false, which means there exist $\{\mathbf{e}^*_i \in \mathbb{Z}_p^{w_2}, \mathbf{e}_i \in \mathbb{Z}_p^{Q_{\mathsf{sk}} \cdot w_1 m_2}, \mathbf{e}'_i \in \mathbb{Z}_p^{Q_{\mathsf{sk}} \cdot m_1 w_2}\}_{i \in [Q_{\mathsf{ct}}]}$ and $i^* \in [Q_{\mathsf{ct}}]$ such that

$$\sum_{i \in [Q_{\mathsf{ct}}]} (\tilde{\alpha} \otimes \mathbf{c}^2_i) \cdot \mathbf{e}^{*\top}_i = \sum_{i \in [Q_{\mathsf{ct}}]} (\mathbf{c}^1_i \otimes \mathbf{k}^2_Y) \cdot \mathbf{e}^{\top}_i + (\mathbf{k}^1_Y \otimes \mathbf{c}^2_i) \cdot \mathbf{e}'^{\top}_i \ , \tag{4}$$

and $\mathbf{e}^*_{i^*} \neq \mathbf{0}$. We evaluate (4) on $\tilde{\mathbf{s}}_{i'} = \mathbf{0}\ \forall i' \in [Q_{\mathsf{ct}}] \setminus \{i^*\}$ and get

$$(\tilde{\alpha} \otimes \mathbf{c}^2_{i^*}) \cdot \mathbf{e}^{*\top}_{i^*} = (\mathbf{c}^1_{i^*} \otimes \mathbf{k}^2_Y) \cdot \mathbf{e}^{\top}_{i^*} + (\mathbf{k}^1_Y \otimes \mathbf{c}^2_{i^*}) \cdot \mathbf{e}'^{\top}_{i^*} \ .$$

That is, $\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}^2_{i^*}\big) \cap \operatorname{span}\big(\mathbf{c}^1_{i^*} \otimes \mathbf{k}^2_Y \| \mathbf{k}^1_Y \otimes \mathbf{c}^2_{i^*}\big) \neq \{0\}$, which contradicts what we showed in Step 1.

Step 3. We now prove (2), which also proceeds by contradiction. Suppose on the contrary that

$$\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}^2_X\big) \cap \operatorname{span}\big(\tilde{\alpha} \| (1 \| \tilde{\mathbf{b}} \| \mathbf{c}^1_X \| \mathbf{k}^1_Y) \otimes (1 \| \mathbf{c}^2_X \| \mathbf{k}^2_Y)\big) \neq \{0\} \ .$$

Then there exist $\mathbf{e}^* \in \mathbb{Z}_p^{Q_{\mathsf{ct}} \cdot w_2}$, $\mathbf{e}_{PK} \in \mathbb{Z}_p^{2+n}$, $\mathbf{e}_X \in \mathbb{Z}_p^{Q_{\mathsf{ct}} \cdot (w_1 + w_2 + w_1 w_2 + n w_2)}$, $\mathbf{e}_Y \in \mathbb{Z}_p^{Q_{\mathsf{sk}} \cdot (m_1 + m_2 + m_1 m_2 + n m_2)}$, $\mathbf{e}_{XY} \in \mathbb{Z}_p^{Q_{\mathsf{ct}} \cdot Q_{\mathsf{sk}} \cdot (w_1 m_2 + m_1 w_2)}$ such that

$$\begin{aligned}(\tilde{\alpha} \otimes \mathbf{c}^2_X) \cdot \mathbf{e}^{*\top} = {} & (1 \| \tilde{\alpha} \| \tilde{\mathbf{b}}) \cdot \mathbf{e}^{\top}_{PK} + (\mathbf{c}^1_X \| \mathbf{c}^2_X \| \mathbf{c}^1_X \otimes \mathbf{c}^2_X \| \tilde{\mathbf{b}} \otimes \mathbf{c}^2_X) \cdot \mathbf{e}^{\top}_X \\ & + (\mathbf{k}^1_Y \| \mathbf{k}^2_Y \| \mathbf{k}^1_Y \otimes \mathbf{k}^2_Y \| \tilde{\mathbf{b}} \otimes \mathbf{k}^2_Y) \cdot \mathbf{e}^{\top}_Y \\ & + (\mathbf{c}^1_X \otimes \mathbf{k}^2_Y \| \mathbf{k}^1_Y \otimes \mathbf{c}^2_X) \cdot \mathbf{e}^{\top}_{XY}\end{aligned} \tag{5}$$

and $\mathbf{e}^* \neq \mathbf{0}$. First, we look at the first three terms on the RHS of (5):

– Evaluating (5) on $\tilde{\alpha} = 0$, $\tilde{\mathbf{r}}_Y = \mathbf{0}$, and $\tilde{\mathbf{s}}_X = 0$ yields $(1 \| 0 \| \tilde{\mathbf{b}}) \cdot \mathbf{e}^{\top}_{PK} = 0$.

– Evaluating (5) on $\tilde{\alpha} = 0$, $\tilde{\mathbf{r}}_Y = \mathbf{0}$ yields $(1 \| 0 \| \tilde{\mathbf{b}}) \cdot \mathbf{e}^{\top}_{PK} + (\mathbf{c}^1_X \| \mathbf{c}^2_X \| \mathbf{c}^1_X \otimes \mathbf{c}^2_X \| \tilde{\mathbf{b}} \otimes \mathbf{c}^2_X) \cdot \mathbf{e}^{\top}_X = 0$.

– Evaluating (5) on $\tilde{\mathbf{s}}_X = 0$ yields $(1 \| \tilde{\alpha} \| \tilde{\mathbf{b}}) \cdot \mathbf{e}^{\top}_{PK} + (\mathbf{k}^1_Y \| \mathbf{k}^2_Y \| \mathbf{k}^1_Y \otimes \mathbf{k}^2_Y \| \tilde{\mathbf{b}} \otimes \mathbf{k}^2_Y) \cdot \mathbf{e}^{\top}_Y = 0$.

Subtracting the first equality from the sum of the second and third implies that the sum of the first three terms on the RHS of (5) is 0. This means

$$(\tilde{\alpha} \otimes \mathbf{c}^2_X) \cdot \mathbf{e}^{*\top} = (\mathbf{c}^1_X \otimes \mathbf{k}^2_Y \| \mathbf{k}^1_Y \otimes \mathbf{c}^2_X) \cdot \mathbf{e}^{\top}_{XY}$$

which contradicts what we showed in Step 2.

Remark 2 (handling $w_1 > 1$). In the proof of the analogue of Claim 2 in [6], they start with

$$\tilde{\alpha} \mathbf{c}^2_{i^*}[1] = \sum_{j \in [Q_{\mathsf{sk}}]} (k^1_j(\tilde{\alpha}, \mathbf{0}, \mathbf{0}) \otimes \mathbf{c}^2_{i^*}) \cdot \mathbf{e}'^{\top}_j \ .$$

They show that if $w_2 = 1$ (a requirement mentioned in the proof[9] but not in the theorem statement), then there exists $j^* \in Y$ such $\mu \cdot \mathbf{c}^2_{i^*}[1] = k^1_{j^*}(1, \mathbf{0}, \mathbf{0}) \otimes \mathbf{c}^2_{i^*} \cdot \mathbf{e}'^{\top}_{j^*}$ and $\mu \neq 0$. However, if we allow $w_2 > 1$, then this claim does not hold in general. In particular, it could be that for all j, $\mathbf{c}^2_{i^*}[1]$ only appears in a linear combination with other elements of $\mathbf{c}^2_{i^*}$, which then all together sum up to $\mathbf{c}^2_{i^*}[1]$. For this reason, we need to strengthen our definition accordingly.

[9] On page 662, they wrote "since we assumed $w_1 = 0$". Here, $c^2(\tilde{\mathbf{s}})$ corresponds to $\overrightarrow{S} = (S_0, \ldots, S_{w_1})$ in [6].

5 Optimal ABE Security in the GGM

We prove symbolic security of PES-ABE implies optimal, adaptive security in the generic group model (GGM). For that, we first recall the generic group model.

5.1 Generic Group Model

In the generic group model, an adversary can perform group operations only via oracle access. We adopt the model by Maurer [42] extended to the pairing group setting, where apart from the group operation, the adversary can also compute the pairing via an oracle. A third party implements the pairing group and maintains a list for $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$. Each list stores group elements of queries by the adversary. Depending on the query, one or multiple entries are appended to the different lists. The adversary can access each entry of the lists by a handle, which is a list index $i \in \mathbb{N}$ and a list identifier $s \in \{1, 2, T\}$. It can also perform equality queries to check if two entries of the same list contain the same group element.

In game G_0 in Figure 2, we model the ABE security game from Section 2.2 in the GGM. That is, the adversary also gets access to oracles $\mathcal{O}_{\mathsf{mpk}}$, $\mathcal{O}_{\mathsf{ct}}$ and $\mathcal{O}_{\mathsf{sk}}$. On each query, the corresponding oracle returns the current length of all modified lists from which the adversary can deduce the corresponding handles since length of ciphertexts and secret keys follow from the definition of the scheme. Furthermore, we model the hash function in our scheme as random oracle, so we additionally provide an oracle H, which also modify the lists. The adversary can then use these indices in further group operation and equality queries as described above.

5.2 Security

The following theorem states that symbolic security implies optimal, adaptive security in GGM.

Theorem 1. *Let $\lambda \in \mathbb{N}$ be the security parameter and $\mathcal{A}$ be an adversary that on input $(1^\lambda, p)$ makes Q_{add}, Q_{pair}, Q_{ct}, Q_{sk}, Q_{eq} queries to oracles $\mathcal{O}_{\mathsf{add}}$, $\mathcal{O}_{\mathsf{pair}}$, $\mathcal{O}_{\mathsf{ct}}$, $\mathcal{O}_{\mathsf{sk}}$, $\mathcal{O}_{\mathsf{eq}}$ and Q_{H} queries to the random oracle H. If PES-ABE is (1,1) symbolically secure (Definition 2), then it is adaptively Many-CT secure in the GGM. In particular,*

$$\mathsf{Adv}^{\mathsf{GGM}}_{\mathsf{ABE},\mathcal{A}}(\lambda) \leq \frac{3 \cdot (Q_{\mathsf{H}} + (w'+1) \cdot Q_{\mathsf{ct}} + m' \cdot Q_{\mathsf{sk}} + Q_{\mathsf{add}} + Q_{\mathsf{pair}})^2}{p} \ ,$$

where $w' \coloneqq w_1 + w_2$ and $m' \coloneqq m_1 + m_2$.

We first recall a useful lemma that is commonly used to prove security in the GGM.

Lemma 2 (Schwartz-Zippel Lemma). *For any prime p and $t \in \mathbb{N}^*$, any polynomial $f \in \mathbb{Z}_p[\tilde{x}_1, \ldots, \tilde{x}_t]$ of degree $d > 0$,*

$$\Pr[f(\mathbf{x}) = 0] \leq \frac{d}{p} \ ,$$

where the probability is taken over $\mathbf{x} \xleftarrow{\$} \mathbb{Z}_p^t$.

We will now state the full proof of Theorem 1. In fact, it is similar to that in [6, Theorem 3.3]. The latter only considers single-ciphertext ABE security, and achieves an additional loss of Q_{sk}, since they apply Schwartz-Zippel to polynomials of degree Q_{sk} in order to handle rational fractions arising in their schemes.

Proof. The proof will be done by a hybrid argument over the queries to oracle $\mathcal{O}_{\mathsf{eq}}$. The first hybrid G_0 is the many-CT CPA security game for the ABE scheme in the generic group model. We will proceed in two main steps.

1. In each hybrid, we replace the check whether two elements are equal by checking whether their corresponding polynomials are equal;
2. in the last game $\mathsf{G}_{Q_{\mathsf{eq}}}$, we use the symbolic security property of the ABE scheme to show that outputs are independent of the challenge bit β.

Now consider the games G_0-$\mathsf{G}_{Q_{\mathsf{eq}}}$ given in Figure 2. In the following we denote the advantage of adversary $\mathcal{A}$ in G_μ by Adv_μ.

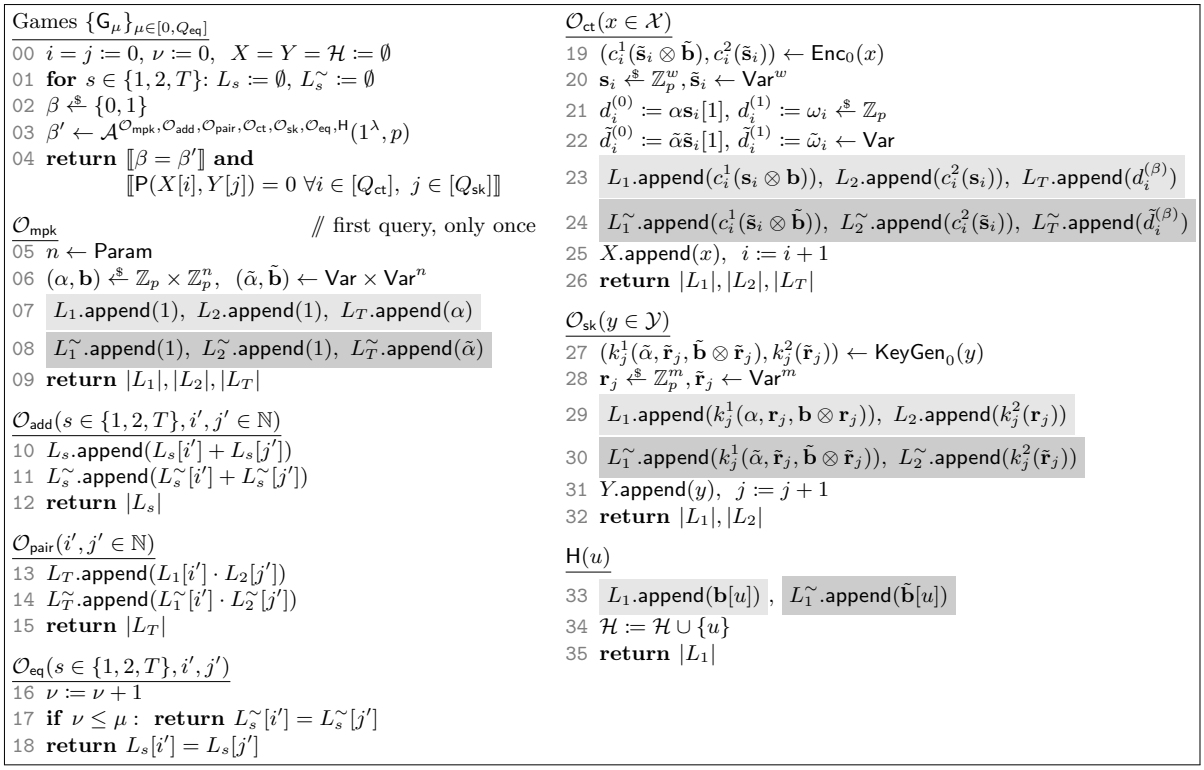


Fig. 2: Games G_μ for $\mu\in[0,Q_{\mathsf{eq}}]$ for the proof of Theorem 1. Note that the games only differ in oracle $\mathcal{O}_{\mathsf{eq}}$ (which depends on μ). Here, G_0 corresponds to the GGM experiment that makes only use of components in light gray frames, whereas $\mathsf{G}_{Q_{\mathsf{eq}}}$ makes only use of components in dark gray frames. W.l.o.g. we assume that no query to $\mathcal{O}_{\mathsf{add}},\mathcal{O}_{\mathsf{pair}},\mathcal{O}_{\mathsf{eq}}$ contains indices $i',j'\in\mathbb{N}$ which exceed the size of the involved lists.

Game G_0. This game captures the ABE security experiment for PES-ABE as described in Section 3 in the generic group model, thus

$$\mathsf{Adv}_0=\mathsf{Adv}^{\mathsf{GGM}}_{\mathsf{ABE},\mathcal{A}}(\lambda)\ .$$

We use lists L_1, L_2 and L_T to store scalars in $\mathbb{Z}_p$ which correspond to public parameters, ciphertexts and secret keys in $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$, respectively. To keep track of queries to $\mathcal{O}_{\mathsf{ct}}$ and $\mathcal{O}_{\mathsf{sk}}$, we store them in lists X and Y.

We also already store polynomials in separate lists which we denote by $L_1^{\sim}$, $L_2^{\sim}$ and $L_T^{\sim}$. As in the previous section, we label formal variables with a tilde, e.g. $\tilde\alpha$ is the formal variable corresponding to the master secret key $\alpha\in\mathbb{Z}_p$. Note that in $\mathcal{O}_{\mathsf{mpk}}$ we already define formal variables $\tilde b_i$ for all $i\in\mathcal{U}$, but they are only added to L_1, when the random oracle is queried on i. All queries to H are additionally stored in a set $\mathcal{H}$.

Game $\{\mathsf{G}_\mu\}_{\mu\in[Q_{\mathsf{eq}}]}$. We now proceed using a hybrid argument. Note that the only difference between $\mathsf{G}_{\mu-1}$ and G_μ lies in how we answer the μ-th query to oracle $\mathcal{O}_{\mathsf{eq}}$: in $\mathsf{G}_{\mu-1}$, we are using L_T thus comparing scalars, and in G_μ, we are using $L_T^{\sim}$ and thus comparing polynomials. Let (s,i,j) denote the μ-th query to oracle $\mathcal{O}_{\mathsf{eq}}$. We claim that

$$|\mathsf{Adv}_{\mu-1}-\mathsf{Adv}_\mu|\le\Pr[(L_s[i']\neq L_s[j'])\wedge(L_s^{\sim}[i']=L_s^{\sim}[j'])]\le\frac{3}{p}\ .$$

where the randomness is over $\alpha\xleftarrow{\$}\mathbb{Z}_p$, $\mathbf{b}\xleftarrow{\$}\mathbb{Z}_p^n$, $\mathbf{s}_i\xleftarrow{\$}\mathbb{Z}_p^w$, $\mathbf{r}_j\xleftarrow{\$}\mathbb{Z}_p^m$, $\omega_i\xleftarrow{\$}\mathbb{Z}_p$. Let $\tilde{\mathbf{d}}_X^{(\beta)}:=(\tilde d_1^{(\beta)}\|\cdots\|\tilde d_{|X|}^{(\beta)})$ and define $\mathbf{c}_X^1,\mathbf{c}_X^2,\mathbf{k}_Y^1,\mathbf{k}_Y^2$ as in Definition 2.

It is easy to see that:

$$
\begin{aligned}
L_1^{\sim} &\subseteq \mathrm{span}\big(\overbrace{1}^{\mathcal{O}_{\mathsf{mpk}}} \| \overbrace{\tilde{\mathbf{b}}}^{\mathsf{H}} \| \overbrace{\mathbf{c}_X^1}^{\mathcal{O}_{\mathsf{ct}}} \| \overbrace{\mathbf{k}_Y^1}^{\mathcal{O}_{\mathsf{sk}}}\big) \\
L_2^{\sim} &\subseteq \mathrm{span}\big(\overbrace{1}^{\mathcal{O}_{\mathsf{mpk}}} \| \overbrace{\mathbf{c}_X^2}^{\mathcal{O}_{\mathsf{ct}}} \| \overbrace{\mathbf{k}_Y^2}^{\mathcal{O}_{\mathsf{sk}}}\big) \\
L_T^{\sim} &\subseteq \mathrm{span}\big(\overbrace{\tilde{\mathbf{d}}_X^{(\beta)}}^{\mathcal{O}_{\mathsf{ct}}} \| \overbrace{\tilde{\alpha}}^{\mathcal{O}_{\mathsf{mpk}}} \| \overbrace{(1\|\tilde{\mathbf{b}}\|\mathbf{c}_X^1\|\mathbf{k}_Y^1) \otimes (1\|\mathbf{c}_X^2\|\mathbf{k}_Y^2)}^{\mathcal{O}_{\mathsf{pair}}}\big)
\end{aligned} \tag{6}
$$

where $\mathrm{span}(\cdot)$ captures queries to $\mathcal{O}_{\mathsf{add}}$. Since $\mathbf{c}_X^1, \mathbf{k}_Y^1$ have degrees at most 2 and $\mathbf{c}_X^2, \mathbf{k}_Y^2$ have degrees at most 1, the polynomials in $L_1^{\sim}, L_2^{\sim}$ and $L_T^{\sim}$ have degrees at most $1, 2$ and 3 respectively (in the formal variables $\tilde{\alpha}, \tilde{\mathbf{b}}, \tilde{\mathbf{s}}_X, \tilde{\mathbf{r}}_Y, \tilde{\omega}_X$). The claim then follows readily from the Schwartz-Zippel Lemma.

Game $\mathsf{G}_{Q_{\mathsf{eq}}}$. We show that $\mathsf{Adv}_{Q_{\mathsf{eq}}} = 0$. In particular, we claim that the view of the adversary in $\mathsf{G}_{Q_{\mathsf{eq}}}$ is independent of the challenge bit β. Observe that the only leakage about β is the output of $\mathcal{O}_{\mathsf{eq}}$ on a query of the form (T, i', j'). By (6), there exists $\mathbf{e}_{i'}, \mathbf{e}_{j'} \in \mathbb{Z}_p^{Q_{\mathsf{ct}}}$ and $\mathbf{v}_{i'}, \mathbf{v}_{j'} \in \mathrm{span}\big(\tilde{\alpha}\|(1\|\tilde{\mathbf{b}}\|\mathbf{c}_X^1\|\mathbf{k}_Y^1) \otimes (1\|\mathbf{c}_X^2\|\mathbf{k}_Y^2)\big)$ such that

$$
L_T^{\sim}[i'] = \tilde{\mathbf{d}}_X^{(\beta)} \cdot \mathbf{e}_{i'}^{\top} + \mathbf{v}_{i'},\ L_T^{\sim}[j'] = \tilde{\mathbf{d}}_X^{(\beta)} \cdot \mathbf{e}_{j'}^{\top} + \mathbf{v}_{j'} \ .
$$

It suffices to show that

$$
L_T^{\sim}[i'] = L_T^{\sim}[j'] \iff \mathbf{e}_{i'} = \mathbf{e}_{j'} \wedge \mathbf{v}_{i'} = \mathbf{v}_{j'}
$$

since the RHS is independent of β.

First, we show that for both cases $\beta = 0$ and $\beta = 1$, it holds that

$$
\mathrm{span}\big(\tilde{\mathbf{d}}_X^{(\beta)}\big) \cap \mathrm{span}\big(\tilde{\alpha}\|(1\|\tilde{\mathbf{b}}\|\mathbf{c}_X^1\|\mathbf{k}_Y^1) \otimes (1\|\mathbf{c}_X^2\|\mathbf{k}_Y^2)\big) = \{0\} \ , \tag{7}
$$

where $\tilde{\mathbf{d}}_X^{(0)} = (\tilde{\alpha}\tilde{\mathbf{s}}_1[1]\| \cdots \|\tilde{\alpha}\tilde{\mathbf{s}}_{Q_{\mathsf{ct}}}[1])$ and $\tilde{\mathbf{d}}_X^{(1)} = (\tilde{\omega}_1\| \cdots \|\tilde{\omega}_{Q_{\mathsf{ct}}})$.

- We first look at the case $\beta = 1$. Obviously, (7) holds since $\tilde{\omega}_i$ are fresh random variables for all challenges and do not appear anywhere else.
- For $\beta = 0$ first note that since $\mathbf{c}_i^2[1] = \mathbf{s}_i[1]$, we have

  $$
  \mathrm{span}\big(\tilde{\mathbf{d}}_X^{(0)}\big) \subseteq \mathrm{span}\big(\tilde{\alpha} \otimes \mathbf{c}_X^2\big) \ .
  $$

  Hence, (7) holds due to strong symbolic security of the ABE scheme, which in turn follows from $(1, 1)$ symbolic security via Lemma 1.

From (7), we have

$$
L_T^{\sim}[i'] = L_T^{\sim}[j'] \iff \tilde{\mathbf{d}}_X^{(\beta)} \cdot \mathbf{e}_{i'}^{\top} = \tilde{\mathbf{d}}_X^{(\beta)} \cdot \mathbf{e}_{j'}^{\top} \wedge \mathbf{v}_{i'} = \mathbf{v}_{j'} \ .
$$

It remains to argue that for both $\beta = 0$ and $\beta = 1$

$$
\tilde{\mathbf{d}}_X^{(\beta)} \cdot \mathbf{e}_{i'}^{\top} = \tilde{\mathbf{d}}_X^{(\beta)} \cdot \mathbf{e}_{j'}^{\top} \iff \mathbf{e}_{i'} = \mathbf{e}_{j'} \ .
$$

This follows from the fact that all terms in $\tilde{\mathbf{d}}_X^{(\beta)}$ are mutually independent. For this, note that $\tilde{\mathbf{s}}_i[1]$ as well as $\tilde{\omega}_i$ are mutually independent for all i and we get $\mathsf{Adv}_{Q_{\mathsf{eq}}} = 0$.

Summing everything up, we obtain

$$
\mathsf{Adv}^{\mathsf{GGM}}_{\mathsf{ABE},\mathcal{A}}(\lambda) \leq \frac{3 \cdot Q_{\mathsf{eq}}}{p} \ .
$$

Then, observing that

$$
\begin{aligned}
Q_{\mathsf{eq}} &\leq (|L_1| + |L_2| + |L_T|)^2 \\
&\leq (Q_{\mathsf{H}} + (w_1 + w_2 + 1) \cdot Q_{\mathsf{ct}} + (m_1 + m_2) \cdot Q_{\mathsf{sk}} + Q_{\mathsf{add}} + Q_{\mathsf{pair}})^2
\end{aligned}
$$

yields the bound stated in the theorem.

Note that the upper bound on Q_{eq} comes from the number of elements the adversary observes during the GGM experiment. Thus, $Q_{\mathsf{eq}} \leq (|L_1| + |L_2| + |L_T|)^2$ which corresponds to the fact that a real-world adversary can detect collisions amongst the group elements it has seen so far in time quasi-linear in $|L_1| + |L_2| + |L_T|$ via sorting, but needs to make $(|L_1| + |L_2| + |L_T|)^2$ queries to $\mathcal{O}_{\mathsf{eq}}$ to detect collisions.

6 Our Schemes: Putting Everything Together

We now show that the FABEO CP-ABE and KP-ABE schemes for monotone span programs described in Figure 1 satisfy the PES-ABE framework and $(1,1)$ symbolic security described in Section 2.2. Combined with the statements from Lemma 1 and Theorem 1, this establishes optimal, adaptive security of our CP-ABE and KP-ABE schemes in GGM (Corollaries 1 and 2). Nonetheless, we additionally prove selective security of our CP-ABE and KP-ABE schemes under the more conservative q-type assumptions in Appendix B.

6.1 CP-ABE

Our CP-ABE scheme is shown in Figure 1. It builds upon the pair encoding scheme 11 in [7] and that in Appendix B.1 in [3] and extends them by attribute hashing and multi-use of attributes. In particular, we can describe the underlying PES-ABE as follows.

- $\underline{\mathsf{Setup}_0}$. Output $n \coloneqq |\mathcal{U}| + 1$.
- $\underline{\mathsf{Enc}_0(\mathbf{M}, \pi)}$. Set $w = n_1 + \tau$, $w_1 = n_1$, $w_2 = \tau + 1$, and output (c^1, c^2) where we parse $\mathbf{s}$ as $(s_1 \| \mathbf{v} \| \mathbf{s}')$ and

$$c^1(\mathbf{s} \otimes \mathbf{b}) \coloneqq (\mathbf{M}_i(s_1\|\mathbf{v})^\top \cdot \mathbf{b}[|\mathcal{U}|+1] + \mathbf{s}'[\rho(i)] \cdot \mathbf{b}[\pi(i)])_{i\in[n_1]},$$
$$c^2(s_1) \coloneqq (s_1 \| \mathbf{s}')$$

- $\underline{\mathsf{KeyGen}_0(\mathcal{S})}$. Set $m = 1$, $m_1 = |\mathcal{S}| + 1$, $m_2 = 1$, and output (k^1, k^2) where we parse $\mathbf{r}$ as (r) and

$$k^1(\alpha, \mathbf{r}, \mathbf{b} \otimes \mathbf{r}) \coloneqq (\alpha + r\mathbf{b}[|\mathcal{U}|+1] \| (r\mathbf{b}[u])_{u\in\mathcal{S}}),$$
$$k^2(\mathbf{r}) \coloneqq (r)$$

Correctness. Let $\mathsf{ct} = (\mathsf{ct}_1, (\mathsf{ct}_{2,j})_{j\in[\tau]}, (\mathsf{ct}_{3,i})_{i\in[n_1]})$ be a ciphertext for $(\mathbf{M}, \pi)$ and $\mathsf{sk} = (\mathsf{sk}_1, (\mathsf{sk}_{2,u})_{u\in\mathcal{S}}, \mathsf{sk}_3)$ be a secret key for $\mathcal{S}$ as defined in Figure 1. Further let $\mathbf{b}[u]$ such that $\mathsf{H}(u) = g_1^{\mathbf{b}[u]}$ and b' such that $\mathsf{H}(|\mathcal{U}|+1) = g_1^{b'}$. If $\mathcal{S}$ satisfies $(\mathbf{M}, \pi)$, then there exist constants $(\gamma_i)_{i\in[n_1]}$ such that $\sum_{i\in I} \gamma_i \mathbf{M}_i = (1, 0, \ldots, 0)$ and decryption computes

(1) $e(g_1^\alpha \cdot \mathsf{H}(|\mathcal{U}|+1)^r, g_2^{s_1}) = [\alpha s_1 + b' r s_1]_T$

(2) $\prod_{j\in[\tau]} e(\prod_{i\in I, \rho(i)=j} \mathsf{H}(\pi(i))^{\gamma_i r}, g_2^{\mathbf{s}'[j]}) = [r \sum_{j\in[\tau]} \sum_{i\in I, \rho(i)=j} \gamma_i \mathbf{b}[\pi(i)]\mathbf{s}'[j]]_T$

(3) $e(\prod_{i\in I} (\mathsf{H}(|\mathcal{U}|+1)^{\gamma_i \mathbf{M}_i (s_1\|\mathbf{v})^\top} \cdot \mathsf{H}(\pi(i))^{\gamma_i \mathbf{s}'[\rho(i)]}), g_2^r) = [br' \underbrace{\sum_{i\in I} \gamma_i \mathbf{M}_i((s_1\|\mathbf{v})^\top]_T}_{=s_1} \cdot \underbrace{[r \sum_{i\in I} \gamma_i \mathbf{b}[\pi(i)]\mathbf{s}'[\rho(i)]]_T}_{=(2)}$

Note that by definition of ρ, (2) and the second term of (3) are the same. Thus computing $(1)\cdot(2)/(3)$ yields $d = [\alpha s_1]_T$.

Symbolic Security. We need to show that for all $(\mathbf{M}, \pi) \in \mathcal{X}$, $\mathcal{S} \in \mathcal{Y}$ such that $\mathsf{P}((\mathbf{M}, \pi), \mathcal{S}) = 0$, it holds that

$$\mathrm{span}\big(\tilde{\alpha} \otimes (\tilde{s}_1 \| \tilde{\mathbf{s}}')\big) \cap \mathrm{span}\big((\mathbf{M}_i(\tilde{s}_1, \tilde{\mathbf{v}})^\top \tilde{b}' + \tilde{\mathbf{s}}'[\rho(i)]\tilde{\mathbf{b}}[\pi(i)])_{i\in[n_1]} \otimes \tilde{r} \| (\tilde{\alpha} + \tilde{r}\tilde{b}' \| (\tilde{r}\tilde{\mathbf{b}}[u])_{u\in\mathcal{S}}) \otimes (\tilde{s}_1 \| \tilde{\mathbf{s}}')\big) = \{0\} \ ,$$

where we define $\tilde{b}' \coloneqq \tilde{\mathbf{b}}[|\mathcal{U}|+1]$.

We prove this property by contradiction. So assume there exists $\mathbf{e}^* \in \mathbb{Z}_p^2$, $\mathbf{e}$, $\mathbf{e}'^{(1)}$, $\mathbf{e}'^{(2)}$, $\mathbf{e}'^{(3)}$ such that $\mathbf{e}^* \neq \mathbf{0}$ and

$$(\tilde{\alpha} \otimes (\tilde{s}_1 \| \tilde{\mathbf{s}}')) \cdot \mathbf{e}^{*\top} = (\mathbf{M}_i(\tilde{s}_1 \| \tilde{\mathbf{v}})^\top \tilde{b}' \tilde{r} + \tilde{\mathbf{s}}'[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{r})_{i\in[n_1]} \cdot \mathbf{e}^\top + ((\tilde{\alpha} + \tilde{r}\tilde{b}') \otimes (\tilde{s}_1 \| \tilde{\mathbf{s}}')) \cdot \mathbf{e}'^{(1)\top}$$
$$+ (\{\tilde{r}\tilde{\mathbf{b}}[u]\tilde{s}_1\}_{u\in\mathcal{S}}) \cdot \mathbf{e}'^{(2)\top} + (\{\tilde{r}\tilde{\mathbf{b}}[u] \otimes \tilde{\mathbf{s}}'\}_{u\in\mathcal{S}}) \cdot \mathbf{e}'^{(3)\top} \ .$$

Now we use the fact that $\mathsf{P}((\mathbf{M},\pi),\mathcal{S})=0$. Recall that this means that there exists a vector $\mathbf{w}\in\mathbb{Z}_p^{n_2}$ such that $\langle \mathbf{w},\mathbf{M}_i\rangle=0$ for all $\pi(i)\in\mathcal{S}$ and that $\mathbf{w}[1]=1$. So evaluating on $(\tilde{s}_1\|\tilde{\mathbf{v}})=\mathbf{w}$ gives us

$$\begin{aligned}(\tilde{\alpha}\otimes(1\|\tilde{\mathbf{s}}'))\cdot\mathbf{e}^{*\top} = (\tilde{\mathbf{s}}'[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{r})_{i\in[n_1],\pi(i)\in\mathcal{S}}\cdot\overline{\mathbf{e}}^{\top} + (\mathbf{M}_i\mathbf{w}^{\top}\tilde{b}'\tilde{r}+\tilde{\mathbf{s}}'[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{r})_{i\in[n_1],\pi(i)\notin\mathcal{S}}\cdot\underline{\mathbf{e}}^{\top}\\ + ((\tilde{\alpha}+\tilde{r}\tilde{b}')\otimes(1\|\tilde{\mathbf{s}}'))\cdot\mathbf{e}'^{(1)\top} + (\tilde{r}\tilde{\mathbf{b}}[u])_{u\in\mathcal{S}}\cdot\mathbf{e}'^{(2)\top} + (\tilde{r}\tilde{\mathbf{b}}[u]\otimes\tilde{\mathbf{s}}')_{u\in\mathcal{S}}\cdot\mathbf{e}'^{(3)\top}\ ,\end{aligned}$$

where we split $\mathbf{e}$ into two vectors $\overline{\mathbf{e}}\in\mathbb{Z}_p^{|\mathcal{S}|}$ and $\underline{\mathbf{e}}\in\mathbb{Z}_p^{n_1-|\mathcal{S}|}$, capturing those rows of $\mathbf{M}$ that belong to $u\in\mathcal{S}$ and those that do not belong to an attribute in $\mathcal{S}$. Note that the monomials $\{\tilde{r}\tilde{\mathbf{b}}[u]\otimes\tilde{\mathbf{s}}'\}_{u\in\mathcal{S}}$ only appear in the first and the last term. By definition of ρ, the monomials $\tilde{\mathbf{s}}'[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{r}$ in the first term are mutually distinct. Thus, we must have $\overline{\mathbf{e}}[i]=-\mathbf{e}'^{(3)}[j]$ for all $i\in[n_1]$ such that $\pi(i)\in\mathcal{S}$ and unique indices j, while all other entries in $\mathbf{e}'^{(3)}$ must be 0. Further looking at monomials on the RHS, $\tilde{\mathbf{s}}'[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{r}$ for $\pi(i)\notin\mathcal{S}$ and $\tilde{r}\tilde{\mathbf{b}}[u]$ for $u\in\mathcal{S}$ are also mutually distinct and only appear in one of the terms, thus $\underline{\mathbf{e}}$ as well as $\mathbf{e}'^{(2)}$ must be $\mathbf{0}$. Therefore, the following equation must be satisfied

$$(\tilde{\alpha}\otimes(1\|\tilde{\mathbf{s}}'))\cdot\mathbf{e}^{*\top}=((\tilde{\alpha}+\tilde{r}\tilde{b}')\otimes(1\|\tilde{\mathbf{s}}'))\cdot\mathbf{e}'^{(1)\top}\ ,$$

which leads to a contradiction that $\mathbf{e}^*\neq\mathbf{0}$ since $\tilde{r}\tilde{b}'$ only appears on the RHS.

Corollary 1. *Let $\lambda\in\mathbb{N}$ be the security parameter and $\mathcal{A}$ be an adversary that on input $(1^\lambda,p)$ makes Q_{op} group operation queries to oracles $\mathcal{O}_{\mathsf{add}}$ and $\mathcal{O}_{\mathsf{pair}}$, as well as Q_{ct}, Q_{sk} queries to oracles $\mathcal{O}_{\mathsf{ct}}$, $\mathcal{O}_{\mathsf{sk}}$, and Q_{H} queries to the random oracle* H. *CP-ABE is adaptively secure in the GGM such that*

$$\mathsf{Adv}^{\mathsf{GGM}}_{\text{CP-ABE},\mathcal{A}}(\lambda)\leq\frac{3\cdot(Q_{\mathsf{H}}+(n_1+3)\cdot Q_{\mathsf{ct}}+(|\mathcal{S}|+2)\cdot Q_{\mathsf{sk}}+Q_{\mathsf{op}})^2}{p}\ ,$$

where $|\mathcal{S}|$ is the maximum size of the attribute sets queried to $\mathcal{O}_{\mathsf{sk}}$ and n_1 is the maximum number of rows of $\mathbf{M}$ queried to $\mathcal{O}_{\mathsf{ct}}$.

6.2 KP-ABE

Our KP-ABE scheme is shown in Figure 1. It builds upon the pair encoding scheme 9 in [7] and that in Appendix B.1 in [3] and extends them by attribute hashing and multi-use of attributes. The underlying PES-ABE can be described as follows.

- $\underline{\mathsf{Setup}_0}$. Output $n\coloneqq|\mathcal{U}|$.
- $\underline{\mathsf{Enc}_0(\mathcal{S})}$. Set $w=1$, $w_1=|\mathcal{S}|$, $w_2=1$, and output (c^1,c^2) where we parse $\mathbf{s}$ as (s) and

$$c^1(\mathbf{s}\otimes\mathbf{b})\coloneqq(s\mathbf{b}[u])_{u\in\mathcal{S}},\quad c^2(s_1)\coloneqq(s)$$

- $\underline{\mathsf{KeyGen}_0(\mathbf{M},\pi)}$. Set $m=\tau+n_2-1$, $m_1=n_1$, $m_2=\tau$, and output (k^1,k^2) where we parse $\mathbf{r}$ as $(\mathbf{r}'\|\mathbf{v})$, and and output

$$\begin{aligned}k^1(\alpha,\mathbf{r},\mathbf{b}\otimes\mathbf{r})&\coloneqq(\mathbf{M}_i(\alpha\|\mathbf{v})^{\top}+\mathbf{r}'[\rho(i)]\mathbf{b}[\pi(i)])_{i\in[n_1]}\\ k^2(\mathbf{r})&\coloneqq(\mathbf{r}')\end{aligned}$$

Correctness. Let $\mathsf{ct}=((\mathsf{ct}_{1,u})_{u\in\mathcal{S}},\mathsf{ct}_2)$ be a ciphertext for $\mathcal{S}$ and $\mathsf{sk}=((\mathsf{sk}_{1,j})_{j\in[\tau]},(\mathsf{sk}_{2,i})_{i\in[n_1]})$ be a secret key for $(\mathbf{M},\pi)$ as defined in Figure 1. Further let $\mathbf{b}[u]$ such that $\mathsf{H}(u)=g_1^{\mathbf{b}[u]}$. If $\mathcal{S}$ satisfies $(\mathbf{M},\pi)$, then there exist constants $(\gamma_i)_{i\in[n_1]}$ such that $\sum_{i\in I}\gamma_i\mathbf{M}_i=(1,0,\ldots,0)$ and decryption computes

(1) $e(\prod_{i\in I}(g_1^{\mathbf{M}_i(\alpha\|\mathbf{v})^{\top}}\cdot\mathsf{H}(\pi(i))^{\mathbf{r}'[\rho(i)]})^{\gamma_i},g_2^s)=[s\underbrace{\textstyle\sum_{i\in I}\gamma_i\mathbf{M}_i(\alpha\|\mathbf{v})^{\top}}_{=\alpha}]_T\cdot\underbrace{[s\textstyle\sum_{i\in I}\gamma_i\mathbf{b}[\pi(i)]\mathbf{r}'[\rho(i)]]_T}_{=(2)}$

(2) $\prod_{j\in[\tau]}e(\prod_{i\in I,\rho(i)=j}\mathsf{H}(\pi(i))^{\gamma_i s},g_2^{\mathbf{r}'[j]})=[s\sum_{j\in[\tau]}\sum_{i\in I,\rho(i)=j}\gamma_i\mathbf{b}[\pi(i)]\mathbf{r}'[j]]_T$

Note that by definition of ρ, the second term of (1) is the same as (2). Thus computing (1)/(2) yields $d=[\alpha s]_T$.

Symbolic Security. Since $|\tilde{\mathbf{s}}| = 1$, we need to show that for all $\mathcal{S} \in \mathcal{X}$, $(\mathbf{M}, \pi) \in \mathcal{Y}$ such that $\mathsf{P}(\mathcal{S}, (\mathbf{M}, \pi)) = 0$, it holds that

$$\tilde{\alpha}\tilde{s} \notin \mathrm{span}\big((\tilde{s}\tilde{\mathbf{b}}[u])_{u\in\mathcal{S}} \otimes \tilde{\mathbf{r}} \| (\mathbf{M}_i(\tilde{\alpha}\|\tilde{\mathbf{v}})^\top + \tilde{\mathbf{r}}[\rho(i)]\tilde{\mathbf{b}}[\pi(i)])_{i\in[n_1]} \otimes \tilde{s}\big) \ .$$

We prove this property by contradiction. So assume there exists $\mathbf{e}$, $\mathbf{e}'$ such that

$$\tilde{\alpha}\tilde{s} = (\tilde{s}\tilde{\mathbf{b}}[u] \otimes \tilde{\mathbf{r}})_{u\in\mathcal{S}} \cdot \mathbf{e}^\top + (\mathbf{M}_i(\tilde{\alpha}\|\tilde{\mathbf{v}})^\top \tilde{s} + \tilde{\mathbf{r}}[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{s})_{i\in[n_1]} \cdot \mathbf{e}'^\top \ .$$

Now we use the fact that $\mathsf{P}(\mathcal{S}, (\mathbf{M}, \pi)) = 0$. Recall that this means that there exists a vector $\mathbf{w} \in \mathbb{Z}_p^{n_2}$ such that $\langle \mathbf{w}, \mathbf{M}_i \rangle = 0$ for all $\pi(i) \in \mathcal{S}$ and that $\mathbf{w}[1] = 1$. So evaluating on $(\alpha\|\tilde{\mathbf{v}}) = \mathbf{w}$ gives us

$$\begin{aligned}\tilde{\alpha}\tilde{s} = (\tilde{s}\tilde{\mathbf{b}}[u] \otimes \tilde{\mathbf{r}})_{u\in\mathcal{S}} \cdot \mathbf{e}^\top + (\tilde{\mathbf{r}}[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{s})_{i\in[n_1],\pi(i)\in\mathcal{S}} \cdot \overline{\mathbf{e}}'^\top \\ + (\mathbf{M}_i(\tilde{\alpha}\|\tilde{\mathbf{v}})^\top \tilde{s} + \tilde{\mathbf{r}}[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{s})_{i\in[n_1],\pi(i)\notin\mathcal{S}} \cdot \underline{\mathbf{e}}'^\top \ .\end{aligned}$$

where we split $\mathbf{e}'$ into two vectors $\overline{\mathbf{e}}' \in \mathbb{Z}_p^{|\mathcal{S}|}$ and $\underline{\mathbf{e}}' \in \mathbb{Z}_p^{n_1-|\mathcal{S}|}$, capturing those rows of $\mathbf{M}$ that belong to attributes $u \in \mathcal{S}$ and those that do not belong to an attribute in $\mathcal{S}$. Note that the monomials $\{\tilde{s}\tilde{\mathbf{b}}[u]\otimes\tilde{\mathbf{r}}\}_{u\in\mathcal{S}}$ only appear in the first two terms. By definition of ρ, the monomials $\tilde{\mathbf{r}}[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{s}$ in the second term are mutually distinct. Thus, we must have $\overline{\mathbf{e}}'[i] = -\mathbf{e}[j]$ for all $i \in [n_1]$ such that $\pi(i) \in \mathcal{S}$ and unique indices j, while all other entries in $\mathbf{e}$ must be 0. Therefore, the following equation must be satisfied as well

$$\tilde{\alpha}\tilde{s} = (\mathbf{M}_i(\tilde{\alpha}\|\tilde{\mathbf{v}})^\top \tilde{s} + \tilde{\mathbf{r}}[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{s})_{i\in[n_1],\pi(i)\notin\mathcal{S}} \cdot \underline{\mathbf{e}}'^\top \ .$$

However, this leads to a contradiction since the monomials $(\tilde{\mathbf{r}}[\rho(i)]\tilde{\mathbf{b}}[\pi(i)]\tilde{s})_{\pi(i)\notin\mathcal{S}}$ are mutually distinct and only appear on the RHS.

Corollary 2. *Let $\lambda \in \mathbb{N}$ be the security parameter and $\mathcal{A}$ be an adversary that on input $(1^\lambda, p)$ makes Q_{op} group operation queries to oracles $\mathcal{O}_{\mathsf{add}}$ and $\mathcal{O}_{\mathsf{pair}}$, as well as Q_{ct}, Q_{sk} queries to oracles $\mathcal{O}_{\mathsf{ct}}$, $\mathcal{O}_{\mathsf{sk}}$, and Q_{H} queries to the random oracle* H. *KP-ABE is adaptively secure in the GGM such that*

$$\mathsf{Adv}^{\mathsf{GGM}}_{\mathsf{KP\text{-}ABE},\mathcal{A}}(\lambda) \leq \frac{3 \cdot (Q_{\mathsf{H}} + (|\mathcal{S}| + 2) \cdot Q_{\mathsf{ct}} + (n_1 + 1) \cdot Q_{\mathsf{sk}} + Q_{\mathsf{op}})^2}{p} \ ,$$

where $|\mathcal{S}|$ is the maximum size of the attribute sets queried to $\mathcal{O}_{\mathsf{ct}}$ and n_1 is the maximum number of rows of $\mathbf{M}$ queried to $\mathcal{O}_{\mathsf{sk}}$.

7 Implementation and Evaluation

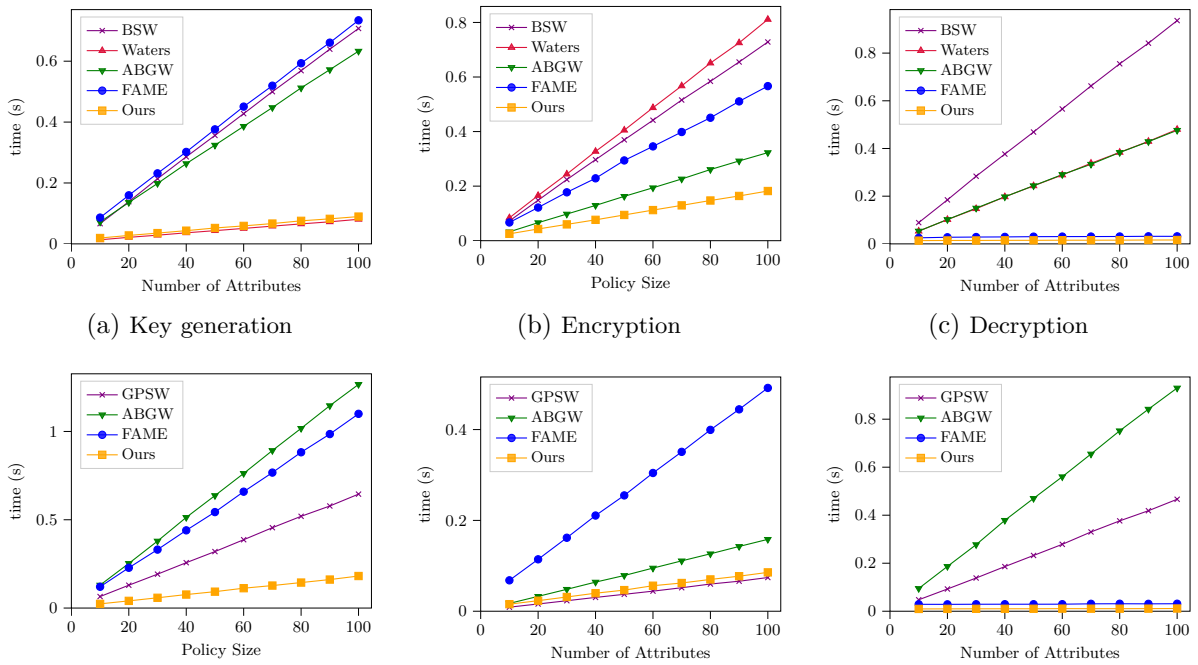
We use two metrics to compare our scheme with prior work, the first is in terms of efficiency and the second is in terms of tightness.

7.1 Efficiency

We implemented several ABE schemes in Python 2.7.12 using the Charm 0.43 framework [4] and the MNT224 curve for pairings.[10] We ran the schemes on a Lenovo Thinkpad Yoga X1 laptop with a 1.80GHz Intel Core i7-10510U CPU and 16GB RAM. Our implementation extends the code of Agrawal and Chase [1] and we provide the implementation on GitHub [44]. In particular, we compare the CP-ABE and KP-ABE schemes described in Table 1.

All schemes are implemented in the asymmetric setting. Agrawal and Chase already transferred the original constructions of BSW, Waters and GPSW that use symmetric bilinear maps to the asymmetric setting [2, Appendices D-F]. Apart from our schemes, we additionally implement the unbounded CP-ABE and KP-ABE of ABGW, see Appendix C for a self-contained description.

In our experiment, we use access policies of the form "Attr_1 **and** Attr_2 **and** ... **and** Attr_N" for $N \in \{10, 20, \ldots, 100\}$ without re-use (i.e., $\tau = 1$). This way, $|\mathcal{S}| = n_1 = n_2 \coloneqq N$ and all attributes are used in decryption. As [2], we first convert the policies into a Boolean formula and then to an MSP using the Lewko-Waters' method [38]. This way, the matrix $\mathbf{M}$ has only entries in $\{0, 1, -1\}$ and the reconstruction coefficients are always 0 or 1, reducing the number of exponentiations.



(a) Key generation (b) Encryption (c) Decryption

(d) Key generation (e) Encryption (f) Decryption

Fig. 3: Running times for CP-ABE (top) and KP-ABE (bottom) schemes. We use one-use formulas (i.e., $\tau = 1$). In particular, for 100 attributes, CP-ABE decryption takes 0.016s in FABEO and 0.032s in FAME, and KP-ABE decryption takes 0.011s in FABEO and 0.031s in FAME.

In Figure 3, we show the average running times for the key generation, encryption and decryption algorithms. For additional reference, the average execution time for different group operations on the MNT224 curve is summarized in Table 2. All our experiments compute the average time in 20 executions. It is worth noting that each algorithm of our two schemes performs better or comparatively the same as all the others. These results are supported by our theoretical overview in Table 3 which lists the number of multiplications and exponentiations for each group as well as the number of hashing and pairing operations. Recall also that exponentiation in $\mathbb{G}_2$ is much slower than in $\mathbb{G}_1$ and the pairing operation is comparatively expensive. Additionally, we provide the number of group elements of secret keys and ciphertexts in Table 5. Since in general elements in $\mathbb{G}_2$ are about 2 to 3 times the size of elements in $\mathbb{G}_1$, our keys and ciphertexts always achieve the same size or even improve considerably upon the other schemes. We provide a more detailed explanation on running times and sizes below.

One-use restriction. FAME has a one-use restriction described in [2, Section 4]. A common way to work around this problem is to make $\hat{\tau}$ copies of each attribute, for some $\hat{\tau}$ chosen at set-up[11]; this way, FAME can support $\hat{\tau}$-use MSPs. The downside of this transformation is that in the CP-ABE, the size of the

[10] The implementations in FAME and ABGW also use the Charm framework. Unfortunately, the PBC library used in Charm does not support BLS12-381.

[11] For FAME and more generally, "unbounded" ABE schemes, this parameter could also be chosen on on a per-key basis during key generation for CP-ABE, or a per-ciphertext basis during encryption for KP-ABE

Groups	*Multiplication*	*Exponentiation*	*Hash*	*Pairing*
$\mathbb{G}_1$	.002	.638	.040	3.836
$\mathbb{G}_2$	.017	4.717	12.342	
$\mathbb{G}_T$	.005	1.078	-	

Table 2: Average time (in milliseconds) for operations on the MNT224 curve.

	Key generation					*Encryption*					*Decryption*		
	$\mathbb{G}_1$			$\mathbb{G}_2$		$\mathbb{G}_1$			$\mathbb{G}_2$		$\mathbb{G}_1$	$\mathbb{G}_T$	
Schemes	Mul	Exp	Hash	Mul	Exp	Mul	Exp	Hash	Mul	Exp	Mul	Mul	Pair
BSW	$m+1$	$m+2$	m	-	m	-	n_1	n_1	-	n_1+1	-	$2I+1$	$2I+1$
Waters	1	$m+1$	-	-	1	n_1	$2n_1$	-	-	n_1+1	I	$I+2$	$I+2$
FAME	$6\hat{\tau}m+9$	$9\hat{\tau}m+9$	$6\hat{\tau}m+6$	-	3	$6n_1n_2+3n_1$	$6n_1$	$6n_1+6n_2$	-	3	$6I+3$	6	6
ABGW	-	-	-	-	$2m+1$	$2n_1$	$5n_1$	-	-	-	$2I$	$I+2$	$I+2$
Ours	1	$m+2$	$m+1$	-	1	n_1	$2n_1$	n_1+1	-	$\tau+1$	$2I$	$\tau+2$	$\tau+2$
GPSW	-	-	-	-	n_1	-	m	-	-	-	-	I	I
FAME	$9n_1n_2+3n_1$	$9n_1+3n_2$	$6n_1+6n_2$	-	3	$3\hat{\tau}m$	$6\hat{\tau}m$	$6\hat{\tau}m$	-	3	$6I$	6	6
ABGW	-	-	-	-	$2n_1$	$2m$	$3m+1$	-	-	-	-	$2I$	$2I$
Ours	n_1	$2n_1$	n_1	-	τ	-	m	m	-	1	$2I$	$\tau+1$	$\tau+1$

Table 3: Number of group operations in $\mathbb{G}_1$ and $\mathbb{G}_2$ for key generation and encryption of CP-ABE (top) and KP-ABE (bottom) schemes. m denotes the number of attributes in the set $\mathcal{S}$, n_1 and n_2 are the number of rows and columns of the MSP matrix and τ denotes the maximum number of multi-use. I denotes the number of attributes used in decryption (counted with multiplicity). Note that $\tau \leq I$. The experiments and most comparison in the text consider $\hat{\tau} = \tau = 1$.

keys grow by a factor of $\hat{\tau}$ though encryption and decryption time are not affected. Similarly, in the KP-ABE, the ciphertexts and encryption time grow by a factor of $\hat{\tau}$. We explicitly account for $\hat{\tau}$ when describing FAME in our comparison tables. For applications where τ may be large and fast decryption is paramount, we can apply the same transformation to our schemes so that decryption only requires 2-3 pairings. For this reason, the experiments and most comparison in the text consider $\hat{\tau} = \tau = 1$. A follow-up to FAME by Tomida, Kawahara and Nishimaki (TKN) [49] shows how to remove the one-use restriction using techniques from [36], paying a multiplicative factor τ in the number of pairings required for decryption, and a much larger security loss in the reduction to DLIN. The TKN scheme essentially coincides with FAME when $\tau = 1$, and for larger τ, remains at least 2-3 times less efficient than FABEO. All of our experiments are for $\tau = 1$, hence the omission of TKN.

Setup. In Table 4, we show the setup time of all schemes listed in our evaluation. For schemes, where the universe size is bounded, we used the minimal bound N. Both our schemes have the shortest setup times (around 0.14s), however all schemes that support large universes are almost equally fast. The universe size of Waters and GPSW are bounded, thus the time increases with the universe size and for $N = 100$, they are about 3-7 times slower than the other schemes.

CP-ABE. Looking at running times of CP-ABE schemes given in Figure 3, only the key generation time of Waters CP-ABE can compete with ours, being only slightly faster (<0.01s for all test samples). This is due to the fact that the number of operations performed by Waters and our scheme are essentially the same. Whereas for our scheme it takes only 0.09s to generate a key for a set of 100 attributes, it takes at least 7 times longer for BSW, ABGW and FAME. This can be explained by that the fact that ABGW and BSW both perform exponentiations in $\mathbb{G}_2$ that are linear in the number of attributes (cf. Table 3). Also it is easy to see that FAME is around 9 times slower than our scheme because of the number of exponentiations in $\mathbb{G}_1$.

Scheme	*Uni size*	*Time (s)*
BSW	-	0.025
Waters	100	0.096
ABGW	-	0.016
FAME	-	0.030
Ours	-	0.014

Scheme	*Uni size*	*Time (s)*
GPSW	100	0.095
ABGW	-	0.015
FAME	-	0.030
Ours	-	0.013

Table 4: Setup times for CP-ABE (left) and KP-ABE (right) schemes.

	Key size		*Ciphertext size*	
Schemes	$\mathbb{G}_1$	$\mathbb{G}_2$	$\mathbb{G}_1$	$\mathbb{G}_2$
BSW	$m+1$	m	n_1	n_1+1
Waters	$m+1$	1	n_1	n_1+1
FAME	$3\hat{\tau}m+3$	3	$3n_1$	3
ABGW	-	$m+2$	$3n_1$	-
Ours	$m+1$	1	n_1	$\tau+1$
GPSW	-	n_1	m	-
FAME	$3n_1$	3	$3\hat{\tau}m$	3
ABGW	-	$2n_1$	$2m$	-
Ours	n_1	τ	m	1

Table 5: Key and ciphertext sizes of CP-ABE (top) and KP-ABE (bottom) schemes. The columns $\mathbb{G}_1$ and $\mathbb{G}_2$ denote the number of elements in the respective group (in general, $|\mathbb{G}_2| \geq 2|\mathbb{G}_1|$). m denotes the number of attributes in the set $\mathcal{S}$, n_1 and n_2 are the number of rows and columns of the MSP matrix and τ denotes the maximum number of multi-use. The experiments consider $\hat{\tau} = \tau = 1$.

The encryption times vary a bit more, with our scheme being the fastest (0.18s for policies of 100 attributes), followed by ABGW and FAME, which are around 2 and 3 times slower. This also shows up in Table 3 and the number of exponentiations in $\mathbb{G}_1$. BSW and Waters both perform n_1 number of exponentiations in $\mathbb{G}_2$, therefore they are the least efficient schemes here.

FAME is the only other scheme that supports fast decryption which does not depend on the number of attributes. Without multi-use of attributes, our scheme is still 2 times faster than FAME. We can see this also in Table 3, since FAME performs 6 pairing operations and our scheme 3 (for $\tau = 1$). For both FAME and our scheme, decryption that uses 100 attributes takes less than 0.04s, whereas it takes almost 0.5s for ABGW and Waters. Those schemes need to perform a pairing operation for each attribute that is used in decryption.

KP-ABE. Looking at KP-ABE, the key generation time of our scheme is much lower than for all the others. Compared to GPSW and ABGW this can be explained by the fact that these two schemes perform all exponentiations in $\mathbb{G}_2$ instead of $\mathbb{G}_1$, whereas our scheme only performs τ (and therefore 1 operation in our experiment) in $\mathbb{G}_2$. On the other hand, the number of exponentiations in FAME depends on n_1 and n_2 and thus is already around 6 times higher than for our scheme. Therefore, in our scheme it takes less than 0.2s for a policy size of 100, whereas GPSW needs more than half a second and FAME and ABGW more than a second. In terms of encryption time, GPSW is slightly below that of our scheme (around 0.01s), but both are around twice as fast as ABGW and considerably faster than FAME. This can be justified by the constant factors in the number of exponentiations in $\mathbb{G}_1$ in Table 3. For decryption we get the same results as in the CP-ABE case since GPSW and ABGW do not support fast decryption and need to perform pairing operations that are linear in the number of attributes.

Key and Ciphertext Sizes. In Table 5, we also show the number of group elements of secret keys and ciphertexts. Note that in general, elements in $\mathbb{G}_2$ are about 2 to 3 times the size of elements in $\mathbb{G}_1$. The secret keys of our CP-ABE scheme consist of $m+1$ elements in $\mathbb{G}_1$ and only 1 element in $\mathbb{G}_2$, which is the same as in Waters. BSW uses fresh randomness for all attributes, thus their keys additionally include m elements in $\mathbb{G}_2$. The number of group elements of keys in ABGW is the same as for our scheme, however all group elements are in $\mathbb{G}_2$, which makes secret keys about three times larger. Secret keys in FAME are also three times larger since they consist of more group elements. The ciphertext sizes of BSW and Waters are exactly the same, requiring about the same number of elements in $\mathbb{G}_1$ and $\mathbb{G}_2$. Our scheme allows randomness reuse and therefore only needs $\tau + 1$ additional elements in $\mathbb{G}_2$. ABGW and FAME both need around 3 times more elements than our scheme. Looking at KP-ABE, the advantage of our scheme is that most elements of the secret keys are in $\mathbb{G}_1$, as is the case for FAME, which again requires around 3 times more elements. Secret keys in GPSW and ABGW contain only elements in $\mathbb{G}_2$. The ciphertext of our scheme is comparable to that of GPSW and about 2 resp. 3 times smaller than those of ABGW and FAME.

Resources			*Bit Security*			
t	Q_{sk}	Q_{ct}	ABGW	FAME	BSW	Ours
2^{40}	2^{20}	2^{20}	2^{-176}	2^{-176}	2^{-196}	2^{-216}
2^{40}	2^{10}	2^{30}	2^{-176}	2^{-176}	2^{-196}	2^{-216}
2^{60}	2^{30}	2^{30}	2^{-136}	2^{-136}	2^{-166}	2^{-196}
2^{60}	2^{20}	2^{40}	2^{-136}	2^{-136}	2^{-156}	2^{-196}
2^{80}	2^{40}	2^{40}	2^{-96}	2^{-96}	2^{-136}	2^{-176}
2^{80}	2^{30}	2^{50}	2^{-96}	2^{-96}	2^{-126}	2^{-176}
2^{128}	2^{40}	2^{40}	2^{-48}	2^{-48}	2^{-88}	2^{-128}
2^{128}	2^{30}	2^{50}	2^{-48}	2^{-48}	2^{-78}	2^{-128}

Table 6: Bit security of ABE schemes depending on the adversary's running time t and number of secret key queries Q_{sk} and ciphertext queries Q_{ct}. Bit security is defined as Adv/t, where we use $p = 2^{256}$. The values coincide for CP-ABE and KP-ABE schemes. For ABGW and FAME we use $\mathsf{Adv} = O(Q_{\mathsf{ct}} Q_{\mathsf{sk}} t^2/p)$, for BSW we use $\mathsf{Adv} = O(Q_{\mathsf{ct}} t^2/p)$ and for ours we use $\mathsf{Adv} = O(t^2/p)$ (cf. evaluation).

7.2 Bit Security based on Tightness

Whereas considering multiple secret key queries in the security definition is considered standard in terms of ABE security, we additionally consider many ciphertext or challenge queries in our security proof. The two definitions are polynomially equivalent, but the non-trivial implication from one to many ciphertexts incurs a security loss linear in the number of ciphertext queries. On the contrary, if the security loss is only constant, we say that the bound is tight, as is the case for our bounds. The security loss plays an important role in choosing the system parameters of the scheme, e.g., the size of the underlying pairing group which provides a determined level of security, which is usually stated in bits. Further, we can define the success ratio of an adversary $\mathcal{A}$ by its advantage Adv and its running time t. For λ-bit security, we then require that $\mathsf{Adv}/t \leq 2^{-\lambda}$. From this value, we can then deduce whether a concrete instantiation provides the desired security level.

In Table 6, we compute the bit security of our scheme, as well as ABGW, FAME and BSW in different scenarios, that is we use different numbers of secret key and ciphertext queries. The running time t captures the *offline* time of an adversary, e.g. to perform group operations or also to evaluate a hash function (thus including random oracle queries). We assume t to be rather large, whereas secret key and ciphertext queries are considered *online* running time and therefore considerably lower. The advantage also depends on the order of the underlying group and for our comparison we assume $p = 2^{256}$. Since a discrete logarithm attack on the elliptic curve group yields a bound $O(t^2/p)$, this parameter choice is based on a security level of around 128 bit and this should be the target for the bit security of the ABE schemes as well.

We consider four different scenarios from small-scale to large-scale adversaries, based on the running time $t \in \{2^{40}, 2^{60}, 2^{80}, 2^{128}\}$. For each scenario, we choose the number of secret key queries Q_{sk} and ciphertexts Q_{ct} accordingly, once for $Q_{\mathsf{ct}} = Q_{\mathsf{sk}}$ and once for $Q_{\mathsf{sk}} < Q_{\mathsf{ct}}$, since in practice an adversary may easily observe a large number of ciphertexts, rather than a large number of keys.

Evaluation. We omit Waters and GSPW here as those schemes are only selectively secure. The numbers in Table 6 are based on the security bounds stated in the corresponding papers as well as an additional hybrid argument on the number of ciphertexts as mentioned above. In particular, the security bounds that we use in our comparison in Table 6 can be explained as follows.

- ABGW: Similar to our work, ABGW uses a pair encoding approach and proves security in the the GGM. However, the use of rational fractions does not allow for the optimal bound and they get an advantage of $O(\frac{Q_{\mathsf{ct}} Q_{\mathsf{sk}} t^2}{p})$, where the additional factor Q_{ct} comes from the hybrid argument.
- FAME: The schemes achieve security under the DLIN assumption using the dual system encryption framework, which also incurs a security loss in the number of secret keys. Since they also only consider the single-ciphertext setting, they achieve the same bound as ABGW, namely $O(\frac{Q_{\mathsf{ct}} Q_{\mathsf{sk}} t^2}{p})$.

– BSW: The scheme in BSW is proven secure in the GGM and single-ciphertext setting, thus there is no security loss in the number of secret keys. We get $O(\frac{Q_{\mathsf{ct}} t^2}{p})$ to account for many ciphertexts as well.

– Ours: Since we prove security directly in the many-ciphertext setting and in the GGM, we achieve the optimal bound $O(\frac{t^2}{p})$.

Although all schemes meet the target bound in a small to medium-scale scenario (cf. Table 6), one can already observe the gap between the theoretical security level of ABGW or FAME and that of our schemes, which comes from their lose bounds. This becomes now relevant in the large-scale scenarios. For $t = 2^{80}$, $Q_{\mathsf{ct}} = Q_{\mathsf{sk}} = 2^{40}$, both ABGW and FAME only provide a security level of 2^{96} bits, compared to the target of 2^{128}. BSW and our scheme both still meet the target, however BSW is slightly below the target when increasing the number of ciphertexts to 2^{50}, since their bound depends on t and Q_{ct}. When allowing for a running time of 2^{128}, then ABGW and FAME only provide 48 bits of security and therefore not suitable for applications in a large-scale scenario. BSW still achieves 78 resp. 88 bits, which may be sufficient for some applications. Due to the tight bound, our scheme exactly meets the target of 128 bits.

Exact bounds. The numbers in Table 6 are computed based on the most dominating terms in the bounds as described above. In fact, when looking at our security statement, our bounds also depend on the number of secret keys and ciphertext and even their corresponding sizes (cf. terms $|\mathsf{ct}|Q_{\mathsf{ct}}$ and $|\mathsf{sk}|Q_{\mathsf{sk}}$ in Theorem 1). However, since $|\mathsf{ct}|Q_{\mathsf{ct}} + |\mathsf{sk}|Q_{\mathsf{sk}} \ll t$, these do not have a huge impact on the exact security of the schemes. For $n_1 = m = 100$, we lose less than two bits of security. The same applies to BSW. Due to the rational fractions used in ABGW, the ciphertext and key sizes are actually an additional multiplicative factor in the bound, i.e., the dominating term is $|\mathsf{ct}| \cdot |\mathsf{sk}| \cdot Q_{\mathsf{ct}} Q_{\mathsf{sk}} t^2$. For large policies this has a considerable impact on the security. In this case, we lose 19 bits of security when $n_1 = m = 100$. Compared to the values in Table 6, a small security loss of around 3 bits also occurs in FAME, which is due to constant factors.

8 Extensions

In this section, we briefly describe how we can extend our definition of PES-ABE to capture more schemes, e.g., ABE for deterministic finite automata (DFA).

8.1 A variant of PES-ABE

We want to capture PES-ABE schemes as in ABGW with Setup_0, Enc_0, KeyGen_0 as before, except:

$$\begin{aligned} \mathsf{mpk} &:= ([\mathbf{b}]_1, [\alpha]_T), \\ \mathsf{msk} &:= (\mathbf{b}, \alpha), \\ \mathsf{ct} &:= ([\mathbf{c}^1]_1, [\mathbf{c}^2]_1), \\ \mathsf{sk} &:= ([\mathbf{k}^1]_2, [\mathbf{k}^2]_2) \end{aligned}$$

For such schemes, we impose an additional constraint on k^1, k^2 as with ABGW, namely that $\mathbf{r} = (\mathbf{r}' \| \mathbf{r}'')$ and $k^1(\alpha, \mathbf{r}'', \mathbf{b} \otimes \mathbf{r}'), k^2(\mathbf{r}')$ (that is, we removed $\mathbf{r}', \mathbf{b} \otimes \mathbf{r}''$ from the input to k^1 and $\mathbf{r}''$ from the input to k^2).[12] This way, we can ensure that $\mathrm{span}\big(\mathbf{k}^1\big) \cap \mathrm{span}\big(\mathbf{k}^2\big) = \{0\}$, which we will need in the proof of strong symbolic security.

Strong Symbolic Security (Variant). For all $Q_{\mathsf{ct}}, Q_{\mathsf{sk}} \in \mathbb{N}$, $X \in \mathcal{X}^{Q_{\mathsf{ct}}}, Y \in \mathcal{Y}^{Q_{\mathsf{sk}}}$ such that $\mathsf{P}(X[i], Y[j]) = 0$ for all $i \in [Q_{\mathsf{ct}}]$, $j \in [Q_{\mathsf{sk}}]$, we have

$$\mathrm{span}\big(\tilde{\alpha} \otimes \mathbf{c}_X^2\big) \cap \mathrm{span}\big(\tilde{\alpha} \parallel (1 \| \tilde{\mathbf{b}} \| \mathbf{c}_X^1 \| \mathbf{c}_X^2) \otimes (1 \| \mathbf{k}_Y^1 \| \mathbf{k}_Y^2)\big) = \{0\} \ ,$$

where $X, Y, \mathbf{c}_X^1, \mathbf{c}_X^2, \mathbf{k}_Y^1, \mathbf{k}_Y^2$ are as in Definition 3.

[12] ABGW refers to $\mathbf{r}'$ as the non-lone variables and $\mathbf{r}''$ as the lone variables. Also, ABGW considers a more general setting for c^1, c^2 with $\mathbf{s} = (\mathbf{s}' \| \mathbf{s}'')$ and $c^2(\mathbf{s}'), c^1(\mathbf{s}'', \mathbf{b} \otimes \mathbf{s}')$. To the best of our knowledge, none of the existing ABE schemes exploit this generalization.

Claim. If PES-ABE satisfies $(1,1)$-symbolic security (Definition 2), then it also satisfies the variant of strong symbolic security.

Proof. The proof is similar to that of Lemma 1. Step 1 and step 2 already tell us that $(1,1)$ symbolic security implies that

$$\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}_X^2\big) \cap \operatorname{span}\big(\mathbf{c}_X^1 \otimes \mathbf{k}_Y^2 \| \mathbf{k}_Y^1 \otimes \mathbf{c}_X^2\big) = \{0\} \ .$$

We need to slightly modify the last step of the proof. We also prove the claim by contradiction. So assume there exist $\mathbf{e}^*$, $\mathbf{e}^{(1)}$, $\mathbf{e}^{(2)}$, $\mathbf{e}^{(3)}$ such that

$$\begin{aligned}(\tilde{\alpha} \otimes \mathbf{c}_X^2) \cdot \mathbf{e}^{*\top} = (\tilde{\alpha} \| \mathbf{c}_X^1 \| \mathbf{c}_X^2 \| (1 \| \tilde{\mathbf{b}}) \otimes (1 \| \mathbf{k}_Y^1 \| \mathbf{k}_Y^2)) \cdot \mathbf{e}^{(1)\top} + (\mathbf{c}_X^1 \otimes \mathbf{k}_Y^1 \| \mathbf{c}_X^2 \otimes \mathbf{k}_Y^2) \cdot \mathbf{e}^{(2)\top} \\ + (\mathbf{c}_X^1 \otimes \mathbf{k}_Y^2 \| \mathbf{k}_Y^1 \otimes \mathbf{c}_X^2) \cdot \mathbf{e}^{(3)\top}\end{aligned}$$

In the same way as in the proof of Lemma 1, we can show that the first term evaluates to 0. It remains to show that also the second term evaluates to 0, which follows from the fact that

$$\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}_X^2\big) \cap \operatorname{span}\big(\mathbf{c}_X^1 \otimes \mathbf{k}_Y^1 \| \mathbf{c}_X^2 \otimes \mathbf{k}_Y^2\big) = \{0\}$$

together with

$$\operatorname{span}\big(\mathbf{c}_X^1 \otimes \mathbf{k}_Y^2 \| \mathbf{c}_X^2 \otimes \mathbf{k}_Y^1\big) \cap \operatorname{span}\big(\mathbf{c}_X^1 \otimes \mathbf{k}_Y^1 \| \mathbf{c}_X^2 \otimes \mathbf{k}_Y^2\big) = \{0\}$$

The first equation holds since $\operatorname{span}\big(\mathbf{c}_X^1 \otimes \mathbf{k}_Y^1\big)$ contains monomials of the form $(\tilde{\mathbf{s}}_X \otimes \tilde{\mathbf{b}}) \otimes (\alpha \| \tilde{\mathbf{r}}_Y'' \| \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}_Y')$ and $\operatorname{span}\big(\mathbf{c}_X^2 \otimes \mathbf{k}_Y^2\big)$ contains monomials of the form $\tilde{\mathbf{s}} \otimes \tilde{\mathbf{r}}'$, which all do not appear in $\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}_X^2\big)$.

The second equation holds since $\operatorname{span}\big(\mathbf{c}_X^1 \otimes \mathbf{k}_Y^2\big)$ contains monomials of the form $\tilde{\mathbf{s}}_X \otimes \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}_Y'$ and $\operatorname{span}\big(\mathbf{c}_X^2 \otimes \mathbf{k}_Y^1\big)$ contains monomials of the form $\tilde{\mathbf{s}}_X \otimes (\alpha \| \tilde{\mathbf{r}}_Y'' \| \tilde{\mathbf{b}} \otimes \tilde{\mathbf{r}}_Y')$, which all do not appear on the RHS of $\cap$. Note that here we use that $\operatorname{span}\big(\tilde{\mathbf{r}}'\big) \cap \operatorname{span}\big(\tilde{\mathbf{r}}''\big) = 0$.

8.2 ABE for DFA

We consider the ABE scheme for DFAs in [27, equation (1)] (building on [52]). Recall that a DFA is specified by a tuple (Q, Σ, δ, F) where the state space is $[Q] := \{1, 2, \ldots, Q\}$; 1 is the unique start state; $F \subseteq [Q]$ is the set of accept states, and $\delta : [Q] \times \Sigma \to [Q]$ is the state transition function.

We provide a self-contained overview of our ABE scheme for DFA in Figure 4. In the following, we describe the underlying PES-ABE.

- $\underline{\mathsf{Setup}_0}$. Output $n := 3 + |\Sigma|$, where we parse $\mathbf{b}$ as $(w_{\text{start}}, w_{\text{end}}, z, \{w_\sigma\}_{\sigma \in \Sigma})$.
- $\underline{\mathsf{Enc}_0(x)}$. Set $w = \ell + 1$, $w_1 = \ell + 2$, $w_2 = \ell + 1$, and output (c^1, c^2) where we parse $\mathbf{s}$ as $(s_\ell, s_0, s_1, \ldots, s_{\ell-1})$ and

$$\begin{aligned}c^1(\mathbf{s} \otimes \mathbf{b}) &:= (s_0 w_{\text{start}} \parallel \{s_{i-1} z + s_i w_{x_i}\}_{i \in [\ell]} \parallel s_\ell w_{\text{end}}), \\ c^2(\mathbf{s}) &:= (\mathbf{s})\end{aligned}$$

- $\underline{\mathsf{KeyGen}_0(Q, \Sigma, \delta, F)}$. Set $m = 2Q$, $m_1 = 1 + Q + Q \cdot |\Sigma| + |F|$, $m_2 = Q$, and output (k^1, k^2) where we parse $\mathbf{r} = (\mathbf{r}' \| \mathbf{r}'') := (\{r_u\}_{u \in [Q]} \parallel \{d_u\}_{u \in [Q]})$ and

$$\begin{aligned}k^1(\alpha, \mathbf{r}'', \mathbf{b} \otimes \mathbf{r}') := (d_1 + w_{\text{start}} r_1 \parallel \{-d_u + z r_u\}_{u \in [Q]} \parallel \{d_{\delta(u,\sigma)} + w_\sigma r_u\}_{u \in [Q], \sigma \in \Sigma} \parallel \\ \{\alpha - d_u + w_{\text{end}} r_u\}_{u \in F}) \\ k^2(\mathbf{r}) := (\mathbf{r}')\end{aligned}$$

In applications, think of $\ell \gg |\Sigma|, Q$. We note that our scheme differs from Waters' scheme in that we reuse r_u for all the transitions starting from u instead of a fresh $r_{u,\sigma}$ for each (u, σ). This modification yields a smaller secret key (cf. Table 7).

We now prove correctness and symbolic security.

$\underline{\mathsf{Setup}(1^\lambda)}$. Run $\mathsf{GroupGen}(1^\lambda)$ to obtain $\mathcal{G} \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$. Pick $\alpha, w_{\text{start}}, w_{\text{end}}, z \xleftarrow{\$} \mathbb{Z}_p$, $\{w_\sigma\}_{\sigma\in\Sigma} \xleftarrow{\$} \mathbb{Z}_p^{|\Sigma|}$. Compute the master public key as

$$\mathsf{mpk} \coloneqq (\mathcal{G}, g_1^{w_{\text{start}}}, g_1^{w_{\text{end}}}, g_1^z, \{g_1^{w_\sigma}\}_{\sigma\in\Sigma}, e(g_1, g_2)^\alpha)$$

Let $\mathsf{msk} \coloneqq (\alpha, w_{\text{start}}, w_{\text{end}}, z, \{w_\sigma\}_{\sigma\in\Sigma})$ be the master secret key.

$\underline{\mathsf{KeyGen}(\mathsf{msk}, (Q, \Sigma, \delta, F))}$. Pick $\{r_u\}_{u\in[Q]} \xleftarrow{\$} \mathbb{Z}_p^Q$, $\{d_u\}_{u\in[Q]} \xleftarrow{\$} \mathbb{Z}_p^Q$. Compute

$$\mathsf{sk}_0 \coloneqq g_2^{d_1 + w_{\text{start}} r_1} \qquad \mathsf{sk}_{1,u} \coloneqq g_2^{r_u} \ ,$$

for all $u \in [Q]$. Then compute

$$\mathsf{sk}_{2,u} \coloneqq g_2^{-d_u + z r_u} \qquad \mathsf{sk}_{3,u,\sigma} \coloneqq g_2^{d_v + w_\sigma r_u},$$

for all $u \in [Q]$, $\sigma \in \Sigma$, where $v = \delta(u, \sigma)$, and

$$\mathsf{sk}_{4,u} = g_2^{\alpha - d_u + w_{\text{end}} r_u}$$

for all $u \in F$. Output $\mathsf{sk} \coloneqq (\mathsf{sk}_0, (\mathsf{sk}_{1,u})_{u\in[Q]}, (\mathsf{sk}_{2,u}, \mathsf{sk}_{3,u,\sigma})_{u\in[Q],\sigma\in\Sigma}, (\mathsf{sk}_{4,u})_{u\in F})$.

$\underline{\mathsf{Enc}(\mathsf{mpk}, x)}$. Pick $(s_\ell, s_0, s_1, \ldots, s_{\ell-1}) \xleftarrow{\$} \mathbb{Z}_p^{\ell+1}$, where $\ell = |x|$. Compute

$$\mathsf{ct}_0 \coloneqq g_1^{s_0} \qquad \mathsf{ct}_1 \coloneqq g_1^{s_0 w_{\text{start}}} \qquad \mathsf{ct}_{2,i} \coloneqq g_1^{s_i} \qquad \mathsf{ct}_{3,i} \coloneqq g_1^{s_{i-1} z + s_i w_{x_i}} \qquad \mathsf{ct}_4 \coloneqq g_1^{s_\ell w_{\text{end}}}$$

for all $i \in [\ell]$. Output $\mathsf{ct} \coloneqq (\mathsf{ct}_0, \mathsf{ct}_1, (\mathsf{ct}_{2,i}, \mathsf{ct}_{3,i})_{i\in[\ell]}, \mathsf{ct}_4)$ and $d \coloneqq e(g_1, g_2)^{\alpha s_\ell}$.

$\underline{\mathsf{Dec}(\mathsf{mpk}, x, (Q, \Sigma, \delta, F), \mathsf{ct}, \mathsf{sk})}$. If (Q, Σ, δ, F) accepts x, then there exists a sequence $u_1, \ldots, u_\ell \in Q$ such that $u_1 = 1$, $\delta(u_i, x_i) = u_{i+1}$ $\forall i \in [\ell - 1]$ and $\delta(u_\ell, x_\ell) \in F$. Compute

$$B \coloneqq \frac{e(\mathsf{ct}_0, \mathsf{sk}_0)}{e(\mathsf{ct}_1, \mathsf{sk}_{1,1})} \cdot \prod_{i\in[\ell]} \frac{e(\mathsf{ct}_{2,i-1}, \mathsf{sk}_{2,u_i}) \cdot e(\mathsf{ct}_{2,i}, \mathsf{sk}_{3,u_i,x_i})}{e(\mathsf{ct}_{3,i}, \mathsf{sk}_{1,u_i})}$$

and output

$$B \cdot \frac{e(\mathsf{ct}_{2,\ell}, \mathsf{sk}_{4,\delta(u_\ell,x_\ell)})}{e(\mathsf{ct}_4, \mathsf{sk}_{1,\delta(u_\ell,x_\ell)})} \ .$$

Fig. 4: Our KP-ABE scheme for DFA.

	Key size		*Ciphertext size*	
Schemes	$\mathbb{G}_1$	$\mathbb{G}_2$	$\mathbb{G}_1$	$\mathbb{G}_2$
Waters	-	$3Q\lvert\Sigma\rvert + 2\lvert F\rvert + 2$	$2\ell + 3$	-
Ours	-	$Q\lvert\Sigma\rvert + Q + \lvert F\rvert + 1$	$2\ell + 3$	-

Table 7: Key and ciphertext sizes of ABE schemes for DFA.

Correctness. Let ct be a ciphertext for x and sk be a secret key for (Q, Σ, δ, F) as defined in Figure 4. If (Q, Σ, δ, F) accepts x, then there exists a sequence $u_1, \ldots, u_\ell \in Q$ such that $u_1 = 1$, $\delta(u_i, x_i) = u_{i+1}$ $\forall i \in [\ell - 1]$ and $\delta(u_\ell, x_\ell) \in F$. Then decryption computes

(1) $\frac{e(\mathsf{ct}_0, \mathsf{sk}_0)}{e(\mathsf{ct}_1, \mathsf{sk}_{1,1})} = [s_0(d_1 + w_{\text{start}} r_1) - s_0 w_{\text{start}} r_1]_T = [s_0 d_1]_T$

(2) $\frac{e(\mathsf{ct}_{2,i-1}, \mathsf{sk}_{2,u_i}) \cdot e(\mathsf{ct}_{2,i}, \mathsf{sk}_{3,u_i,x_i})}{e(\mathsf{ct}_{3,i}, \mathsf{sk}_{1,u_i})} = [s_{i-1}(-d_{u_i} + z r_{u_i}) + s_i(d_{\delta(u_i,x_i)} + w_{x_i} r_{u_i}) - (s_{i-1} z + s_i w_{x_i}) r_{u_i}]_T$
$= [s_i d_{\delta(u_i,x_i)} - s_{i-1} d_{u_i}]_T$

(3) $\frac{e(\mathsf{ct}_{2,\ell}, \mathsf{sk}_{4,\delta(u_\ell,x_\ell)})}{e(\mathsf{ct}_4, \mathsf{sk}_{1,\delta(u_\ell,x_\ell)})} = [s_\ell(\alpha - d_{\delta(u_\ell,x_\ell)} + w_{\text{end}} r_{\delta(u_\ell,x_\ell)}) - s_\ell w_{\text{end}} r_{\delta(u_\ell,x_\ell)}]_T = [\alpha s_\ell - s_\ell d_{\delta(u_\ell,x_\ell)}]_T$

Correctness follows from the fact that $(1) \cdot \prod_{i\in[\ell]} (2) \cdot (3)$ yields $d = [\alpha s_\ell]_T$.

Symbolic security. We need to show that for all x and all (Q, Σ, δ, F) such that $\mathsf{P}(x, (Q, \Sigma, \delta, F)) = 0$, it holds that

$$\operatorname{span}\big(\tilde{\alpha} \otimes \mathbf{c}^2\big) \cap \operatorname{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2 \parallel \mathbf{k}^1 \otimes \mathbf{c}^2\big) = \{0\} \ .$$

W.l.o.g. one can assume that $|F| = 1$, i.e., there is only one accepting state and that it has no outgoing transitions. In the following, we will denote the accepting state by ac. Further, we will denote the sequence of states on input x by $(u_1, \ldots, u_\ell, \mathsf{fin})$, where $u_1 = 1$ is the start state, $u_{i+1} = \delta(u_i, x_i)$ for $i \in [\ell]$ is the state reached upon reading $x_1, \ldots, x_i$, and $\mathsf{fin} = \delta(u_\ell, x_\ell)$ is the final state.

We will prove the above claim by contradiction. Thus, assume on the contrary that

$$\mathrm{span}\big(\tilde{\alpha} \otimes \mathbf{c}^2\big) \cap \mathrm{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2 \parallel \mathbf{k}^1 \otimes \mathbf{c}^2\big) \neq \{0\} \ .$$

We now proceed in three steps and make the following claims.

- Claim 1: $\mathrm{span}\big(\tilde{\alpha} \otimes (\tilde{s}_0 \| \cdots \| \tilde{s}_{\ell-1})\big) \cap \mathrm{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2 \parallel \mathbf{k}^1 \otimes \mathbf{c}^2\big) = \{0\}$.
- Claim 2: If $\tilde{\alpha}\tilde{s}_\ell \in \mathrm{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2 \| \mathbf{k}^1 \otimes \mathbf{c}^2\big)$, then
$$\tilde{s}_\ell \tilde{d}_{\mathsf{ac}} \in \mathrm{span}\big(\tilde{s}_0 \tilde{d}_1 \| \{\tilde{s}_i \tilde{d}_{\delta(u,x_i)} - \tilde{s}_{i-1}\tilde{d}_u\}_{i\in[\ell],u\in[Q]}\big) \ .$$
- Claim 3: $\tilde{s}_\ell \tilde{d}_{\mathsf{ac}} \notin \mathrm{span}\big(\tilde{s}_0 \tilde{d}_1 \| \{\tilde{d}_{\tilde{s}_i \delta(u,x_i)} - \tilde{s}_{i-1}\tilde{d}_u\}_{i\in[\ell],u\in[Q]}\big)$.

Claim 1 basically tells us that we can ignore all terms in $\mathrm{span}\big(\tilde{\alpha} \otimes \mathbf{c}^2\big)$ except for $\tilde{\alpha}\tilde{s}_\ell$. Claim 2 shows that the only degree two monomials we learn from $\mathrm{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2 \| \mathbf{k}^1 \otimes \mathbf{c}^2\big)$ are linear combinations of the terms $\tilde{s}_i \tilde{d}_{\delta(u,x_i)} - \tilde{s}_{i-1}\tilde{d}_u$. Claim 3 then yields a contradiction to claims 1 and 2, thus proving symbolic security.

Step 1. To prove claim 1, we look at the monomials on the RHS of $\cap$. Note that the terms $\tilde{\alpha}\tilde{s}_i$ for $i \in \{0, \ldots, \ell-1\}$ only appear together with $\tilde{w}_{\mathrm{end}}\tilde{r}_{\mathsf{ac}}\tilde{s}_i$. However, these monomials do not appear anywhere else, since $\tilde{w}_{\mathrm{end}}$ only appears in $\tilde{w}_{\mathrm{end}}\tilde{r}_u\tilde{s}_\ell$ and the above equation must hold.

Step 2. Assume

$$\tilde{\alpha}\tilde{s}_\ell \in \mathrm{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2 \| \mathbf{k}^1 \otimes \mathbf{c}^2\big) \ . \tag{8}$$

In order to prove claim 2, we want to look at the monomials on the RHS. The definition of the scheme tells us that

$$\mathrm{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2\big) = \mathrm{span}\big(\{\underline{\tilde{w}_{\mathrm{start}}\tilde{s}_0\tilde{r}_u \| \tilde{w}_{\mathrm{end}}\tilde{s}_\ell\tilde{r}_u}\}_{u\in[Q]} \| \{(\tilde{z}\tilde{s}_{i-1} + \tilde{w}_{x_i}\tilde{s}_i)\tilde{r}_u\}_{u\in[Q],i\in[\ell]}\big)$$

and

$$\begin{aligned}\mathrm{span}\big(\mathbf{k}^1 \otimes \mathbf{c}^2\big) = \mathrm{span}\big((\tilde{\alpha} - \tilde{d}_{\mathsf{ac}} + \underline{\tilde{w}_{\mathrm{end}}\tilde{r}_{\mathsf{ac}}}) \otimes (\underline{\tilde{s}_0 \| \cdots \| \tilde{s}_{\ell-1}} \| \tilde{s}_\ell) \| (\tilde{d}_1 + \underline{\tilde{w}_{\mathrm{start}}\tilde{r}_1}) \otimes (\tilde{s}_0 \| \underline{\tilde{s}_1 \| \cdots \| \tilde{s}_\ell}) \| \\ (\{-\tilde{d}_u + \tilde{z}\tilde{r}_u \| \underbrace{\tilde{d}_{\delta(u,\sigma)} + \tilde{w}_\sigma \tilde{r}_u}\}_{u\in[Q],\sigma\in\Sigma}) \otimes \tilde{\mathbf{s}}\big)\end{aligned}$$

where terms that are underlined will be removed in the next step. We also show that some of the terms that are underlined with a wavy line can also be removed. We will explain the details in the next paragraph.

In particular, we first look at the terms containing $\tilde{w}_{\mathrm{start}}$ and $\tilde{w}_{\mathrm{end}}$. Only the monomial $\tilde{w}_{\mathrm{start}}\tilde{r}_1\tilde{s}_0$ appears more than once, so we can remove all terms with $\tilde{w}_{\mathrm{start}}\tilde{r}_u\tilde{s}_i$ for $i \in [\ell], u \in [Q] \setminus \{1\}$ since their coefficients must be 0. Also note that we can ignore $\tilde{w}_{\mathrm{start}}\tilde{r}_1\tilde{s}_0$ in $(\tilde{d}_1 + \tilde{w}_{\mathrm{start}}\tilde{r}_1)\tilde{s}_0$ and in $\mathrm{span}\big(\mathbf{c}^1 \otimes \mathbf{k}^2\big)$ since their coefficients must always match to cancel out. The same applies for $\tilde{w}_{\mathrm{end}}\tilde{r}_{\mathsf{ac}}\tilde{s}_\ell$. Monomials $\tilde{w}_{\mathrm{end}}\tilde{r}_u\tilde{s}_i$ for $u \in [Q] \setminus \{\mathsf{ac}\}$ and $i \in [0, \ell-1]$ do not appear anywhere else and we can also ignore $\tilde{w}_{\mathrm{end}}\tilde{r}_{\mathsf{ac}}\tilde{s}_\ell$ in $(\tilde{\alpha} - \tilde{d}_{\mathsf{ac}} + \tilde{w}_{\mathrm{end}}\tilde{r}_{\mathsf{ac}})\tilde{s}_\ell$.

Now we look at the terms underlined with a wave. The monomials $\tilde{w}_\sigma\tilde{r}_u\tilde{s}_i$ for $\sigma \neq x_i$ only appear once in $(\tilde{d}_{\delta(u,\sigma)} + \tilde{w}_\sigma\tilde{r}_u)\tilde{s}_i$ and thus these coefficients must also be 0. Also $\tilde{w}_\sigma\tilde{r}_u\tilde{s}_0$ only appears once, so we can remove this term as well.

Using these observations, we must have that

$$\begin{aligned}\tilde{\alpha}\tilde{s}_\ell \in \mathrm{span}\big((\tilde{\alpha} - \tilde{d}_{\mathsf{ac}})\tilde{s}_\ell \| \tilde{s}_0\tilde{d}_1 \| \{(\tilde{z}\tilde{s}_{i-1} + \tilde{w}_{x_i}\tilde{s}_i)\tilde{r}_u\}_{u\in[Q],i\in[\ell]} \| \{(-\tilde{d}_u + \tilde{z}\tilde{r}_u)\tilde{s}_i\}_{i\in[0,\ell],u\in[Q]} \| \\ \{(\tilde{d}_{\delta(u,x_i)} + \tilde{w}_{x_i}\tilde{r}_u)\tilde{s}_i\}_{i\in[\ell],u\in[Q]}\big) \ .\end{aligned} \tag{9}$$

Since $\tilde{\alpha}\tilde{s}_\ell$ only appears together with $\tilde{s}_\ell\tilde{d}_{\mathsf{ac}}$, it follows that $\tilde{s}_\ell\tilde{d}_{\mathsf{ac}}$ must be in the span of everything that comes after the first $\|$ on the RHS.

Finally, fix $i \in [\ell]$ and $u \in [Q]$. Note that the coefficients of the last three terms must be such that

$$\tilde{s}_i\tilde{d}_{\delta(u,x_i)} - \tilde{s}_{i-1}\tilde{d}_u = \tilde{s}_i(\tilde{d}_{\delta(u,x_i)} + \tilde{w}_{x_i}\tilde{r}_u) - (\tilde{z}\tilde{s}_{i-1} + \tilde{w}_{x_i}\tilde{s}_i)\tilde{r}_u + \tilde{s}_{i-1}(-\tilde{d}_u + \tilde{z}\tilde{r}_u) \ ,$$

since the monomials containing $\tilde{w}_{x_i}$ and $\tilde{z}$ only appear on the RHS of (9) and only together with $\tilde{s}_i\tilde{d}_u$.

From (9) and the observation above, we get

$$\tilde{s}_\ell\tilde{d}_{\mathsf{ac}} \in \mathrm{span}\big(\tilde{s}_0\tilde{d}_1 \| \{\tilde{s}_i\tilde{d}_{\delta(u,x_i)} - \tilde{s}_{i-1}\tilde{d}_u\}_{i\in[\ell],u\in[Q]}\big) \ ,$$

which proves claim 2.

Step 3. We prove the final claim by contradiction. Assume on the contrary that

$$\tilde{s}_\ell\tilde{d}_{\mathsf{ac}} \in \mathrm{span}\big(\tilde{s}_0\tilde{d}_1 \| \{\tilde{s}_i\tilde{d}_{\delta(u,x_i)} - \tilde{s}_{i-1}\tilde{d}_u\}_{i\in[\ell],u\in[Q]}\big) \ . \tag{10}$$

Since fin is the final state, correctness tells us that

$$\tilde{s}_\ell\tilde{d}_{\mathsf{fin}} \in \tilde{s}_0\tilde{d}_1 + \mathrm{span}\big(\{\tilde{s}_i\tilde{d}_{\delta(u,x_i)} - \tilde{s}_{i-1}\tilde{d}_u\}_{i\in[\ell],u\in[Q]}\big) \ .$$

Then we can rewrite (10) as

$$\tilde{s}_\ell\tilde{d}_{\mathsf{ac}} \in \mathrm{span}\big(\tilde{s}_\ell\tilde{d}_{\mathsf{fin}} \| \{\tilde{s}_i\tilde{d}_{\delta(u,x_i)} - \tilde{s}_{i-1}\tilde{d}_u\}_{i\in[\ell],u\in[Q]}\big) \ .$$

We now argue that all coefficients of elements on the RHS must be 0 and thus we get a contradiction to (10). We look at the monomials that appear in the above equation. Since $\tilde{s}_0\tilde{d}_u$ for $u \in [Q]$ only appears on the RHS, the coefficients of $\{\tilde{s}_1\tilde{d}_{\delta(u,x_1)} - \tilde{s}_0\tilde{d}_u\}_{u\in[Q]}$ must be 0. However, now the only term where $\tilde{s}_1\tilde{d}_u$ appears is $(\tilde{s}_2\tilde{d}_{\delta(u,x_2)} - \tilde{s}_1\tilde{d}_u)$, so these coefficients must also be 0 for all $u \in [Q]$. We can go on with this argument until we reach the terms $(\tilde{s}_\ell\tilde{d}_{\delta(u,x_\ell)} - \tilde{s}_{\ell-1}\tilde{d}_u)$. Obviously, the coefficients of these terms must also be 0 and thus we get

$$\tilde{s}_\ell\tilde{d}_{\mathsf{ac}} \in \mathrm{span}\big(\tilde{s}_\ell\tilde{d}_{\mathsf{fin}}\big) \ .$$

However, this yields a contradiction to the fact that $\mathsf{P}(x, (Q, \Sigma, \delta, F)) = 0$ since $\mathsf{ac} \neq \mathsf{fin}$, which concludes the proof of symbolic security.

Acknowledgements

Doreen Riepel was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2092 CASA - 390781972 and the European Union (ERC AdG REWORC - 101054911). Part of this work was done during an internship at NTT Research and a visit at UC Berkeley. We would also like to thank Kei Karasawa and his team for motivating discussions, as well as Sanjam Garg for hosting Doreen Riepel at UC Berkeley.

References

1. Agrawal, S., Chase, M.: Attribute-based encryption. https://github.com/sagrawal87/ABE (2017)
2. Agrawal, S., Chase, M.: FAME: Fast attribute-based message encryption. In: Thuraisingham, B.M., Evans, D., Malkin, T., Xu, D. (eds.) ACM CCS 2017. pp. 665–682. ACM Press (Oct / Nov 2017)
3. Agrawal, S., Chase, M.: Simplifying design and analysis of complex predicate encryption schemes. In: Coron, J.S., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part I. LNCS, vol. 10210, pp. 627–656. Springer, Heidelberg (Apr / May 2017)
4. Akinyele, J.A., Garman, C., Miers, I., Pagano, M.W., Rushanan, M., Green, M., Rubin, A.D.: Charm: a framework for rapidly prototyping cryptosystems. Journal of Cryptographic Engineering 3(2), 111–128 (Jun 2013)
5. Akinyele, J.A., Pagano, M.W., Green, M.D., Lehmann, C.U., Peterson, Z.N.J., Rubin, A.D.: Securing electronic medical records using attribute-based encryption on mobile devices. In: Jiang, X., Bhattacharya, A., Dasgupta, P., Enck, W. (eds.) SPSM'11, Proceedings of the 1st ACM Workshop Security and Privacy in Smartphones and Mobile Devices, Co-located with CCS 2011, October 17, 2011, Chicago, IL, USA. pp. 75–86. ACM (2011), http://doi.acm.org/10.1145/2046614.2046628

6. Ambrona, M., Barthe, G., Gay, R., Wee, H.: Attribute-based encryption in the generic group model: Automated proofs and new constructions. In: Thuraisingham, B.M., Evans, D., Malkin, T., Xu, D. (eds.) ACM CCS 2017. pp. 647–664. ACM Press (Oct / Nov 2017)
7. Attrapadung, N.: Dual system encryption via doubly selective security: Framework, fully secure functional encryption for regular languages, and more. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 557–577. Springer, Heidelberg (May 2014)
8. Attrapadung, N.: Dual system encryption framework in prime-order groups via computational pair encodings. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 591–623. Springer, Heidelberg (Dec 2016)
9. Baden, R., Bender, A., Spring, N., Bhattacharjee, B., Starin, D.: Persona: An online social network with user-defined privacy. In: Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication. pp. 135–146. SIGCOMM '09, ACM, New York, NY, USA (2009), http://doi.acm.org/10.1145/1592568.1592585
10. Bauer, B., Fuchsbauer, G., Plouviez, A.: The one-more discrete logarithm assumption in the generic group model. In: Tibouchi, M., Wang, H. (eds.) Advances in Cryptology – ASIACRYPT 2021. pp. 587–617. Springer International Publishing, Cham (2021)
11. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Denning, D.E., Pyle, R., Ganesan, R., Sandhu, R.S., Ashby, V. (eds.) ACM CCS 93. pp. 62–73. ACM Press (Nov 1993)
12. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: 2007 IEEE Symposium on Security and Privacy. pp. 321–334. IEEE Computer Society Press (May 2007)
13. Blazy, O., Kiltz, E., Pan, J.: (Hierarchical) identity-based encryption from affine message authentication. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 408–425. Springer, Heidelberg (Aug 2014)
14. Boneh, D., Boyen, X., Goh, E.J.: Hierarchical identity based encryption with constant size ciphertext. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 440–456. Springer, Heidelberg (May 2005)
15. Chase, M., Maller, M., Meiklejohn, S.: Déjà Q all over again: Tighter and broader reductions of q-type assumptions. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 655–681. Springer, Heidelberg (Dec 2016)
16. Chen, J., Gay, R., Wee, H.: Improved dual system ABE in prime-order groups via predicate encodings. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part II. LNCS, vol. 9057, pp. 595–624. Springer, Heidelberg (Apr 2015)
17. Chen, J., Gong, J., Weng, J.: Tightly secure IBE under constant-size master public key. In: Fehr, S. (ed.) PKC 2017, Part I. LNCS, vol. 10174, pp. 207–231. Springer, Heidelberg (Mar 2017)
18. Chen, J., Wee, H.: Fully, (almost) tightly secure IBE and dual system groups. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 435–460. Springer, Heidelberg (Aug 2013)
19. Cheon, J.H.: Security analysis of the strong Diffie-Hellman problem. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 1–11. Springer, Heidelberg (May / Jun 2006)
20. Chhatrapati, A., Hohenberger, S., Trombo, J., Vusirikala, S.: A performance evaluation of pairing-based broadcast encryption systems. In: Applied Cryptography and Network Security. Springer International Publishing (2022)
21. Diemert, D., Gellert, K., Jager, T., Lyu, L.: More efficient digital signatures with tight multi-user security. In: Garay, J. (ed.) PKC 2021, Part II. LNCS, vol. 12711, pp. 1–31. Springer, Heidelberg (May 2021)
22. Diemert, D., Jager, T.: On the tight security of TLS 1.3: Theoretically sound cryptographic parameters for real-world deployments. Journal of Cryptology 34(3), 30 (Jul 2021)
23. Faz-Hernández, A., Scott, S., Sullivan, N., Wahby, R.S., Wood, C.A.: Hashing to elliptic curves. https://datatracker.ietf.org/doc/draft-irtf-cfrg-hash-to-curve/ (2022)
24. Gentry, C.: Practical identity-based encryption without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 445–464. Springer, Heidelberg (May / Jun 2006)
25. Gjøsteen, K., Jager, T.: Practical and tightly-secure digital signatures and authenticated key exchange. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part II. LNCS, vol. 10992, pp. 95–125. Springer, Heidelberg (Aug 2018)
26. Gong, J., Dong, X., Chen, J., Cao, Z.: Efficient IBE with tight reduction to standard assumption in the multi-challenge setting. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 624–654. Springer, Heidelberg (Dec 2016)
27. Gong, J., Waters, B., Wee, H.: ABE for DFA from k-Lin. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part II. LNCS, vol. 11693, pp. 732–764. Springer, Heidelberg (Aug 2019)
28. Gong, J., Wee, H.: Adaptively secure ABE for DFA from k-Lin and more. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 278–308. Springer, Heidelberg (May 2020)
29. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Juels, A., Wright, R.N., De Capitani di Vimercati, S. (eds.) ACM CCS 2006. pp. 89–98. ACM Press (Oct / Nov 2006), available as Cryptology ePrint Archive Report 2006/309

30. Groth, J.: On the size of pairing-based non-interactive arguments. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 305–326. Springer, Heidelberg (May 2016)
31. Groth, J., Shoup, V.: On the security of ECDSA with additive key derivation and presignatures. Cryptology ePrint Archive, Report 2021/1330 (2021), https://eprint.iacr.org/2021/1330
32. Hoang, V.T., Tessaro, S., Thiruvengadam, A.: The multi-user security of GCM, revisited: Tight bounds for nonce randomization. In: Lie, D., Mannan, M., Backes, M., Wang, X. (eds.) ACM CCS 2018. pp. 1429–1440. ACM Press (Oct 2018)
33. Hofheinz, D., Koch, J., Striecks, C.: Identity-based encryption with (almost) tight security in the multi-instance, multi-ciphertext setting. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 799–822. Springer, Heidelberg (Mar / Apr 2015)
34. Ion, M., Zhang, J., Schooler, E.M.: Toward content-centric privacy in ICN: attribute-based encryption and routing. In: Ohlman, B., Polyzos, G.C., Zhang, L. (eds.) ICN'13, Proceedings of the 3rd, 2013 ACM SIGCOMM Workshop on Information-Centric Networking, August 12, 2013, Hong Kong, China. pp. 39–40. ACM (2013), http://doi.acm.org/10.1145/2491224.2491237
35. Kim, T., Barbulescu, R.: Extended tower number field sieve: A new complexity for the medium prime case. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016, Part I. LNCS, vol. 9814, pp. 543–571. Springer, Heidelberg (Aug 2016)
36. Kowalczyk, L., Wee, H.: Compact adaptively secure ABE for NC^1 from k-Lin. In: Ishai, Y., Rijmen, V. (eds.) EUROCRYPT 2019, Part I. LNCS, vol. 11476, pp. 3–33. Springer, Heidelberg (May 2019)
37. Langrehr, R., Pan, J.: Hierarchical identity-based encryption with tight multi-challenge security. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020, Part I. LNCS, vol. 12110, pp. 153–183. Springer, Heidelberg (May 2020)
38. Lewko, A.B., Waters, B.: Unbounded HIBE and attribute-based encryption. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 547–567. Springer, Heidelberg (May 2011)
39. Lewko, A.B., Waters, B.: New proof methods for attribute-based encryption: Achieving full security through selective techniques. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 180–198. Springer, Heidelberg (Aug 2012)
40. Li, J., Au, M.H., Susilo, W., Xie, D., Ren, K.: Attribute-based signature and its applications. In: Feng, D., Basin, D.A., Liu, P. (eds.) ASIACCS 10. pp. 60–69. ACM Press (Apr 2010)
41. Lin, H., Luo, J.: Compact adaptively secure ABE from k-Lin: Beyond NC^1 and towards NL. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 247–277. Springer, Heidelberg (May 2020)
42. Maurer, U.M.: Abstract models of computation in cryptography (invited paper). In: Smart, N.P. (ed.) 10th IMA International Conference on Cryptography and Coding. LNCS, vol. 3796, pp. 1–12. Springer, Heidelberg (Dec 2005)
43. de la Piedra, A., Venema, M., Alpár, G.: ABE squared: Accurately benchmarking efficiency of attribute-based encryption. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2022(2), 192–239 (2022), https://doi.org/10.46586/tches.v2022.i2.192-239
44. Riepel, D., Wee, H.: https://github.com/DoreenRiepel/FABEO (2022)
45. Sahai, A., Waters, B.R.: Fuzzy identity-based encryption. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 457–473. Springer, Heidelberg (May 2005)
46. Sakemi, Y., Kobayashi, T., Saito, T., Wahby, R.: Pairing-friendly curves. Internet-Draft https://datatracker.ietf.org/doc/draft-irtf-cfrg-pairing-friendly-curves/ (2021)
47. Shoup, V.: Lower bounds for discrete logarithms and related problems. In: Fumy, W. (ed.) EUROCRYPT'97. LNCS, vol. 1233, pp. 256–266. Springer, Heidelberg (May 1997)
48. Sullivan, N.: Geo Key Manager: How it works. https://blog.cloudflare.com/geo-key-manager-how-it-works/ (2017)
49. Tomida, J., Kawahara, Y., Nishimaki, R.: Fast, compact, and expressive attribute-based encryption. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020, Part I. LNCS, vol. 12110, pp. 3–33. Springer, Heidelberg (May 2020)
50. Waters, B.: Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 619–636. Springer, Heidelberg (Aug 2009)
51. Waters, B.: Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In: Catalano, D., Fazio, N., Gennaro, R., Nicolosi, A. (eds.) PKC 2011. LNCS, vol. 6571, pp. 53–70. Springer, Heidelberg (Mar 2011)
52. Waters, B.: Functional encryption for regular languages. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 218–235. Springer, Heidelberg (Aug 2012)
53. Wee, H.: Dual system encryption via predicate encodings. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 616–637. Springer, Heidelberg (Feb 2014)

A Attribute-based Encryption in the PKE Setting

Many-Ciphertext CPA Security. We define security by a game between a challenger and an adversary $\mathcal{B}$. The challenger picks a random challenge bit b and provides the following oracles to $\mathcal{B}$.

- Setup oracle $\mathcal{O}_{\mathsf{mpk}}$: This oracle can only be queried once and it must be the first query. The challenger runs Setup to obtain $(\mathsf{msk}, \mathsf{mpk})$ and outputs mpk to $\mathcal{B}$.
- Ciphertext (or challenge) oracle $\mathcal{O}_{\mathsf{ct}}$: On the i-th query, $\mathcal{B}$ provides $x_i \in \mathcal{X}$ as well as two messages $(m_i^{(0)}, m_i^{(1)})$ of equal length. The challenger runs $\mathsf{ct}_i \leftarrow \mathsf{Enc}(\mathsf{mpk}, x_i, m_i^{(b)})$ and outputs ct_i.
- Secret key oracle $\mathcal{O}_{\mathsf{sk}}$: On the j-th query, $\mathcal{B}$ provides $y_j \in \mathcal{Y}$. The challenger runs $\mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, y_i)$ and outputs sk_j.

Finally, $\mathcal{B}$ outputs a bit b'. We say that $\mathcal{B}$ wins the game if $b = b'$ and for all queries x_i and y_j, either $\mathsf{P}(x_i, y_j) = 0$ or $m_i^{(0)} = m_i^{(1)}$. Note that this ensures that $\mathcal{B}$ cannot win trivially.

Definition 4. *An* ABE *scheme is adaptively many-ciphertext secure in the PKE setting if for all efficient* $\mathcal{B}$

$$\mathsf{Adv}_{\mathsf{ABE},\mathcal{B}}(\lambda) := \left|\Pr[b = b'] - \tfrac{1}{2}\right|$$

is negligible in λ.

It is commonly known that the security notions in the KEM and PKE setting are equivalent. For completeness, we briefly sketch the proof that many-challenge KEM security as used in the main body of this paper tightly implies the many-ciphertext CPA security as defined above. Let $\mathcal{B}$ be an adversary against many-ciphertext CPA security in the PKE setting. We construct a reduction $\mathcal{A}$ against many-challenge KEM security. $\mathcal{A}$ draws a random challenge bit $b \xleftarrow{\$} \{0,1\}$ and then runs adversary $\mathcal{B}$. It answers $\mathcal{B}$'s queries as follows.

- When $\mathcal{B}$ queries $\mathcal{O}_{\mathsf{mpk}}$, $\mathcal{A}$ queries its own $\mathcal{O}_{\mathsf{mpk}}$ oracle to receive the master public key mpk which it forwards to $\mathcal{B}$.
- When $\mathcal{B}$ issues a ciphertext query $(x_i, m_i^{(0)}, m_i^{(1)})$, where $m_i^{(0)} \neq m_i^{(1)}$, $\mathcal{A}$ queries its own challenge oracle on x_i to receive an encapsulation c_i and challenge key $d_i^{(\beta)}$, where β is the challenge bit of the KEM security experiment. It outputs $\mathsf{ct}_i = (c_i, d_i^{(\beta)} \oplus m_i^{(b)})$ as ciphertext.
- When $\mathcal{B}$ issues a ciphertext query of the form (x_i, m_i, m_i), $\mathcal{A}$ runs the encapsulation algorithm on its own using the master public key to compute a pair (c_i, d_i) and outputs $\mathsf{ct}_i = (c_i, d_i \oplus m_i)$.
- When $\mathcal{B}$ issues a secret key query y_j, $\mathcal{A}$ forwards this query to its own secret key oracle and returns the result back to $\mathcal{B}$.

Note that queries of the third type are the only possibility for $\mathcal{B}$ to issue queries such that $\mathsf{P}(x_i, y_j) = 1$ for some secret key query y_j. Since $\mathcal{A}$ does not query any of its own oracles at this point, it behaves correctly.

Finally $\mathcal{B}$ outputs a bit b' and if $b' = b$, $\mathcal{A}$ outputs $\beta' = 0$ (real key). Otherwise, it outputs $\beta' = 1$ (random key).

We analyze the success probability by distinguishing two cases:

- Case $\beta = 0$: In this case $\mathcal{A}$ always receives the real KEM key and it perfectly simulates the many-ciphertext CPA game for $\mathcal{B}$. Thus, $\Pr[\beta = \beta' \mid \beta = 0] = \mathsf{Adv}_{\mathsf{ABE},\mathcal{B}}(\lambda) + 1/2$.
- Case $\beta = 1$: In this case $\mathcal{A}$ always receives a random KEM key. Thus, the challenge ciphertexts do not leak any information about β. Also queries of the second type are independent of β. We get $\Pr[\beta = \beta' \mid \beta = 1] = 1/2$.

Collecting the probabilities yields $\mathsf{Adv}_{\mathsf{ABE},\mathcal{B}}(\lambda) \leq 2 \cdot \mathsf{Adv}_{\mathsf{ABE},\mathcal{A}}(\lambda)$.

B Selective Security from q-type Assumptions

We can prove selective security of our CP-ABE and KP-ABE scheme (in the standard one-ct, many-sk setting) under a q-type assumption. In the selective security setting, the adversary must first commit on a challenge x^*. We recall the formal definition below.

Selective CPA Security. We define security by a game between a challenger and an adversary $\mathcal{A}$. At the beginning of the experiment, $\mathcal{A}$ chooses a challenge x^* and sends it to the challenger. Then the challenger picks a random challenge bit β and provides the following oracles to $\mathcal{A}$.

- Setup oracle $\mathcal{O}_{\mathsf{mpk}}$: This oracle can only be queried once and it must be the first query. The challenger runs Setup to obtain $(\mathsf{msk}, \mathsf{mpk})$ and outputs mpk to $\mathcal{A}$.
- Ciphertext (or challenge) oracle $\mathcal{O}_{\mathsf{ct}}$: The challenger runs $(\mathsf{ct}, d^{(0)}) \leftarrow \mathsf{Enc}(\mathsf{mpk}, x^*)$, chooses a random key $d^{(1)} \xleftarrow{\$} \mathcal{K}$ and outputs $(\mathsf{ct}, d^{(\beta)})$.
- Secret key oracle $\mathcal{O}_{\mathsf{sk}}$: On the j-th query, $\mathcal{A}$ provides $y_j \in \mathcal{Y}$. The challenger runs $\mathsf{sk}_j \leftarrow \mathsf{KeyGen}(\mathsf{msk}, y_j)$ and outputs sk_j.

We consider the single-challenge setting here, which means that $\mathcal{O}_{\mathsf{ct}}$ can only be queried once, whereas $\mathcal{O}_{\mathsf{sk}}$ can be queried adaptively and an arbitrary polynomial number of times. Finally, $\mathcal{A}$ outputs a bit β'. We say that $\mathcal{A}$ wins the game if $\beta = \beta'$ and $\mathsf{P}(x^*, y_j) = 0$ for all queries y_j.

Definition 5. *An* ABE *scheme is selectively secure if for all efficient* $\mathcal{A}$*,*

$$\mathsf{Adv}^{\mathsf{selective}}_{\mathsf{ABE},\mathcal{A}}(\lambda) \coloneqq \left|\Pr[\beta = \beta'] - \tfrac{1}{2}\right|$$

is negligible in λ*.*

Our q-type assumption. Our new assumption is parameterized by n. It allows us to prove security of both schemes based on the same assumption but for different parameters n (see Appendices B.1 and B.2). In Appendix B.3 we show how to relate our assumption to another assumptions used in the literature which in turn has been analyzed in the GGM. Our new assumption is defined below.

Definition 6 (n-q-type Assumption). *Let* $\mathcal{G} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$ *be a pairing group. Pick* $(x \| \delta \| \mathbf{y}) \xleftarrow{\$} \mathbb{Z}_p \times \mathbb{Z}_p \times \mathbb{Z}_p^n$*. Compute*

$$D^1 = \left(g_1,\ g_1^x,\ \left\{ g_1^{\delta \mathbf{y}[i]},\ g_1^{\frac{1}{x\mathbf{y}[i]}} \right\}_{i \in [n]},\ \left\{ g_1^{\frac{\delta \mathbf{y}[i]}{x\mathbf{y}[j]}} \right\}_{i,j \in [n], i \neq j} \right),$$

$$D^2 = \left(g_2^{\frac{1}{\delta}},\ g_2^{\frac{x}{\delta}},\ \left\{ g_2^{\mathbf{y}[i]} \right\}_{i \in [n]} \right)$$

and $T_0 = e(g_1, g_2)$*,* $T_1 \xleftarrow{\$} \mathbb{G}_T$*. The* n*-q-type assumption says that*

$$\mathsf{Adv}^{\mathsf{q\text{-}type}}_{\mathcal{D},n}(1^\lambda) \coloneqq \left|\Pr[\mathcal{D}(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, D^1, D^2, T_0) \Rightarrow 1] - \Pr[\mathcal{D}(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, D^1, D^2, T_1) \Rightarrow 1]\right|$$

is negligible in λ*.*

B.1 CP-ABE

We prove security of our CP-ABE scheme (cf. Figure 1) for $\tau = 1$ in the random oracle model under the above q-type assumption for $n = n_2$, where n_2 is the number of columns of $\mathbf{M}$ used in the challenge query.

Theorem 2. *Let* λ *be the security parameter. For any adversary* $\mathcal{A}$ *against selective security of* $\mathsf{CP\text{-}ABE}$ *with* $\tau = 1$ *that issues at most* Q_{sk} *queries to* $\mathcal{O}_{\mathsf{sk}}$*, one query to* $\mathcal{O}_{\mathsf{ct}}$ *and at most* Q_{H} *queries to the random oracle* H*, there exists an adversary* $\mathcal{D}$ *such that*

$$\mathsf{Adv}^{\mathsf{selective}}_{\mathsf{CP\text{-}ABE},\mathcal{A}}(1^\lambda) \leq \mathsf{Adv}^{\mathsf{q\text{-}type}}_{\mathcal{D},n_2}(1^\lambda) + \frac{1}{p}\ ,$$

where n_2 *is the number of columns of* $\mathbf{M}$ *used in the challenge query.*

Proof. We prove the theorem by reduction. Let $\mathcal{A}$ be an adversary against selective security of the KP-ABE scheme in Figure 1. We construct an adversary $\mathcal{D}$ against our q-type assumption that simulates the security game for $\mathcal{A}$.

$\mathcal{D}$ inputs $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, D^1, D^2, T_b)$ and runs adversary $\mathcal{A}$. First $\mathcal{A}$ commits on a challenge $(\mathbf{M}, \pi)$. Now $\mathcal{D}$ must simulate the first query to $\mathcal{O}_{\mathsf{mpk}}$, the challenge query to $\mathcal{O}_{\mathsf{ct}}$ and adaptive queries to $\mathcal{O}_{\mathsf{sk}}$ and H.

Simulating $\mathcal{O}_{\mathsf{mpk}}$. We implicitly set $\alpha = \frac{\delta}{x}$. Output public parameters

$$\left(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2^{\frac{1}{\delta}}, e(g_1, g_2^{\frac{1}{\delta}})^{\alpha}\right) ,$$

where the last term can be computed by $e(g_1^{\frac{1}{x\mathbf{y}[1]}}, g_2^{\mathbf{y}[1]})$.

Simulating H. We simulate the random oracle by lazy sampling and using a list $\mathcal{H}$ to store all (distinct) queries and outputs. In particular, the list contains entries of the form $(u, \mathsf{H}(u))$, where u is the string (attribute) queried to H and $\mathsf{H}(u)$ is a group element. On rach query, we first check if u is contained in the list and if this is the case, answer consistent to the previous query. Otherwise, we will simulate the output as follows. First pick a new exponent $\epsilon_u \xleftarrow{\$} \mathbb{Z}_p$ which will be used for randomization. Then we distinguish three cases and compute

$$\mathsf{H}(u) \coloneqq \begin{cases} g_1^{-\frac{1}{x\mathbf{y}[1]}} & \text{if } u = |\mathcal{U}| + 1 \\ g_1^{\sum_{j\in[n_2]} \frac{\mathbf{M}_{i,j}}{x\mathbf{y}[j]}} \cdot g_1^{\epsilon_u} & \text{if } \exists i \text{ s.t. } \pi(i) = u \\ g_1^{\epsilon_u} & \text{otherwise} \end{cases}$$

Then add $(u, \mathsf{H}(u))$ to the list and output $\mathsf{H}(u)$. Note that we can compute all the terms using $g_1, g_1^{\frac{1}{x\mathbf{y}[j]}}$ provided by the assumption.

Simulating $\mathcal{O}_{\mathsf{ct}}$. To compute the challenge for $(\mathbf{M}, \pi)$ provided at the beginning, we (implicitly) set $s_1 = x + \varepsilon_1$, $s' = x + \varepsilon'$ and $\mathbf{v}[j] = \frac{x\mathbf{y}[1]}{\mathbf{y}[j]} + \varepsilon_j$ for $j \in [2, n_2]$.

$$\mathsf{ct}_1 \coloneqq g_2^{\frac{x}{\delta}} \cdot g_2^{\frac{\varepsilon_1}{\delta}}$$

$$\mathsf{ct}_2 \coloneqq g_2^{\frac{x}{\delta}} \cdot g_2^{\frac{\varepsilon'}{\delta}}$$

$$\mathsf{ct}_{3,i} \coloneqq g_1^{-\frac{\mathbf{M}_i(s_1 \| \mathbf{v})^\top}{x\mathbf{y}[1]}} \cdot \left(g_1^{\sum_{j\in[n_2]} \frac{\mathbf{M}_{i,j}}{x\mathbf{y}[j]}} \cdot g_1^{\epsilon_{\pi(i)}}\right)^{s'}$$

$$d \coloneqq T_b \cdot e(g_1^{\frac{1}{x\mathbf{y}[1]}}, g_2^{\mathbf{y}[1]})^{\varepsilon_1}$$

It is easy to see that we can simulate ct_1 and ct_2 using the terms provided by the assumption. For $\mathsf{ct}_{3,i}$ we take a closer look at the exponent. Applying the definition of s_1, s' and $\mathbf{v}$, we get

$$-\frac{\mathbf{M}_i\left(x + \varepsilon_1 \left\| \frac{x\mathbf{y}[1]}{\mathbf{y}[2]} + \varepsilon_2 \right\| \cdots \left\| \frac{x\mathbf{y}[1]}{\mathbf{y}[n_2]} + \varepsilon_{n_2}\right)^\top}{x\mathbf{y}[1]} + \sum_{j\in[n_2]} \frac{\mathbf{M}_{i,j}(x + \varepsilon')}{x\mathbf{y}[j]} + \epsilon_{\pi(i)}(x + \varepsilon') \tag{11}$$

$$= -\sum_{j\in[n_2]} \frac{\mathbf{M}_{i,j}}{\mathbf{y}[j]} - \frac{\mathbf{M}_i(\varepsilon_1 \| \cdots \| \varepsilon_{n_2})^\top}{x\mathbf{y}[1]} + \sum_{j\in[n_2]} \frac{\mathbf{M}_{i,j}}{\mathbf{y}[j]} + \sum_{j\in[n_2]} \frac{\mathbf{M}_{i,j}\varepsilon'}{x\mathbf{y}[j]} + \epsilon_{\pi(i)}x + \epsilon_{\pi(i)}\varepsilon' \tag{12}$$

Observe that the first and the third term sum up to 0 and that the other terms can be simulated using the g_1 terms of the assumption.

Also note that if $b = 0$, then $d = e(g_1, g_2) \cdot e(g_1, g_2)^{\frac{\varepsilon_1}{x}} = e(g_1, g_2^{\frac{1}{\delta}})^{\frac{\delta}{x}(x+\varepsilon_1)} = e(g_1, g_2^{\frac{1}{\delta}})^{\alpha s_1}$ is the real KEM key and if $b = 1$, then d is uniformly distributed in $\mathbb{G}_T$.

Simulating $\mathcal{O}_{\mathsf{sk}}$. On a query $\mathcal{S}$ to $\mathcal{O}_{\mathsf{sk}}$, first compute $\mathbf{w} \in \mathbb{Z}_p^{n_2}$ such that $\langle \mathbf{w}, \mathbf{M}_i^\top \rangle = 0$ for all $i \in [n_1]$ such that $\pi(i) \in \mathcal{S}$. Note that such a vector $\mathbf{w}$ is efficiently computable and we can assume that w.l.o.g. $\mathbf{w}[1] = 1$. We implicitly set $r = \delta \sum_{j\in[n_2]} \mathbf{w}[j]\mathbf{y}[j] + \gamma$, where $\gamma \xleftarrow{\$} \mathbb{Z}_p$. Now we can compute sk as follows:

$$\mathsf{sk}_1 \coloneqq g_1^{\frac{\delta}{x}} \cdot g_1^{-\frac{\delta\sum_{j\in[n_2]} \mathbf{w}[j]\mathbf{y}[j] + \gamma}{x\mathbf{y}[1]}} = g_1^{-\frac{\sum_{j\in[2,n_2]} \delta\mathbf{w}[j]\mathbf{y}[j]}{x\mathbf{y}[1]}} \cdot g_1^{-\frac{\gamma}{x\mathbf{y}[1]}}$$

$$\mathsf{sk}_{2,u} \coloneqq \mathsf{H}(u)^{\delta \sum_{j\in[n_2]} \mathbf{w}[j]\mathbf{y}[j]+\gamma}$$

$$\mathsf{sk}_3 \coloneqq (g_2^{\frac{1}{\delta}})^{\delta \sum_{j\in[n_2]} \mathbf{w}[j]\mathbf{y}[j]+\gamma} = g_2^{\sum_{j\in[n_2]} \mathbf{w}[j]\mathbf{y}[j]} \cdot g_2^{\frac{\gamma}{\delta}}$$

Observe that sk_1 and sk_3 can be simulated using the terms provided by the assumption. For $\mathsf{sk}_{2,u}$ we take a closer look at the exponent. Since the value depends on $\mathsf{H}(u)$, we distinguish two cases, starting with the case that there exists an index i such that $\pi(i) = u$. Applying the definition of $\mathsf{H}(u)$, we get

$$\begin{aligned}
&\left(\sum_{j\in[n_2]} \frac{\mathbf{M}_{i,j}}{x\mathbf{y}[j]} + \epsilon_u\right)\left(\delta \sum_{j\in[n_2]} \mathbf{w}[j]\mathbf{y}[j] + \gamma\right) \\
= &\underbrace{\sum_{\substack{j\in[n_2]\\k\in[n_2]}} \frac{\delta \mathbf{M}_{ij}\mathbf{w}[k]\mathbf{y}[k]}{x\mathbf{y}[j]}}_{\frac{\delta\mathbf{M}_i\mathbf{w}^\top}{x} + \sum_{\substack{j\in[n_2]\\k\in[n_2]\\j\neq k}} \frac{\delta\mathbf{M}_{ij}\mathbf{w}[k]\mathbf{y}[k]}{x\mathbf{y}[j]}} + \sum_{j\in[n_2]} \frac{\mathbf{M}_{i,j}\gamma}{x\mathbf{y}[j]} + \sum_{j\in[n_2]} \delta\epsilon_u \mathbf{w}[j]\mathbf{y}[j] + \epsilon_u\gamma
\end{aligned}$$

The first term underneath the curly bracket is 0 since $\mathbf{w}$ is chosen such that $\mathbf{M}_i\mathbf{w}^\top = 0$ for all $\pi(i) \in \mathcal{S}$. The other terms can be simulated using the g_1 terms of the assumption.

It remains to show that we can simulate $\mathsf{sk}_{2,u}$ in case there does not exist an index for all i such that $\pi(i) = u$. In this case $\mathsf{H}(u)$ is simply $g_1^{\epsilon_u}$, so we get

$$g_1^{\epsilon_u\left(\delta \sum_{j\in[n_2]} \mathbf{w}[j]\mathbf{y}[j]+\gamma\right)} = g_1^{\epsilon_u\delta \sum_{j\in[n_2]} \mathbf{w}[j]\mathbf{y}[j]} \cdot g_1^{\epsilon_u\gamma} \ ,$$

which can be computed using the terms provided by the assumption.

Finalize. At the end, the adversary $\mathcal{A}$ will output a bit β' which will also be $\mathcal{D}$'s output. Note that if $\mathcal{A}$ wins, then $\mathcal{D}$ will always output the correct bit except when T_1 coincides with T_0 which happens with probability $1/p$. This concludes the proof of Theorem 2.

B.2 KP-ABE

In a similar way, we prove security of our KP-ABE scheme (cf. Figure 1) for $\tau = 1$ in the random oracle model for parameter $n = Q_{\mathsf{H}}$, where Q_{H} is the number of queries to the random oracle H.

Theorem 3. *Let λ be the security parameter. For any adversary $\mathcal{A}$ against selective security of* KP-ABE *with $\tau = 1$ that issues at most Q_{sk} queries to $\mathcal{O}_{\mathsf{sk}}$, one query to $\mathcal{O}_{\mathsf{ct}}$ and at most Q_{H} queries to the random oracle H, there exists an adversary $\mathcal{D}$ such that*

$$\mathsf{Adv}^{\mathsf{selective}}_{\mathsf{KP\text{-}ABE},\mathcal{A}}(1^\lambda) \leq \mathsf{Adv}^{\mathsf{q\text{-}type}}_{\mathcal{D},Q_{\mathsf{H}}}(1^\lambda) + \frac{1}{p} \ .$$

Proof. The proof is similar to that of Theorem 2. Let $\mathcal{A}$ be an adversary against selective security of the KP-ABE scheme in Figure 1. We construct an adversary $\mathcal{D}$ against our q-type assumption that simulates the security game for $\mathcal{A}$.

$\mathcal{D}$ inputs $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, D^1, D^2, T_b)$ and runs adversary $\mathcal{A}$. First $\mathcal{A}$ commits on a challenge attribute set $\mathcal{S}$. Now $\mathcal{D}$ must simulate the first query to $\mathcal{O}_{\mathsf{mpk}}$, the challenge query to $\mathcal{O}_{\mathsf{ct}}$ and adaptive queries to $\mathcal{O}_{\mathsf{sk}}$ and H.

Simulating $\mathcal{O}_{\mathsf{mpk}}$. This is done as in the proof of Theorem 2. We implicitly set $\alpha = \frac{\delta}{x}$ and output public parameters

$$\left(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2^{\frac{1}{\delta}}, e(g_1, g_2^{\frac{1}{\delta}})^\alpha\right) \ .$$

Simulating H. The random oracle is simulated by lazy sampling as described above. Note that the assumption is parameterized by the maximum number of random oracle queries and we index each element of the vector $\mathbf{y}$ by the values u queried to H, i.e., $\mathbf{y}[u]$. Thus, on each query which is distinct to previous ones, we will first pick a new exponent $\epsilon_u \xleftarrow{\$} \mathbb{Z}_p$ and output

$$\mathsf{H}(u) \coloneqq \begin{cases} g_1^{\frac{1}{x\mathbf{y}[u]}} \cdot g_1^{\epsilon_u} & \text{if } \exists u \notin \mathcal{S} \\ g_1^{\epsilon_u} & \text{otherwise} \end{cases}$$

Simulating ct. To compute the challenge for $\mathcal{S}$ provided at the beginning, we (implicitly) set $s = x + \varepsilon$ for $\varepsilon \xleftarrow{\$} \mathbb{Z}_p$. Then compute and output

$$\mathsf{ct}_{1,u} \coloneqq \mathsf{H}(u)^{x+\varepsilon} = g_1^{x\epsilon_u} \cdot g_1^{\epsilon_u \varepsilon}$$

$$\mathsf{ct}_2 \coloneqq g_2^{\frac{x}{\delta}} \cdot g_2^{\frac{\varepsilon}{\delta}}$$

$$d \coloneqq T_b \cdot e(g_1^{\frac{1}{x\mathbf{y}[1]}}, g_2^{\mathbf{y}[1]})^{\varepsilon}$$

Note that if $b = 0$, then $d = e(g_1, g_2) \cdot e(g_1, g_2)^{\frac{\varepsilon}{x}} = e(g_1, g_2^{\frac{1}{\delta}})^{\frac{\delta}{x}(x+\varepsilon)} = e(g_1, g_2^{\frac{1}{\delta}})^{\alpha s}$ is the real KEM key and if $b = 1$, then d is uniformly distributed in $\mathbb{G}_T$.

Simulating sk. On a query $(\mathbf{M}, \pi)$ to $\mathcal{O}_{\mathsf{sk}}$, first compute $\mathbf{w} \in \mathbb{Z}_p^{n_2}$ such that $\langle \mathbf{w}, \mathbf{M}_i^\top \rangle = 0$ for all $i \in [n_1]$ such that $\pi(i) \in \mathcal{S}$. Note that such a vector $\mathbf{w}$ is efficiently computable and we can assume that w.l.o.g. $\mathbf{w}[1] = 1$. We implicitly set $r' = -\delta \sum_{i \in [n_1]} \mathbf{M}_i \mathbf{w}^\top \mathbf{y}[\pi(i)] + \gamma_1$ and $\mathbf{v}[j] = \frac{\delta \mathbf{w}[j]}{x} + \gamma_j$ for $j \in [2, n_2]$, where $\gamma_1, \ldots, \gamma_{n_2} \xleftarrow{\$} \mathbb{Z}_p$. Now we can compute sk as follows:

$$\mathsf{sk}_1 \coloneqq (g_2^{\frac{1}{\delta}})^{-\delta \sum_{i \in [n_1]} \mathbf{M}_i \mathbf{w}^\top \mathbf{y}[\pi(i)] + \gamma_1} = g_2^{\sum_{i \in [n_1]} \mathbf{M}_i \mathbf{w}^\top \mathbf{y}[\pi(i)]} \cdot g_2^{\frac{\gamma_1}{\delta}}$$

$$\mathsf{sk}_{2,i} \coloneqq g_1^{\mathbf{M}_i (\frac{\delta}{x} \| \mathbf{v})^\top} \cdot \mathsf{H}(\pi(i))^{-\delta \sum_{i \in [n_1]} \mathbf{M}_i \mathbf{w}^\top \mathbf{y}[\pi(i)] + \gamma_1}$$

It is easy to see that we can simulate sk_1 using $g_2^{\frac{1}{\delta}}$ and $g_2^{\mathbf{y}}[u]$ terms of the assumption. For $\mathsf{sk}_{2,i}$ we take a closer look at the exponent. Since the value depends on $\mathsf{H}(\pi(i))$, we distinguish two cases, starting with the case that $\pi(i) \notin \mathcal{S}$. Applying the definition of $\mathbf{v}$ and $\mathsf{H}(u)$, we get

$$\mathbf{M}_i \left(\frac{\delta}{x} \left\| \frac{\delta \mathbf{w}[2]}{x} + \gamma_2 \right\| \cdots \left\| \frac{\delta \mathbf{w}[n_2]}{x} + \gamma_{n_2} \right)^\top - \left(\frac{1}{x\mathbf{y}[\pi(i)]} + \epsilon_{\pi(i)} \right) \left(\sum_{k \in [n_1]} \delta \mathbf{M}_k \mathbf{w}^\top \mathbf{y}[\pi(k)] + \gamma_1 \right)$$

$$= \frac{\delta \mathbf{M}_i \mathbf{w}^\top}{x} + \sum_{j \in [2, n_2]} \mathbf{M}_{i,j} \gamma_j - \underbrace{\sum_{k \in [n_1]} \frac{\delta \mathbf{M}_k \mathbf{w}^\top \mathbf{y}[\pi(k)]}{x\mathbf{y}[\pi(i)]}}_{\frac{\delta \mathbf{M}_i \mathbf{w}^\top}{x} + \sum_{\substack{k \in [n_1] \\ k \neq i}} \frac{\delta \mathbf{M}_k \mathbf{w}^\top \mathbf{y}[\pi(k)]}{x\mathbf{y}[\pi(i)]}} - \sum_{k \in [n_1]} \delta \mathbf{M}_k \mathbf{w}^\top \mathbf{y}[\pi(k)] \epsilon_{\pi(i)} - \frac{\gamma_1}{x\mathbf{y}[\pi(i)]} - \epsilon_{\pi(i)} \gamma_1$$

Observe that the first term cancels with the first term underneath the curly bracket and that the other terms can be simulated using the g_1 terms of the assumption.

It remains to show that we can simulate $\mathsf{sk}_{2,i}$ for all i such that $\pi(i) \in \mathcal{S}$. In this case $\mathsf{H}(\pi(i))$ is simply $g_1^{\epsilon_{\pi(i)}}$, so we get

$$\frac{\delta \mathbf{M}_i \mathbf{w}^\top}{x} + \sum_{j \in [2, n_2]} \mathbf{M}_{i,j} \gamma_j - \sum_{k \in [n_1]} \delta \mathbf{M}_k \mathbf{w}^\top \mathbf{y}[\pi(k)] \epsilon_{\pi(i)} - \epsilon_{\pi(i)} \gamma_1$$

where the first term is 0 since $\mathbf{w}$ is chosen such that $\mathbf{M}_i \mathbf{w}^\top = 0$ for all $\pi(i) \in \mathcal{S}$. The remaining terms can be computed using the terms provided by the assumption.

Finalize. At the end, the adversary $\mathcal{A}$ will output a bit β' which will also be $\mathcal{D}$'s output. Note that if $\mathcal{A}$ wins, then $\mathcal{D}$ will always output the correct bit except when T_1 coincides with T_0 which happens with probability $1/p$. This concludes the proof of Theorem 3.

B.3 Relation to Other q-type Assumptions

In order to justify our new q-type assumption, we show that it is implied by another known assumption. We recall the expanded Diffie-Hellman exponent assumption-2 (EDHE2) introduced by Attrapadung [7, Def. 4]. The assumption is defined for a group $\mathbb{G}_1$ of prime order p with generator $\tilde{g_1}$. It is parameterized by two integers n, m and says that given

$$
\begin{aligned}
& \tilde{g_1}, \tilde{g_1}^{a}, \tilde{g_1}^{b}, \tilde{g_1}^{\frac{a^{n-1}c}{z}}, \\
\forall i \in [n], j, j' \in [m], j \neq j' : \quad & \tilde{g_1}^{\frac{a^i}{d_j^2}}, \tilde{g_1}^{\frac{a^i b}{d_j}}, \tilde{g_1}^{d_j}, \tilde{g_1}^{\frac{a^i d_j}{d_{j'}^2}}, \tilde{g_1}^{\frac{a^i b d_j}{d_{j'}}}, \tilde{g_1}^{\frac{a^i}{d_j^6}}, \tilde{g_1}^{\frac{a^i d_j}{d_{j'}^6}}, \\
\forall i \in [0, n-1] : \quad & \tilde{g_1}^{a^i c}, \tilde{g_1}^{a^i bcd_j}, \\
\forall i \in [0, n], j \in [m] : \quad & \tilde{g_1}^{a^i bcd_j^5}, \\
\forall i \in [2n-1], j, j' \in [m], j \neq j' : \quad & \tilde{g_1}^{\frac{a^i bcd_j}{d_{j'}^2}}, \tilde{g_1}^{\frac{a^i bcd_j^5}{d_{j'}^6}}, \\
\forall i \in [2n-1], i \neq n, j \in [m] : \quad & \tilde{g_1}^{\frac{a^i bc}{d_j}}, \\
\forall i \in [2n-1], j, j' \in [m] : \quad & \tilde{g_1}^{\frac{a^i c}{d_j^2}}, \tilde{g_1}^{\frac{a^i b^2 cd_j}{d_{j'}}}, \tilde{g_1}^{\frac{a^i bcd_j}{d_{j'}^6}}, \tilde{g_1}^{\frac{a^i}{d_j^6}}, \tilde{g_1}^{\frac{a^i bcd_j^5}{d_{j'}^2}}, \tilde{g_1}^{\frac{a^i b^2 cd_j^5}{d_{j'}}},
\end{aligned}
$$

$\tilde{T_0} = \tilde{g_1}^{abz}$ is computationally indistinguishable from a random element $\tilde{T_1} \xleftarrow{\$} \mathbb{G}_1$.

We will look at the asymmetric version of the assumption where the adversary is also given $\tilde{g_2}$ raised to the same exponents and the challenge is provided in $\mathbb{G}_2$. The terms highlighted in gray will be needed for our reduction. Then we can show that the asymmetric $(2, n)$-EDHE2 assumption implies our n-q-type assumption. Given an adversary $\mathcal{A}$ against the n-q-type assumption, we construct a reduction $\mathcal{B}$ as follows. $\mathcal{B}$ inputs the terms of the asymmetric $(2, n)$-EDHE2 assumption as described above, it then (implicitly) sets

$$
\begin{aligned}
g_1 &= \tilde{g_1}^{ac} \\
g_2 &= \tilde{g_2}^{ab} \\
\delta &= \frac{b\gamma}{c} \quad \text{where } \gamma \xleftarrow{\$} \mathbb{Z}_p \\
x &= \frac{\omega}{a} \quad \text{where } \omega \xleftarrow{\$} \mathbb{Z}_p \\
\mathbf{y}[i] &= \frac{d_i}{ab}
\end{aligned}
$$

It computes the terms of the n-q-type assumption as

$$
\begin{aligned}
g_1^x &= (\tilde{g_1}^{ac})^{\frac{\omega}{a}} = (\tilde{g_1}^{c})^{\omega} \\
g_1^{\delta \mathbf{y}[i]} &= (\tilde{g_1}^{ac})^{\frac{b\gamma}{c} \cdot \frac{d_i}{ab}} = \left(\tilde{g_1}^{d_i}\right)^{\gamma} \\
g_1^{\frac{1}{x\mathbf{y}[i]}} &= (\tilde{g_1}^{ac})^{\frac{a}{\omega} \cdot \frac{ab}{d_i}} = \left(\tilde{g_1}^{\frac{a^3 bc}{d_i}}\right)^{\frac{1}{w}} \\
g_1^{\frac{\delta \mathbf{y}[i]}{x\mathbf{y}[j]}} &= (\tilde{g_1}^{ac})^{\frac{b\gamma}{c} \cdot \frac{a}{\omega} \cdot \frac{d_i}{d_j}} = \left(\tilde{g_1}^{\frac{a^2 bd_i}{d_j}}\right)^{\frac{\gamma}{\omega}} \\
g_2^{\frac{1}{\delta}} &= (\tilde{g_2}^{ab})^{\frac{c}{b\gamma}} = (\tilde{g_2}^{ac})^{\frac{1}{\gamma}} \\
g_2^{\frac{x}{\delta}} &= (\tilde{g_2}^{ab})^{\frac{\omega}{a} \cdot \frac{c}{b\gamma}} = (\tilde{g_2}^{c})^{\frac{\omega}{\gamma}} \\
g_2^{\mathbf{y}[i]} &= (\tilde{g_2}^{ab})^{\frac{d_i}{ab}} = \tilde{g_2}^{d_i} \\
T_b &= e(\tilde{g_1}^{\frac{ac}{z}}, \tilde{T_b})
\end{aligned}
$$

Note that for $\tilde{T_0} = g_2^{abz}$, we have $T_b = e(\tilde{g_1}^{\frac{ac}{z}}, \tilde{g_2}^{abz}) = e(\tilde{g_1}^{ac}, \tilde{g_2}^{ab}) = e(g_1, g_2)$ and for $\tilde{T_1} \xleftarrow{\$} \mathbb{G}_2$, T_b is a random group element in $\mathbb{G}_T$. Thus, if $\mathcal{A}$ outputs the correct bit, so does the reduction $\mathcal{B}$.

C ABGW Schemes

Below are the CP-ABE and KP-ABE scheme of ABGW [6] that we implemented. $\mathbf{M}$, π are defined as in Section 2.2.

The CP-ABE scheme is defined as follows.

- $\underline{\mathsf{Setup}(1^\lambda)}$. Run $\mathsf{GroupGen}(1^\lambda)$ to obtain $\mathcal{G} \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$. Pick $(\alpha \| \mathbf{b}) \xleftarrow{\$} \mathbb{Z}_p \times \mathbb{Z}_p^4$. Output $(\mathcal{G}, (g_1^{\mathbf{b}[j]})_{j\in[4]}, e(g_1,g_2)^\alpha)$ as the master public key mpk and $(\alpha, \mathbf{b})$ as the master secret key msk.
- $\underline{\mathsf{KeyGen}(\mathsf{msk}, \mathcal{S})}$. Pick $r \xleftarrow{\$} \mathbb{Z}_p$. Compute

$$\mathsf{sk}_1 \coloneqq g_2^{\alpha - \mathbf{b}[4]r}, \quad \mathsf{sk}_2 \coloneqq g_2^r$$

  as well as

$$\mathsf{sk}_{3,u} \coloneqq g_2^{\frac{r\mathbf{b}[3]}{\mathbf{b}[1]+u\mathbf{b}[2]}}$$

  for all $u \in \mathcal{S}$. Output the secret key $\mathsf{sk} \coloneqq (\mathsf{sk}_1, \mathsf{sk}_2, (\mathsf{sk}_{3,u})_{u\in\mathcal{S}})$.
- $\underline{\mathsf{Enc}(\mathsf{mpk}, (\mathbf{M}, \pi))}$. Pick $(s_1 \| \mathbf{s}' \| \mathbf{v}) \xleftarrow{\$} \mathbb{Z}_p \times \mathbb{Z}_p^{n_1} \times \mathbb{Z}_p^{n_2}$. Let $\mu_i \coloneqq \mathbf{M}_i(s_1 \| \mathbf{v})^\top$ for all $i \in [n_1]$. Then compute

$$\mathsf{ct}_{1,i} \coloneqq g_1^{\mu_i}, \quad \mathsf{ct}_{2,i} \coloneqq g_1^{-\mathbf{b}[3]\mathbf{s}'[i]} \cdot g_1^{\mathbf{b}[4]\mu_i}, \quad \mathsf{ct}_{3,i} \coloneqq \left(g_1^{\mathbf{b}[1]} \cdot g_1^{\mathbf{b}[2]\pi(i)}\right)^{\mathbf{s}'[i]}$$

  and $d \coloneqq e(g_1,g_2)^{\alpha s_1}$. Output the ciphertext $\mathsf{ct} \coloneqq (\mathsf{ct}_{1,i}, \mathsf{ct}_{2,i}, \mathsf{ct}_{3,i})_{i\in[n_1]}$ and key d.
- $\underline{\mathsf{Dec}(\mathsf{mpk}, (\mathbf{M}, \pi), \mathcal{S}, \mathsf{ct}, \mathsf{sk})}$. If $\mathcal{S}$ satisfies $(\mathbf{M}, \pi)$, there exist constants $\{\gamma_i\}_{i\in I}$ s.t. $\sum_{i\in I} \gamma_i \mathbf{M}_i = (1, 0, \ldots, 0)$. Reconstruct d by computing

$$e\left(\prod_{i\in I} \mathsf{ct}_{1,i}^{\gamma_i}, \mathsf{sk}_1\right) \cdot e\left(\prod_{i\in I} \mathsf{ct}_{2,i}^{\gamma_i}, \mathsf{sk}_2\right) \cdot \prod_{i\in I} e(\mathsf{ct}_{3,i}, \mathsf{sk}_{3,\pi(i)})^{\gamma_i}$$

  and output the result.

The KP-ABE scheme is defined as follows.

- $\underline{\mathsf{Setup}(1^\lambda)}$. Run $\mathsf{GroupGen}(1^\lambda)$ to obtain $\mathcal{G} \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$. Pick $(\alpha \| \mathbf{b}) \xleftarrow{\$} \mathbb{Z}_p \times \mathbb{Z}_p^2$. Output $(\mathcal{G}, g_1^{\mathbf{b}[1]}, g_1^{\mathbf{b}[2]}, e(g_1,g_2)^\alpha)$ as the master public key mpk and $(\alpha, \mathbf{b})$ as the master secret key msk.
- $\underline{\mathsf{KeyGen}(\mathsf{msk}, (\mathbf{M}, \pi))}$. Pick $\mathbf{r} \xleftarrow{\$} \mathbb{Z}_p^{n_2-1}$. Let $\mu_i \coloneqq \mathbf{M}_i(\alpha \| \mathbf{r})^\top$ for all $i \in [n_1]$. Then compute

$$\mathsf{sk}_{1,i} \coloneqq g_2^{\mu_i}, \quad \mathsf{sk}_{2,i} \coloneqq g_2^{\frac{\mu_i}{\mathbf{b}[1]+\pi(i)\mathbf{b}[2]}}$$

  and output the secret key $\mathsf{sk} \coloneqq (\mathsf{sk}_{1,i}, \mathsf{sk}_{2,i})_{i\in[n_1]}$.
- $\underline{\mathsf{Enc}(\mathsf{mpk}, \mathcal{S})}$. Pick $(s_1 \| \mathbf{s}') \xleftarrow{\$} \mathbb{Z}_p \times \mathbb{Z}_p^{|\mathcal{S}|}$. For all $u \in \mathcal{S}$, compute

$$\mathsf{ct}_{1,u} \coloneqq g_1^{s_1 - \mathbf{s}'[u]}, \quad \mathsf{ct}_{2,u} \coloneqq \left(g_1^{\mathbf{b}[1]} \cdot g_1^{\mathbf{b}[2]u}\right)^{\mathbf{s}'[u]}$$

  and $d \coloneqq e(g_1,g_2)^{\alpha s_1}$. Output the ciphertext $\mathsf{ct} \coloneqq (\mathsf{ct}_{1,u}, \mathsf{ct}_{2,u})_{u\in\mathcal{S}}$ and key d.
- $\underline{\mathsf{Dec}(\mathsf{mpk}, \mathcal{S}, (\mathbf{M}, \pi), \mathsf{ct}, \mathsf{sk})}$. If $\mathcal{S}$ satisfies $(\mathbf{M}, \pi)$, there exist constants $\{\gamma_i\}_{i\in I}$ s.t. $\sum_{i\in I} \gamma_i \mathbf{M}_i = (1, 0, \ldots, 0)$. Reconstruct d by computing

$$\frac{\prod_{i\in I} e(\mathsf{ct}_{1,\pi(i)}, \mathsf{sk}_{1,i})^{\gamma_i}}{\prod_{i\in I} e(\mathsf{ct}_{2,\pi(i)}, \mathsf{sk}_{2,i})^{\gamma_i}}$$

  and output the result.