IACR Communications in Cryptology
ISSN 3006-5496, Vol. 1, No. 3, 31 pages.

https://doi.org/10.62056/ahsdkmp-3y

Exponent-Inversion P-Signatures and Accountable Identity-Based Encryption from SXDH

Tsz Hon Yuen[1], Sherman S. M. Chow[2], Huangting Wu[2], Cong Zhang[3] and Siu-Ming Yiu[4]

[1] Faculty of Information Technology, Monash University, Melbourne, VIC, Australia
[2] Department of Information Engineering, Chinese University of Hong Kong, Shatin, Hong Kong
[3] The State Key Laboratory of Blockchain and Data Security, Zhejiang University, China
[4] Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong

Abstract. Salient in many cryptosystems, the exponent-inversion technique began without randomization in the random oracle model (SCIS '03, PKC '04), evolved into the Boneh-Boyen short signature scheme (JoC '08) and exerted a wide influence. Seen as a notable case, Gentry's (EuroCrypt '06) identity-based encryption (IBE) applies exponent inversion on a randomized base in its identity-based trapdoors. Making use of the non-static q-strong Diffie-Hellman assumption, Boneh-Boyen signatures are shown to be unforgeable against q-chosen-message attacks, while a variant q-type decisional assumption is used to establish the security of Gentry-IBE. Challenges remain in proving their security under weaker static assumptions.

Supported by the dual form/system framework (Crypto '09, AsiaCrypt '12), we propose dual form exponent-inversion Boneh-Boyen signatures and Gentry-IBE, with security proven under the symmetric external Diffie-Hellman (SXDH) assumption. Starting from our signature scheme, we extend it into P-signatures (TCC '08), resulting in the first anonymous credential scheme from the SXDH assumption, serving as a competitive alternative to the static-assumption construction of Abe *et al.* (JoC '16). Moreover, from our Gentry-IBE variant, we propose an accountable-authority IBE scheme also from SXDH, surpassing the fully secure Sahai-Seyalioglu scheme (PKC '11) in efficiency and the generic Kiayias-Tang transform (ESORICS '15) in security. Collectively, we present a suite of results under static assumptions.

Keywords: Dual form signature · Dual system encryption · Exponent inversion · P-Signatures · Anonymous credentials · Identity-based encryption · Black-box accountability · Static assumptions · Symmetric eXternal Diffie-Hellman

1 Introduction

Given a public key (g, g^{α}), a commonly used signature form for a message M is $\sigma = g^{\frac{1}{\alpha+M}}$, which appears in various schemes analyzed in the random oracle model, from the identity-based encryption (IBE) scheme of Sakai and Kasahara [SK03] to the short signature scheme of Zhang, Safavi-Naini, and Susilo [ZSS04] in the random oracle model. This paper focuses on achieving adaptive security in the standard model, specifically referencing the Boneh-Boyen short signature scheme [BB08] to highlight their randomization techniques for achieving such security. This "inversion in the exponent" structure underpins a wide

This is the full version of our preprint at Cryptology ePrint Archive [YCZY14]. Chow (corresponding) is supported by GRF 14210621. We are grateful to the reviewers for recognizing our contributions and preserving our anonymity in their thorough review while several subsequent works cited our work.

E-mail: john.tszhonyuen@monash.edu (Tsz Hon Yuen), smchow@ie.cuhk.edu.hk (Sherman S. M. Chow), wh017@ie.cuhk.edu.hk (Huangting Wu), congresearch@zju.edu.cn (Cong Zhang)

Received: 2024-07-09 Accepted: 2024-09-02

range of signature schemes, such as Boneh-Boyen-Shacham (BBS) signatures [BBS04], their extension BBS+ (enabling the signing of multiple messages with randomness incorporated) [ASMC13], structure-preserving signatures, and blind signatures [AFG+10], among others. It can also be used as a "second-tier" secret, such as the user signing keys of group signatures [BBS04, Gro07], decryption keys of dynamic threshold decryption [DP08], identity-based (user) secret keys (or trapdoors) of IBE [Gen06], identity-based broadcast encryption [Del07, CCH+12], hierarchical IBE with polynomially many levels [GH09], and more. Last but not least, this structure also appears in other cryptographic primitives like verifiable random function [DY05], accumulator [Ngu05, ACN13], *etc.* [CY11].

The security of the Boneh-Boyen signature is based on the q-strong Diffie-Hellman (SDH) assumption — for α randomly selected from $\mathbb{Z}_p$, we have:

$$\text{Given } g, g^{\alpha}, g^{\alpha^2}, \ldots, g^{\alpha^q} \in \mathbb{G}, \text{ it is hard to output } (c, g^{\frac{1}{\alpha+c}}), \text{ where } c \in \mathbb{Z}_p.$$

The number q is polynomially bounded in the security parameter λ (while p, the order of the group $\mathbb{G}$, is exponential in λ). These q elements of $\mathbb{G}$ are used to simulate $(q-1)$ signing oracle queries. This reliance makes q-SDH a *non-static* assumption, also called a q-type assumption, in contrast to the traditional *static* ones like computational DH.

1.1 Technical Overview

1.1.1 Dual Form Boneh-Boyen (or Gentry/Exponent-Inversion) Signatures

Dual form signatures is a framework from Gerbush *et al.* [GLOW12] for proving security based on static assumptions. The "form B" signatures can only be generated by an algorithm Sign_B but not the regular signing algorithm Sign_A. The security proof involves a sequence of transformations, transitioning from using Sign_A to Sign_B in answering signing oracle queries. It also involves a challenge signature that can take either form, depending on its randomness, but remains indistinguishable to adversaries. There are a few instantiations using composite-order groups, none of which involve the exponent-inversion structure.

In this paper, we propose the *dual form Boneh-Boyen signatures*, proving their security via dual form signatures using static assumptions. We then demonstrate how it helps eliminate non-static assumptions from a number of higher cryptographic applications. Simply instantiating the Boneh-Boyen signature in a composite-order group $\mathbb{G}_N$ for $N = p_1 p_2 p_3$ does not work. Consider randomizing the main structure in $\mathbb{G}_{p_1} = \langle g_1 \rangle$ as in the randomized Boneh-Boyen signatures [BB08]: $(g_1^{1/(\alpha+M+\beta r)}, r)$, where $(\alpha, \beta) \in (\mathbb{Z}_N)^2$ is the secret key, and $r \in \mathbb{Z}_N$. It is unclear how the signing oracle can simulate this structure for multiple M's without embedding powers of α in g_1 to enable inversion in the exponent.

We consider an alternative randomization of the $\mathbb{G}_{p_1}$ component, which applies the exponent inversion on a random base element rather than exponentiating a fixed base by a randomized inversion. We start with a signature similar to the key structure of Gentry-IBE [Gen06]: $(\sigma_1' = (h_1 g_1^{-r})^{\frac{1}{\alpha-M}}, \sigma_2' = r)$, where $h_1 \in \mathbb{G}_{p_1}$ comes from the public key. Given this root, we could refer our resulting scheme as *dual form Gentry signatures* or, more broadly, *dual form exponent-inversion signatures*. We still need a few more changes. First, we use h_1 as a private signing key, without relying on $g^{\alpha^2}, \ldots, g^{\alpha^q}$ for signing oracle simulation (*cf.*, [BB08]). The public key now includes $\hat{e}(g_1, h_1)$. Second, the hard problem instances used in typical dual form schemes [LW10, GLOW12] do not allow the leakage of the randomness r directly. One could set $\sigma_2' = g_1^r$ instead. Still, g_1^r, g_1^{α} from the public key, and message M uniquely determine $(g_1^{-r})^{\frac{1}{\alpha-M}}$ in σ_1'. So, such randomization does not suffice for enabling dual forms. Our final changes are introducing randomness in $\mathbb{G}_{p_3} = \langle g_3 \rangle$ and breaking the direct determination of $(g_1^{-r})^{\frac{1}{\alpha-M}}$ in σ_1' by replacing it with $(u_1^{-r})^{\frac{1}{\alpha-M}}$ using another generator $u_1 \in \mathbb{G}_{p_1}$. Our signature on M thus becomes:

$$(\sigma_1 = (h_1 u_1^{-r})^{\frac{1}{\alpha-M}} g_3^{x_1}, \quad \sigma_2 = g_1^r g_3^{x_2}),$$

where r, x_1, x_2 are randomly chosen from $\mathbb{Z}_N$. The signing oracles are simulated by using h_1 or $h_1 X_2$ for some random $X_2 \in \mathbb{G}_{p_2}$. This gives rise to the dual forms with different exponents mod p_2 depending on the problem instance used in different security games.

The partial secrecy of α remains crucial in the transition between two forms of signature simulation. We only give g_1^α to the adversary, which contains information about α mod p_1, but this is not correlated to α mod p_2 due to the Chinese remainder theorem. Such an information-theoretic argument ensures an indistinguishable simulation to adversaries.

1.1.2 Dual Form Gentry-IBE

Apart from exponent inversion, commutative blinding forms another major family of pairing-based IBE in the standard model [Boy07]. Typical reductions for this family incur a loss by a factor of q while using static assumptions. Gentry-IBE [Gen06], using exponent inversion in the keys, has a tight security reduction but relies on a q-type assumption.

Dual system encryption, developed by Waters [Wat09], is a framework aimed at constructing adaptively secure IBE schemes from static assumptions. It is later applied for hierarchical IBE [LW10] and other security features, such as security against related-key attacks [YZC22]. Existing IBE schemes based on dual system encryption [Wat09, LW10] are from the commutative blinding family.

This paper firstly presents an IBE scheme with a key structure based on exponent inversion but also the commutative blinding property across the key and the ciphertext for the session key derivation[1] (in the form $\hat{e}(g, h_1)^s$, where h_1 is a part of the master secret key in our case). It is secure in the standard model under static assumptions. We refer to our scheme as dual form Gentry-IBE, as it is based on dual system encryption [Wat09, LW10]. It is similar to Lewko-Waters IBE [LW10] in the commutative blinding framework, and hence shares similar efficiency with it. Specifically, their identity-based secret keys contain the master secret key as a fixed factor, whereas ours are obtained from exponent inversion.

Our result leads to an adaptively secure anonymous IBE scheme in composite/prime-order groups, which remains competitive with the “optimal” IBE scheme of Wee [Wee16], a candidate prime-order scheme without a security proof.

1.1.3 P-Signatures and Anonymous Credentials

We stress that our signature and IBE proposals serve not as an endpoint but as a foundation for the multitude of possibilities they can unlock. An important application is P-signatures [BCKL08]. P-signatures integrate with a commitment scheme and provide: (1) an interactive protocol for obtaining a signature on a committed (hidden) value, and (2) a non-interactive proof system for proving that a commitment contains a signed value. The commitment scheme needs to support proofs that two commitments correspond to the same value. Our signature scheme enables an anonymous credential system featuring non-interactive credential proof [BCKL08] without using any q-type non-static assumptions.

In more detail, one can use Groth-Sahai non-interactive zero-knowledge (NIZK) proof system [GS12] to prove the possession of structure-preserving signatures (SPS) or P-signatures as an anonymous credential.[2] To our knowledge, the most efficient SPS based on standard assumptions is the one by Abe *et al.* [ACD$^+$16]. The signature size of their SXDH-based instantiation is 11 group elements, while ours is 8.

[1] The session key of an exponent-inversion IBE is usually computed from the public parameters only.

[2] Gerbush*et al.* [GLOW12] showed how to prove the security of a variant of Camenisch-Lysyanskaya signatures [CL04] with static assumption. However, whether this scheme can be used in anonymous credentials is unclear since their Sign_A and Sign_B algorithms handle the message differently, which should be hidden in a commitment and associated with NIZK proof when used in an anonymous credential. In particular, the message appears in the signature produced by Sign_A algorithm twice but just once in Sign_B. We see no immediate simple solution without changing the scheme or the proof.

Table 1: Comparison of Accountable-Authority IBE

Scheme	Malicious PKG	Ciphertext Size	Traceability
Libert-Vergnaud [LV11]	Weak black-box	$O(1)$	Private
Sahai-Seyalioglu [SS11]	Black-box	$O(\lambda)$	Private
Lai-Deng-Zhao-Weng [LDZW13]	Weak black-box	$O(1)$	Public
Kiayias-Tang-I [KT15]	Weak black-box	$O(1)$	Private
Kiayias-Tang-III [KT15]	Weak black-box	$O(1)$	Public
Ours	Black-box	$O(1)$	Private

Our signature scheme also serves as an alternative instantiation for higher-level applications, including other signature notions such as group signatures (in Section 1.2.2).

1.1.4 Accountable-Authority IBE

An aim of studying our IBE construction is to inherit the promising properties of Gentry-IBE for higher applications. We focus on accountable-authority IBE (A-IBE) [Goy07, LV11]. A-IBE features a tracing algorithm that can determine if a decryption key ("white-box") or a decoder box ("black-box") was created by the (malicious) private key generator (PKG), so any party who leaks a key can be held accountable and proven guilty of key leakage.[3]

In Gentry-IBE, the identity-based secret key has a field element r generated solely by the PKG, which the user cannot re-randomize after key issuance. This locked-in randomness property is critical for key tracing, as it ensures that the secret key remains fixed and tied to a particular user. In the event of key leakage, the unchangeable r value allows tracing the leaked key back to its original user. If re-randomization were possible, users could modify their keys to evade detection, undermining accountability.

Our dual form Gentry-IBE scheme incorporates anonymity, interactive key generation protocol, and tracing algorithm to enable a fully secure black-box A-IBE scheme. The only existing scheme with this security level [SS11] relies on dummy identities to support black-box tracing with full security, incurring a multiplicative overhead of $O(\lambda)$ for both key and ciphertext sizes. Our dual form Gentry-IBE supports decryption oracle queries by using semi-functional keys without this extra overhead. Previous A-IBE constructions based on Gentry-IBE or generic (white-box traceable) constructions [KT15] are, at most, *weakly black-box traceable* [LV11] and do not allow decryption in arguing dishonest PKG security.

Table 1 compares existing schemes, with size measured by the number of group elements.

1.2 Related Works

We review related developments that generally have different goals from ours.

1.2.1 Accountable-Authority IBE

Lai *et al.* [LDZW13] and Kiayias and Tang [KT15] consider public traceability but operate under a weak black-box setting, which is less robust than ours. Additionally, Kiayias and Tang proposed generic transformations that add accountability to any IBE scheme by leveraging identity-reuse, a specific feature that allows users to request multiple keys for the same identity. Refer to Table 1 for a comparison of the major features.

[3] Dishonest PKG security of A-IBE [Goy07] does not mean that the PKG cannot generate a valid key for a user, but cannot identify which specific key family the user obtained a key from, due to the secret input of the user during the key request. This is different from other escrow-free notions, *e.g.*, anonymous ciphertext indistinguishability [Cho09, YZCL13].

Table 2: Comparison of Our Signature Scheme with Randomizable Schemes

Scheme	Assumption	Signature Size
Libert-Mouhartem-Peters-Yung [LMPY16]	SXDH	$4\|\mathbb{G}\|$
Pointcheval-Sanders [PS16]	Interactive	$2\|\mathbb{G}\|$
Pointcheval-Sanders [PS18]	q-type	$2\|\mathbb{G}\|$
Chatterjee-Kabaleeshwaran [CK19]	SXDH	$4\|\mathbb{G}\|$
Ours (Optimized)	SXDH	$4\|\mathbb{G}\|$

1.2.2 Randomizable Signatures and Group Signatures

Many randomizable signature schemes have been proposed, with some (eventually) proven secure under static assumptions. The scheme by Libert *et al.* [LMPY16] is based on the symmetric external Diffie-Hellman (SXDH) assumption. The scheme by Chatterjee and Kabaleeshwaran [CK19] is also based on the SXDH assumption. Similar to our work, they first proposed a composite-order group scheme based on the subgroup-hiding assumption and then its prime-order variant. Unlike us, both works aim for public rerandomizability. Pointcheval and Sanders [PS16] proposed a short randomizable signature scheme based on an interactive assumption, later improved [PS18] with a security proof under a variant of the q-SDH assumption. See Table 2 for a comparison, where the signature size counts the number of group elements without accounting for differences between the two base groups $\mathbb{G}_1$ and $\mathbb{G}_2$, such as size or efficiency when instantiated with different curves.

In some applications, such as cryptocurrency or blockchain, strong existential unforgeability is desired, and public randomizability can be harmful. Additionally, strongly-unforgeable BBS+ signatures [ASMC13], featuring the exponent-inversion structure, are one of the mainstreams in privacy-enhancing primitives. Secure multi-party computation techniques over this structure have been proposed [DKL$^+$23, WMC24] for threshold signing.

Recall that the original Boneh-Boyen signature was utilized to build two-level hierarchical signatures and eventually group signatures (notably, [BBS04]). Here, we highlight two examples that benefit from dual form signature schemes based on static assumptions. Chow *et al.* [CZZ17] propose a concurrently secure "real hidden" identity-based signature scheme without random oracles. This class of group signature schemes enables anonymity revocation without requiring any form of membership list "for real": even implicitly, such as a table indexed for storing discrete logarithm solutions needed to identify specific members. Chatterjee and Kabaleeshwaran [CK18] have further explored the application of dual form signatures, resulting in a dual form two-level hierarchical signature scheme and, eventually, a dual form group signature scheme under static assumptions.

1.2.3 More on Pairing-Based IBE and Other IBE Schemes

Pairing-based IBE fully secure in the standard model can be classified into two families [Boy07]: *commutative blinding* and *exponent inversion.* Roughly speaking, commutative blinding creates blinding factors from two secret coefficients in a way that makes them "commute" (*i.e.*, not depend on the application order) using pairings. In exponent-inversion IBE, the recipient's ID is embedded in the exponent via a secret function f, while keeping $g^{f(\mathsf{ID})}$ publicly computable. Consider a ciphertext having $(g^{f(\mathsf{ID})})^s$; decryption is done by pairing it with a private key of the form $\hat{g}^{1/f(\mathsf{ID})}$ to get a session key $\hat{e}(g, \hat{g})^s$.

Boyen [Boy07] proposed a framework capturing the properties of the exponent-inversion IBE (including Sakai-Kasahara IBE [SK03] and the second IBE scheme of Boneh-Boyen), referred to as linear IBE. It enables the construction of hierarchical IBE, fuzzy IBE, and attribute-based encryption [Boy07]. In our dual form Gentry-IBE, although its identity-based secret keys have an exponent-inversion structure, the ciphertext session key is

$\hat{e}(g_1, h_1)^s$, where h_1 is part of the master secret key. This resembles the commutative blinding family (a function of the master secret key and s) and does not belong to the linear IBE family.

Exponent-inversion IBE studied in this paper has a unique structure: its identity-based secret key is probabilistically generated but not publicly randomizable. We observe that this property, though not explicitly mentioned, plays a key role in leakage-resilient IBE [CDRW10]. In particular, the last scheme of Chow *et al.* [CDRW10] can be seen as extending their prior IBE schemes with this structure for attaining leakage resilience. Gentry-IBE is also the only IBE scheme in the standard model to achieve anonymous ciphertext indistinguishability, as shown in Chow's study [Cho09, Cho10].

We remark on the existence of alternative construction paradigms in IBE, such as lattice-based schemes (*e.g.*, [ABB10, CHKP12]) and non-black-box constructions [BLSV18, DG21, WC23], which provide diverse approaches and security guarantees.

1.2.4 Equivalence with q-SDH or Reduction to Boneh-Boyen Signatures

The Boneh-Boyen signature scheme is provably secure in the standard model under the q-SDH assumption [BB08]. The converse is also true [JY09], therefore, forging Boneh-Boyen signatures is equivalent to solving the q-SDH problem. However, we cannot reduce the security of our dual form Boneh-Boyen signatures to either the original scheme [BB08] or the q-SDH problem. This is because our scheme includes g^r as part of the signature, whereas the original scheme [BB08] only outputs r. In addition, our security proof requires the secrecy of both h_1 and α, while the q-SDH problem only guarantees the secrecy of α.

1.3 Revisiting q-Type Assumptions

Transitioning from q-type to static assumptions holds philosophical significance for removing the reliance of the assumption on adversarial actions. We see our work as an orthogonal approach to the Déjà Q frameworks of Chase *et al.* [CM14, CMM16]. Chase and Meiklejohn [CM14] reduced a number of q-type assumptions to the subgroup-hiding assumption in composite-order groups. Their reduction can be adapted to the Boneh-Boyen signatures. Compared to our work, theirs provides a more generic approach by working directly on the assumption, and hence can prove the security of some deterministic algorithms, such as the (modified) Dodis-Yampolskiy pseudorandom function [DY05] under static assumptions.

Specifically, they require the use of asymmetric pairings, with all secret parameters crucial to the assumption confined to one side of the pairing. Their reduction requires a decisional assumption regarding the base group elements. This rules out Gentry-IBE, with security relying on the q-augmented decisional bilinear Diffie-Hellman inversion problem [Gen06], which involves distinguishing a target group element from random.

In contrast, our study focuses on the scheme level, which is less generic but still covers an important class of pairing-based schemes, namely, the exponent-inversion framework. Specifically, we can now replace the use of decisional q-type assumptions involving secret values on both sides of the pairing and prove the security of our variant of Gentry-IBE and our A-IBE scheme based on static assumptions, which have not been achieved before [CM14]. Although our description starts with the composite-order group setting for an easier understanding of the essence, they can be instantiated by prime-order groups under the SXDH assumption. In general, prime-order-group instantiations are more efficient than their composite-order group counterparts.

Finally, we remark that the improved Déjà Q frameworks of Chase *et al.* [CMM16] cover more assumptions, particularly those over symmetric pairing groups, albeit requiring tailored efforts, namely, sorting out the dependency of graphs that reflect the usage of all values in the source groups and how they interact with each other and with the pairing.

2 Background

2.1 Notations, Pairing Groups, and Complexity Assumptions

Composite-Order Groups. Let $\mathcal{G}$ be a composite-order bilinear group context generator that takes a security parameter 1^λ as input, where $\lambda \in \mathbb{N}$, and outputs a description of bilinear group $(N = p_1p_2p_3, \mathbb{G}, \mathbb{G}_T, \hat{e})$, where p_1, p_2, p_3 are distinct λ-bit primes. $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of order N, and $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a symmetric bilinear map such that $\forall g, h \in \mathbb{G}$ and $a, b \in \mathbb{Z}_N$, $\hat{e}(g^a, h^b) = \hat{e}(g, h)^{ab}$. $\hat{e}(g, g)$ generates $\mathbb{G}_T$ if g is a generator of $\mathbb{G}$. All group operations, $\hat{e}$, and $\mathcal{G}$ run in probabilistic polynomial time (PPT).

For $i \in \{1, 2, 3\}$, let $\mathbb{G}_{p_i}$ denote the subgroup of order p_i in $\mathbb{G}$. Let g_i be a generator of $\mathbb{G}_{p_i}$. For all $h_i \in \mathbb{G}_{p_i}$ and $h_j \in \mathbb{G}_{p_j}$, if $i \neq j$, then $\hat{e}(h_i, h_j) = 1$. We also use $\mathbb{G}_{p_2p_3}$ to denote the subgroup of order p_2p_3 in $\mathbb{G}$. For all $T \in \mathbb{G}_{p_2p_3}$, T can be uniquely expressed as the product of an element from $\mathbb{G}_{p_2}$ and an element from $\mathbb{G}_{p_3}$. We refer to these elements as the "$\mathbb{G}_{p_2}$ part of T" and the "$\mathbb{G}_{p_3}$ part of T" respectively. We also use similar notations, *e.g.*, for $\mathbb{G}_{p_1p_2}$ and $\mathbb{G} = \mathbb{G}_{p_1p_2p_3}$. Finally, we assume $\mathcal{G}$ also outputs generators for certain subgroups of $\mathbb{G}$, namely, g_1, g_3, and $g_{2,3}$ for $\mathbb{G}_{p_1}$, $\mathbb{G}_{p_3}$, and $\mathbb{G}_{p_2p_3}$, respectively.

Prime-Order Groups. Let $\mathcal{G}$ be a prime-order bilinear group context generator that takes a security parameter 1^λ as input and outputs a description of bilinear group $(p, \mathbb{G}_1 = \langle g_1 \rangle, \mathbb{G}_2 = \langle g_2 \rangle, \mathbb{G}_T, \hat{e})$ and generators g_1, g_2, where p is prime. $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_T$ are cyclic groups of order p, and $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an asymmetric bilinear map.

For a fixed dimension n, we choose two random bases $\mathbb{B} := (\overrightarrow{b_1}, \ldots, \overrightarrow{b_n})$ and $\mathbb{B}^* := (\overrightarrow{b_1^*}, \ldots, \overrightarrow{b_n^*}))$ of $\mathbb{Z}_p^n$, subject to the constraint that they are "dual orthonormal" [OT08]. This means that $\overrightarrow{b_i} \cdot \overrightarrow{b_j^*} = 0 \bmod p$ for all $i \neq j$, and $\overrightarrow{b_i} \cdot \overrightarrow{b_i^*} = \psi$ for all i, where ψ is picked uniformly at random from $\mathbb{Z}_p$ and $\vec{a} \cdot \vec{b}$ denotes the inner product of vectors $\vec{a}$ and $\vec{b}$.

We define $\hat{e}_n$ as the product of the component-wise pairings for the vectors $\overrightarrow{v} = (v_1, \ldots, v_n)$, $\overrightarrow{w} = (w_1, \ldots, w_n)$:

$$\hat{e}_n(g_1^{\overrightarrow{v}}, g_2^{\overrightarrow{w}}) := \prod_{i=1}^{n} \hat{e}(g_1^{v_i}, g_2^{w_i}) = \hat{e}(g_1, g_2)^{\overrightarrow{v} \cdot \overrightarrow{w}}.$$

Choosing random dual orthonormal bases $(\mathbb{B}, \mathbb{B}^*)$ is equivalent to randomly choosing a basis $\mathbb{B}$ and a vector $\overrightarrow{b_1^*}$, subject to the constraint that it is orthogonal to $\overrightarrow{b_2}, \ldots, \overrightarrow{b_n}$. Then $\overrightarrow{b_2^*}$ is chosen so that it is orthogonal to $\overrightarrow{b_1}, \overrightarrow{b_3}, \ldots, \overrightarrow{b_n}$, and has dot product with $\overrightarrow{b_2}$ equal to $\overrightarrow{b_1} \cdot \overrightarrow{b_1^*}$, which is defined as ψ, and so on.

For a given dimension n and prime p, we define $(\mathbb{B}, \mathbb{B}^*) \leftarrow \mathsf{Dual}(\mathbb{Z}_p^n)$ to represent the selection of random dual orthonormal bases $\mathbb{B}$ and $\mathbb{B}^*$ of $\mathbb{Z}_p^n$.

Symmetric External Diffie-Hellman (SXDH) Assumption. First, we define the decisional Diffie-Hellman (DDH) assumption in $\mathbb{G}_1$ as follows. Given a prime-order bilinear group context generator $\mathcal{G}$, we define the following distribution:

$$\begin{aligned} &(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}) \xleftarrow{R} \mathcal{G}(1^\lambda), \quad g_1 \xleftarrow{R} \mathbb{G}_1, \quad g_2 \xleftarrow{R} \mathbb{G}_2, \quad a, b, c \xleftarrow{R} \mathbb{Z}_p, \\ &T_0 := g_1^{ab}, \quad T_1 := g_1^{ab+c}, \quad D := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}, g_1, g_2, g_1^a, g_1^b), \end{aligned}$$

where $\xleftarrow{R}$ denotes sampling uniformly at random or assignment of a random output from an algorithm. Assume that for any PPT algorithm $\mathcal{A}$ with output in $\{0, 1\}$, the advantage

$$\mathsf{Adv}_{\mathcal{G},\mathcal{A}} := |\Pr[(D, T_0) = 1] - \Pr[(D, T_1) = 1]| = \mathsf{negl}(\lambda).$$

By reversing the roles of $\mathbb{G}_1$ and $\mathbb{G}_2$ above, we obtain the DDH assumption in $\mathbb{G}_2$. The symmetric external Diffie-Hellman (SXDH) assumption holds if the DDH problems are intractable in both $\mathbb{G}_1$ and $\mathbb{G}_2$.

(k,n)-Decisional Subspace Assumption (in $\mathbb{G}_1$) [CLL+12]. Given a prime-order bilinear group context generator $\mathcal{G}$, we define the following distribution:

$$\begin{aligned}
&(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}) \overset{R}{\leftarrow} \mathcal{G}(1^\lambda), \quad (\mathbb{B}, \mathbb{B}^*) \leftarrow Dual(\mathbb{Z}_p^n),\\
&g_1 \overset{R}{\leftarrow} \mathbb{G}_1, \quad g_2 \overset{R}{\leftarrow} \mathbb{G}_2, \quad \tau_1, \tau_2, \mu_1, \mu_2 \overset{R}{\leftarrow} \mathbb{Z}_p,\\
&U_1 := g_2^{\mu_1 \overrightarrow{b_1^*} + \mu_2 \overrightarrow{b_{k+1}^*}}, \quad U_2 := g_2^{\mu_1 \overrightarrow{b_2^*} + \mu_2 \overrightarrow{b_{k+2}^*}}, \quad \dots, \quad U_k := g_2^{\mu_1 \overrightarrow{b_k^*} + \mu_2 \overrightarrow{b_{2k}^*}},\\
&V_1 := g_1^{\tau_1 \overrightarrow{b_1}}, \quad V_2 := g_1^{\tau_1 \overrightarrow{b_2}}, \quad \dots, \quad V_k := g_1^{\tau_1 \overrightarrow{b_k}},\\
&W_1 := g_1^{\tau_1 \overrightarrow{b_1} + \tau_2 \overrightarrow{b_{k+1}}}, \quad W_2 := g_1^{\tau_1 \overrightarrow{b_2} + \tau_2 \overrightarrow{b_{k+2}}}, \quad \dots, \quad W_k := g_1^{\tau_1 \overrightarrow{b_k} + \tau_2 \overrightarrow{b_{2k}}},\\
&D := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}, g_2^{\overrightarrow{b_1^*}}, \dots, g_2^{\overrightarrow{b_k^*}}, g_2^{\overrightarrow{b_{2k+1}^*}}, \dots, g_2^{\overrightarrow{b_n^*}}, g_1^{\overrightarrow{b_1}}, \dots, g_1^{\overrightarrow{b_n}}, U_1, \dots, U_k, \mu_2).
\end{aligned}$$

Assume that for any PPT algorithm $\mathcal{A}$ with output in $\{0,1\}$, the advantage

$$\mathsf{Adv}_{\mathcal{G},\mathcal{A}} := |\Pr[(D, V_1, \dots, V_k) = 1] - \Pr[(D, W_1, \dots, W_k) = 1]| = \mathsf{negl}(\lambda).$$

By reversing the roles of $\mathbb{G}_1$ and $\mathbb{G}_2$ above, we obtain the subspace assumption in $\mathbb{G}_2$. The SXDH assumption implies the decisional subspace assumption [CLL+12].

2.2 Formal Model of Identity-Based Encryption

An IBE scheme consists of four PPT algorithms:

- Setup: On input of a security parameter 1^λ, it outputs a system parameter param and a master public/private key pair $(\mathsf{mpk}, \mathsf{msk})$. The public parameter param implicitly defines an identity space $\mathcal{I}$ and a message space $\mathcal{M}$. It is treated as an implicit input of all other algorithms, and is omitted for simplicity.
- $\mathsf{Extract}$: On msk and an identity $\mathsf{ID} \in \mathcal{I}$, it outputs an identity-based secret key $\mathsf{sk}_{\mathsf{ID}}$.
- Enc: On input of mpk, ID, and a message $M \in \mathcal{M}$, it outputs a ciphertext C.
- Dec: On input of $\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}$, and C, it outputs a message M or $\perp$ for failed decryption.

Correctness. For all $M \in \mathcal{M}$, $\mathsf{ID} \in \mathcal{I}$; $M = \mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, \mathsf{Enc}(\mathsf{mpk}, \mathsf{ID}, M))$, where $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and $\mathsf{sk}_{\mathsf{ID}} \leftarrow \mathsf{Extract}(\mathsf{msk}, \mathsf{ID})$.

Confidentiality. A PPT adversary $\mathcal{A}$ plays the indistinguishability-based game below to launch adaptive chosen-identity and plaintext attacks (IND-ID-CPA) [BF03].

1. Setup. The challenger $\mathcal{C}$ runs $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and gives mpk to $\mathcal{A}$.
2. $\mathsf{Query\ 1}$. $\mathcal{A}$ can adaptively query the following oracles:
   - Extraction Oracle $\mathcal{KEO}(\mathsf{ID})$: On input of an identity $\mathsf{ID} \in \mathcal{I}$, it returns an identity-based secret key $\mathsf{sk}_{\mathsf{ID}} \leftarrow \mathsf{Extract}(\mathsf{msk}, \mathsf{ID})$.
3. $\mathsf{Challenge}$. $\mathcal{A}$ sends two messages $M_0^*, M_1^* \in \mathcal{M}$, and an identity $\mathsf{ID}^* \in \mathcal{I}$ to $\mathcal{C}$. $\mathcal{C}$ picks a random bit b', computes $C^* \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ID}^*, M_{b'}^*)$, and sends C^* to $\mathcal{A}$.
4. $\mathsf{Query\ 2}$. $\mathcal{A}$ is allowed to query the above oracles adaptively.
5. Output. $\mathcal{A}$ returns a guess b^* for b'.

$\mathcal{A}$ wins the game if $b' = b^*$ with no $\mathcal{KEO}(\mathsf{ID}^*)$ query was issued. The advantage of $\mathcal{A}$ is the probability of winning the game minus 1/2. An IBE scheme is IND-ID-CPA secure if there is no PPT $\mathcal{A}$ with a non-negligible advantage.

2.3 Formal Model of Accountable-Authority IBE

An A-IBE scheme consists of five PPT algorithms:

- Setup: On input of a security parameter 1^λ, it outputs a system parameter param and a master public/private key pair $(\mathsf{mpk}, \mathsf{msk})$. The public parameter param implicitly defines an identity space $\mathcal{I}$ and a message space $\mathcal{M}$. It is treated as an implilcit input of all other algorithms, and is omitted for simplicity.

- $\mathsf{Extract}$: On input of msk and an identity ID from the identity space $\mathcal{I}$, it engages in an interactive protocol with the user. The user receives an identity-based secret key $\mathsf{sk}_{\mathsf{ID}}$ in the end. Note that the issuer may not know the exact key that the user obtains.

- Enc: On input of mpk, ID, and a message $M \in \mathcal{M}$, it outputs a ciphertext C.

- Dec: On input of mpk, $\mathsf{sk}_{\mathsf{ID}}$, and C, it outputs a message M or $\perp$ if decryption fails.

- $\mathsf{Trace}^{\mathbf{D}}$: On input of mpk, $\mathsf{sk}_{\mathsf{ID}}$, and black-box accesses to an ϵ-useful decoder box $\mathbf{D}$ (defined below) for an identity ID, the algorithm determines if $\mathbf{D}$ was created by the PKG or the user ID.

Correctness. For all $M \in \mathcal{M}$, $\mathsf{ID} \in \mathcal{I}$; $M = \mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, \mathsf{Enc}(\mathsf{mpk}, \mathsf{ID}, M))$, where $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and $\mathsf{sk}_{\mathsf{ID}} \leftarrow \mathsf{Extract}(\mathsf{msk}, \mathsf{ID})$. Some A-IBE schemes only require correctness to hold with an overwhelming probability [SS11].

Usefulness of Decoder. For non-negligible ϵ, a PPT algorithm $\mathbf{D}$ is an ϵ-useful decoder box for an identity ID if $\Pr[M \leftarrow \mathcal{M} : \mathbf{D}(\mathsf{Enc}(\mathsf{mpk}, \mathsf{ID}, M)) = M] \geq \epsilon$.

Confidentiality. We consider the following indistinguishability-based game against adaptive chosen-identity and chosen-ciphertext attacks (IND-ID-CCA).

1. Setup. The challenger $\mathcal{C}$ runs $(\mathsf{param}, \mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$, withholds msk, and gives $(\mathsf{param}, \mathsf{mpk})$ to the adversary $\mathcal{A}$.

2. $\mathsf{Query\ 1}$. $\mathcal{A}$ can adaptively query the following oracles:

   - Extraction Oracle $\mathcal{KEO}(\mathsf{ID})$: On input of an identity $\mathsf{ID} \in \mathcal{I}$, it returns an identity-based secret key $\mathsf{sk}_{\mathsf{ID}} \leftarrow \mathsf{Extract}(\mathsf{msk}, \mathsf{ID})$.
   - Decryption Oracle $\mathcal{DO}(\mathsf{ID}, C)$: On input of an identity $\mathsf{ID} \in \mathcal{I}$ and a ciphertext C, it returns the decryption result $\mathsf{Dec}(\mathsf{mpk}, \mathsf{Extract}(\mathsf{msk}, \mathsf{ID}), C)$.

3. $\mathsf{Challenge}$. $\mathcal{A}$ sends two messages $M_0^*, M_1^* \in \mathcal{M}$, and an identity $\mathsf{ID}^* \in \mathcal{I}$ to $\mathcal{C}$. $\mathcal{C}$ picks a random bit b', computes $C^* \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ID}^*, M_{b'}^*)$, and sends C^* to $\mathcal{A}$.

4. $\mathsf{Query\ 2}$. $\mathcal{A}$ is allowed to query the above oracles adaptively.

5. Output. $\mathcal{A}$ returns a guess b^* for b'.

$\mathcal{A}$ wins the game if $b' = b^*$. We require that there was no $\mathcal{KEO}(\mathsf{ID}^*)$ or $\mathcal{DO}(\mathsf{ID}^*, C^*)$ query was made. The advantage of $\mathcal{A}$ is the probability of winning the game minus 1/2. An A-IBE scheme is IND-ID-CCA secure if there is no PPT $\mathcal{A}$ with a non-negligible advantage.

Dishonest User Security. We consider the ComputeNewKey(-CCA) game against adaptive chosen-identity attacks (and chosen-ciphertext attacks) below.

1. Setup. The challenger runs $(\mathsf{param}, \mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$, withholds msk, and gives $(\mathsf{param}, \mathsf{mpk})$ to the adversary $\mathcal{A}$.

2. Query. $\mathcal{A}$ can adaptively query the following oracles:

- Extraction Oracle $\mathcal{KEO}(\mathsf{ID})$: On input of an identity $\mathsf{ID} \in \mathcal{I}$, it returns an identity-based secret key $\mathsf{sk}_{\mathsf{ID}} \leftarrow \mathsf{Extract}(\mathsf{msk}, \mathsf{ID})$.
- Decryption Oracle $\mathcal{DO}(\mathsf{ID}, C)$: On input of an identity $\mathsf{ID} \in \mathcal{I}$ and a ciphertext C, it returns the decryption result $\mathsf{Dec}(\mathsf{mpk}, \mathsf{Extract}(\mathsf{msk}, \mathsf{ID}), C)$.

3. Output. Adversary $\mathcal{A}$ outputs an ϵ-useful decoder box $\mathbf{D}^*$ and a key $\mathsf{sk}_{\mathsf{ID}^*}$ for ID^*.

Adversary $\mathcal{A}$ wins the game if $\mathsf{Trace}^{\mathbf{D}^*}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}^*}, \epsilon) =$ 'PKG'. An A-IBE scheme is said to be ComputeNewKey-CCA secure if there is no PPT adversary $\mathcal{A}$ with a non-negligible advantage in winning the game above. The extra decryption oracle given to $\mathcal{A}$ may help it create a decoder box $\mathbf{D}^*$ since $\mathbf{D}^*$ mimics the function of a decryption oracle. Thus, the ComputeNewKey-CCA model is stronger than its CPA variant.

Dishonest PKG Security. We consider the FindNewKey game against adaptive chosen-ciphertext attacks (FindNewKey-CCA) [SS11].

1. $\mathsf{Initialize}$. The challenger gives param to the adversary $\mathcal{A}$.
2. Setup. $\mathcal{A}$ gives the master public key mpk and an identity $\mathsf{ID}^* \in \mathcal{I}$ to the challenger. The challenger aborts if they are not well-formed.
3. $\mathsf{Extract}$. The challenger and $\mathcal{A}$ engage in the extract protocol for ID^*. If neither party aborts, the challenger receives $\mathsf{sk}_{\mathsf{ID}^*}$ as output.
4. Query. $\mathcal{A}$ can adaptively query the following oracles:
   - Decryption Oracle $\mathcal{DO}(C)$: On input of a ciphertext C, it returns the decryption result $M/\perp \leftarrow \mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}^*}, C)$.
5. Output. Adversary $\mathcal{A}$ outputs an ϵ-useful decoder box $\mathbf{D}^*$.

Adversary $\mathcal{A}$ wins the game if $\mathsf{Trace}^{\mathbf{D}^*}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}^*}, \epsilon) =$ 'User'. An A-IBE scheme is FindNewKey-CCA secure if no PPT adversary $\mathcal{A}$ can win with a non-negligible advantage.

2.4 Formal Model of Non-interactive P-signatures

A non-interactive P-signature scheme extends a signature scheme $(\mathsf{KG}, \mathsf{Sign}, \mathsf{Verify})$ and a non-interactive commitment scheme $(\mathsf{Setup}, \mathsf{Commit})$ with the algorithms below [BCKL08].

- Setup: On input of a security parameter 1^λ, it generates public parameters param. These include parameters for the signature and commitment schemes.
- $\mathsf{ObtainSig} \leftrightarrow \mathsf{IssueSig}$: These two interactive algorithms execute a signature-issuing protocol between a user and an issuer. The user takes as input $(\mathsf{param}, \mathsf{pk}, M, C, \mathsf{Open})$ such that the value $C = \mathsf{Commit}(\mathsf{param}, M, \mathsf{Open})$ and obtains a signature σ as output. The issuer takes $(\mathsf{param}, \mathsf{sk}, C)$ as input and gets nothing as output.
- Prove: On input $(\mathsf{param}, \mathsf{pk}, M, \sigma)$, it outputs $\mathsf{comm} = \mathsf{Commit}(\mathsf{param}, M, \mathsf{Open})$ and π as a proof of knowledge of a signature σ on M. This algorithm is non-interactive.
- $\mathsf{VerifyPf}$: On input of a commitment to a message M and a proof π that the message has been signed by the owner of public key pk, it outputs 1 if π is a valid proof of knowledge of $F(M)$ and a signature on M where F is a given function; 0 otherwise.
- $\mathsf{EqCommProve}$: On input of a message and two commitment opening values, it outputs a proof π that $\mathsf{comm} = \mathsf{Commit}(M, \mathsf{Open})$ is a commitment to the same value as $\mathsf{comm}' = \mathsf{Commit}(M, \mathsf{Open}')$.

- $\mathsf{EqCommVerify}$: On input of two commitments $\mathsf{comm}, \mathsf{comm}'$ and a proof π, it outputs 1 if π is a proof that $\mathsf{comm}, \mathsf{comm}'$ commit to the same value; 0 otherwise.

Correctness. $\mathsf{VerifyPf}(\mathsf{comm}, \pi) = 1$, for all $\lambda \in \mathbb{N}^+$, $(\mathsf{param}, \mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda)$, $C \leftarrow \mathsf{Commit}(\mathsf{param}, M, \mathsf{Open})$, $\sigma \leftarrow \mathsf{ObtainSig}(\mathsf{param}, \mathsf{pk}, M, C, \mathsf{Open}) \leftrightarrow \mathsf{IssueSig}(\mathsf{param}, \mathsf{sk}, C)$, $(\mathsf{comm}, \pi) \leftarrow \mathsf{Prove}(\mathsf{param}, \mathsf{pk}, M, \sigma)$.

Signer Privacy. A P-signature scheme has signer privacy if there exists a simulator $\mathsf{SimIssue}$ such that no PPT $\mathcal{A}$ can obtain any non-negligible advantage in the game below:

1. Setup. The challenger runs $(\mathsf{param}, \mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda)$ and gives $(\mathsf{param}, \mathsf{pk}, \mathsf{sk})$ to $\mathcal{A}$.
2. $\mathsf{Challenge}$. $\mathcal{A}$ sends a message M and an opening Open to the challenger. The challenger picks a random bit b' and computes $C \leftarrow \mathsf{Commit}(\mathsf{param}, M, \mathsf{Open})$. If $b' = 0$, the challenger runs $\mathsf{IssueSig}(\mathsf{param}, \mathsf{sk}, C)$ with $\mathcal{A}$; otherwise, computes $\sigma \leftarrow \mathsf{Sign}(\mathsf{param}, \mathsf{sk}, M)$ and runs $\mathsf{SimIssue}(\mathsf{param}, C, \sigma)$ with $\mathcal{A}$.
3. Output. $\mathcal{A}$ returns a guess b^* for b'. The advantage of $\mathcal{A}$ is the absolute difference between the probability of $b^* = b'$ and $1/2$.

User Privacy. A P-signature scheme has user privacy if there exists a simulator $\mathsf{SimIssue}$ such that no PPT adversary $\mathcal{A}$ can obtain any non-negligible advantage in the game below:

1. Setup. The challenger runs $(\mathsf{param}, \cdot, \cdot) \leftarrow \mathsf{Setup}(1^\lambda)$ and gives param to adversary $\mathcal{A}$.
2. $\mathsf{Challenge}$. $\mathcal{A}$ sends a public key pk, a message M, and an opening Open to the challenger. The challenger computes $C \leftarrow \mathsf{Commit}(\mathsf{param}, M, \mathsf{Open})$ and picks a random bit b'. If $b' = 0$, the challenger runs $\mathsf{ObtainSig}(\mathsf{param}, \mathsf{pk}, M, C, \mathsf{Open})$ with $\mathcal{A}$; otherwise, it runs $\mathsf{SimObtain}(\mathsf{param}, \mathsf{pk}, C)$ with $\mathcal{A}$.
3. Output. $\mathcal{A}$ returns a guess b^* for b'. The advantage of $\mathcal{A}$ is the absolute difference between the probability of $b^* = b'$ and $1/2$.

Zero-knowledge. A P-signature scheme is zero-knowledge if there exists a simulator $(\mathsf{SimSetup}, \mathsf{SimProve}, \mathsf{SimEqComm})$ such that, for all PPT adversaries $\mathcal{A}$, under parameters output by $\mathsf{SimSetup}$, Commit is perfectly hiding and

(1) the parameters output by $\mathsf{SimSetup}$ are indistinguishable from those output by Setup, but $\mathsf{SimSetup}$ also outputs a special auxiliary string aux,

(2) when param is generated by $\mathsf{SimSetup}$, outputs of $\mathsf{SimProve}(\mathsf{param}, \mathsf{aux}, \mathsf{pk})$ are indistinguishable from those of $\mathsf{Prove}(\mathsf{param}, \mathsf{pk}, M, \sigma)$ for all (pk, M, σ) where $\sigma \in \sigma_{\mathsf{pk}}(M)$, and

(3) if param is generated by $\mathsf{SimSetup}$, outputs of $\mathsf{SimEqComm}(\mathsf{param}, \mathsf{aux}, \mathsf{comm}, \mathsf{comm}')$ are indistinguishable from those of $\mathsf{EqCommProve}(\mathsf{param}, M, \mathsf{Open}, \mathsf{Open}')$, $\forall M$, Open, and Open' such that $\mathsf{comm} = \mathsf{Commit}(\mathsf{param}, M, \mathsf{Open})$ and $\mathsf{comm}' = \mathsf{Commit}(\mathsf{param}, M, \mathsf{Open}')$.

Unforgeability. A P-signature scheme is unforgeable if there exists a PPT extractor $(\mathsf{ExtractSetup}, \mathsf{Extract})$ and a bijective function F such that, for all PPT adversaries $\mathcal{A}$,

(1) the output of $\mathsf{ExtractSetup}(1^\lambda)$ is indistinguishable from the output of $\mathsf{Setup}(1^\lambda)$, and

(2) $\mathcal{A}$ cannot output a proof π that $\mathsf{VerifyPf}$ outputs 1, but from which we extract $F(M)$ and σ such that either (a) σ is not a valid signature on M, (b) C is not a commitment to M, or (c) $\mathcal{A}$ has never previously queried the signing oracle on M.

3 Exponent-Inversion Identity-Based Encryption

3.1 Dual Form Gentry-IBE

- $\mathsf{Setup}(1^\lambda)$: The PKG runs the composite-order bilinear group context generator $\mathcal{G}(1^\lambda)$ to get $(N = p_1p_2p_3, \mathbb{G}, \mathbb{G}_T, \hat{e})$ and generators $g_1 \in \mathbb{G}_{p_1}$ and $g_3 \in \mathbb{G}_{p_3}$. The

PKG randomly picks $\alpha \in \mathbb{Z}_N$, $u_1, h_1 \in \mathbb{G}_{p_1}$. The master public key is $\mathsf{mpk} = (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g_1, u_1, \hat{e}(g_1, h_1), g_1^{\alpha})$. The master secret key is $\mathsf{msk} = (\alpha, h_1, g_3)$.

- $\mathsf{Extract}(\mathsf{msk}, \mathsf{ID})$: The PKG randomly picks $r \in \mathbb{Z}_N$ and $X_3, X_3' \in \mathbb{G}_{p_3}$, and outputs[4] $\mathsf{sk}_{\mathsf{ID}} = (K_1 = (h_1 u_1^{-r})^{\frac{1}{\alpha - \mathsf{ID}}} X_3,\ K_2 = g_1^r X_3')$.
- $\mathsf{Enc}(\mathsf{mpk}, \mathsf{ID}, M)$: To encrypt $M \in \mathbb{G}_T$ for $\mathsf{ID} \in \mathbb{Z}_N \backslash \{\alpha\}$, the sender randomly picks $s \in \mathbb{Z}_N$ and outputs $C = (C_0 = M \cdot \hat{e}(g_1, h_1)^s, C_1 = g_1^{s(\alpha - \mathsf{ID})}, C_2 = u_1^s)$.
- $\mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, C)$: Parse $\mathsf{sk}_{\mathsf{ID}} = (K_1, K_2)$, return $M = C_0 / \hat{e}(C_1, K_1) \cdot \hat{e}(C_2, K_2)$.

Theorem 1. *Our IBE scheme is IND-ID-CPA secure under Assumptions 1, 2, and 3.*

Assumptions 1, 2, and 3 and the proof are provided in Appendix A.

Prime-Order Group Version. We can convert our IBE scheme into its prime-order version by the method of Chen *et al.* [CLL+12]. With the notations for dual pairing vector spaces and dual orthonormal bases in Section 2, the prime-order version is presented below.

- $\mathsf{Setup}(1^\lambda)$: The PKG runs the prime-order bilinear group context generator $\mathcal{G}(1^\lambda)$ to get $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e})$ and generators $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$. It samples random dual orthonormal bases $(\mathbb{D}, \mathbb{D}^*) \leftarrow \mathsf{Dual}(\mathbb{Z}_p^4)$. We let $\overrightarrow{d_1}, \ldots, \overrightarrow{d_4}$ denote the elements of $\mathbb{D}$ and $\overrightarrow{d_1^*}, \ldots, \overrightarrow{d_4^*}$ denote the elements of $\mathbb{D}^*$. The PKG randomly picks $\alpha, \beta \in \mathbb{Z}_p$ and computes $g_T = \hat{e}(g_1, g_2)^{\overrightarrow{d_1} \cdot \overrightarrow{d_1^*}}$. The master public key is $\mathsf{mpk} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}, g_1^{\overrightarrow{d_1}}, g_1^{\overrightarrow{d_2}}, g_1^{\alpha\overrightarrow{d_1}}, g_T^{\beta})$. The master secret key is $\mathsf{msk} = (g_2^{\overrightarrow{d_1^*}}, g_2^{\overrightarrow{d_2^*}}, g_2^{\beta\overrightarrow{d_1^*}}, \alpha)$.
- $\mathsf{Extract}(\mathsf{msk}, \mathsf{ID})$: The PKG randomly picks $r \in \mathbb{Z}_p$ and outputs $\mathsf{sk}_{\mathsf{ID}} = (\overrightarrow{K} = g_2^{(\frac{\beta - r}{\alpha - \mathsf{ID}})\overrightarrow{d_1^*} + r\overrightarrow{d_2^*}})$.
- $\mathsf{Enc}(\mathsf{mpk}, \mathsf{ID}, M)$: To encrypt a message $M \in \mathbb{G}_T$ for $\mathsf{ID} \in \mathbb{Z}_p$, the sender randomly picks $s \in \mathbb{Z}_p$ and outputs $C = (C_0, C_1)$, where $C_0 = M \cdot g_T^{\beta \cdot s}, \overrightarrow{C_1} = g_1^{s(\alpha - \mathsf{ID})\overrightarrow{d_1} + s\overrightarrow{d_2}}$.
- $\mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, C)$: Given a ciphertext $C = (C_0, \overrightarrow{C_1})$ and a secret key $\mathsf{sk}_{\mathsf{ID}} = \overrightarrow{K}$, the recipient calculates $M = C_0 / \hat{e}_4(\overrightarrow{C_1}, \overrightarrow{K})$.

Theorem 2. *Our prime-order IBE scheme is IND-ID-CPA secure under SXDH.*

Proof. We prove this by a hybrid argument using a sequence of games. The first game $\text{Game}_{\text{real}}$ is the IND-ID-CPA game. Let the challenge identity be ID^*. Let q be the number of extraction oracle queries.

For $k \in \{0, \ldots, q\}$, we define Game_k the same as $\text{Game}_{\text{real}}$, except that the challenge ciphertext is semi-functional (SF), and the keys used to answer first k oracle queries are SF. We define SF key or SF ciphertext, which is distributed like their normal version $\overrightarrow{K}$ or $(C_0, \overrightarrow{C_1})$ but "perturbed" by $\overrightarrow{d_3}, \overrightarrow{d_4}, \overrightarrow{d_3^*}, \overrightarrow{d_4^*}$ in the exponent as follows.

- An *SF key* is in the form of $\overrightarrow{K}' = \overrightarrow{K} \cdot g_2^{\gamma_3\overrightarrow{d_3^*} + \gamma_4\overrightarrow{d_4^*}}$, where $\gamma_3, \gamma_4 \in \mathbb{Z}_p$.
- An *SF ciphertext* is in the form of $C_0' = C_0, \overrightarrow{C_1}' = \overrightarrow{C_1} \cdot g_1^{\delta_3\overrightarrow{d_3} + \delta_4\overrightarrow{d_4}}$, where $\delta_3, \delta_4 \in \mathbb{Z}_p$.

The last game is $\text{Game}_{\text{final}}$, the same as Game_q, except that the challenge ciphertext is an SF encryption of a random message but not any one of the two challenge messages.

We proceed by proving the indistinguishability between these games.

[4] Similar to other exponent-inversion schemes, if the inverse of $\alpha - \mathsf{ID} \bmod N$ does not exist, the algorithm outputs $\perp$. We ignore this negligible case for the rest of the paper.

Lemma 1. *We can construct an algorithm $\mathcal{B}$ with advantage ϵ in breaking SXDH if there exists $\mathcal{A}$ such that* $\mathsf{Adv}_{\mathcal{A}}(\mathit{Game}_{\mathit{real}}) - \mathsf{Adv}_{\mathcal{A}}(\mathit{Game}_0) = \epsilon$.

Lemma 2. *We can construct an algorithm $\mathcal{B}$ with advantage ϵ in breaking SXDH if there exists $\mathcal{A}$ such that* $\mathsf{Adv}_{\mathcal{A}}(\mathit{Game}_{\ell-1}) - \mathsf{Adv}_{\mathcal{A}}(\mathit{Game}_{\ell}) = \epsilon$.

Lemma 3. *For any adversary $\mathcal{A}$,* $\mathsf{Adv}_{\mathcal{A}}(\mathit{Game}_q) - \mathsf{Adv}_{\mathcal{A}}(\mathit{Game}_{\mathit{final}}) = \mathsf{negl}(\lambda)$.

In $\text{Game}_{\text{final}}$, the value of b' is information-theoretically hidden from $\mathcal{A}$. Hence, $\mathcal{A}$ has no advantage in winning $\text{Game}_{\text{final}}$. □

Proof of Lemma 1. Given $D = (g_2^{\overrightarrow{b_1^*}}, g_2^{\overrightarrow{b_2^*}}, g_1^{\overrightarrow{b_1}}, \ldots, g_1^{\overrightarrow{b_4}}, U_1, U_2, \mu_2)$ along with $T = (T_1, T_2)$, which is either (V_1, V_2) or (W_1, W_2) from $(2,4)$-decisional subspace assumption in $\mathbb{G}_1$, $\mathcal{B}$ can simulate $\text{Game}_{\text{real}}$ or Game_0 with $\mathcal{A}$. First, $\mathcal{B}$ chooses a random invertible matrix $\mathbf{A} \in \mathbb{Z}_q^{2\times 2}$, and implicitly sets $\overrightarrow{d_1} = \overrightarrow{b_1}$, $\overrightarrow{d_2} = \overrightarrow{b_2}$, $(\overrightarrow{d_3}, \overrightarrow{d_4}) = (\overrightarrow{b_3}, \overrightarrow{b_4})\mathbf{A}$, $\overrightarrow{d_1^*} = \overrightarrow{b_1^*}$, $\overrightarrow{d_2^*} = \overrightarrow{b_2^*}$, $(\overrightarrow{d_3^*}, \overrightarrow{d_4^*}) = (\overrightarrow{b_3^*}, \overrightarrow{b_4^*})(\mathbf{A}^{-1})^{\mathsf{T}}$. $\mathcal{B}$ chooses random α, β and computes $\mathsf{mpk} = (g_1^{\overrightarrow{b_1}}, g_1^{\overrightarrow{b_2}}, (g_1^{\overrightarrow{b_1}})^{\alpha}, g_2^{\overrightarrow{b_2^*}}, \hat{e}(g_1^{\overrightarrow{b_1}}, g_2^{\overrightarrow{b_1^*}})^{\beta})$ and $\mathsf{msk} = (g_2^{\overrightarrow{b_1^*}}, (g_2^{\overrightarrow{b_1^*}})^{\beta}, \alpha)$ according to Setup. For extraction oracle queries, $\mathcal{B}$ answers by calculating $\mathsf{sk}_{\mathsf{ID}}$ using msk.

When $\mathcal{A}$ sends $\mathcal{B}$ two messages M_0^*, M_1^*, and an identity ID^*, $\mathcal{B}$ randomly picks a bit $b' \in \{0,1\}$. $\mathcal{B}$ calculates the challenge ciphertext as:

$$C_0^* = M_{b'}^* \cdot \hat{e}(T_1, g_2^{\overrightarrow{b_1^*}})^{\beta}, \quad C_1^* = T_1^{(\alpha - \mathsf{ID}^*)} T_2$$

If $T = (g_1^{\tau_1 \overrightarrow{b_1}}, g_1^{\tau_1 \overrightarrow{b_2}})$, this is a normal ciphertext with $s = \tau_1$, and hence $\mathcal{B}$ simulates $\text{Game}_{\text{real}}$. If $T = (g_1^{\tau_1 \overrightarrow{b_1} + \tau_2 \overrightarrow{b_3}}, g_1^{\tau_1 \overrightarrow{b_2} + \tau_2 \overrightarrow{b_4}})$, C_1^* has an additional term of $(\tau_2(\alpha - \mathsf{ID}^*)\overrightarrow{b_3} + \tau_2 \overrightarrow{b_4})$ in its exponent. As $\mathbf{A}$ is random, these coefficients are also random in the basis of $\overrightarrow{d_3}, \overrightarrow{d_4}$. $\mathcal{B}$ thus simulates Game_0. If $\mathcal{A}$ can distinguish between $\text{Game}_{\text{real}}$ and Game_0, $\mathcal{B}$ can break SXDH.

Proof of Lemma 2. Given $D = (g_1^{\overrightarrow{b_1}}, g_1^{\overrightarrow{b_2}}, g_2^{\overrightarrow{b_1^*}}, \ldots, g_2^{\overrightarrow{b_4^*}}, U_1, U_2, \mu_2)$ along with $T = (T_1, T_2)$, which is either (V_1, V_2) or (W_1, W_2) from $(2,4)$-decisional subspace assumption in $\mathbb{G}_2$, $\mathcal{B}$ can simulate $\text{Game}_{\ell-1}$ or Game_{ℓ} with $\mathcal{A}$. $\mathcal{B}$ first chooses a random invertible matrix $\mathbf{A} \in \mathbb{Z}_q^{2\times 2}$, and implicitly sets $\overrightarrow{d_1} = \overrightarrow{b_1}$, $\overrightarrow{d_2} = \overrightarrow{b_2}$, $(\overrightarrow{d_3}, \overrightarrow{d_4}) = (\overrightarrow{b_3}, \overrightarrow{b_4})\mathbf{A}$, $\overrightarrow{d_1^*} = \overrightarrow{b_1^*}$, $\overrightarrow{d_2^*} = \overrightarrow{b_2^*}$, $(\overrightarrow{d_3^*}, \overrightarrow{d_4^*}) = (\overrightarrow{b_3^*}, \overrightarrow{b_4^*})(\mathbf{A}^{-1})^{\mathsf{T}}$. $\mathcal{B}$ chooses random α, β and computes $\mathsf{mpk} = (g_1^{\overrightarrow{b_1}}, g_1^{\overrightarrow{b_2}}, (g_1^{\overrightarrow{b_1}})^{\alpha}, g_2^{\overrightarrow{b_2^*}}, \hat{e}(g_1^{\overrightarrow{b_1}}, g_2^{\overrightarrow{b_1^*}})^{\beta})$ and $\mathsf{msk} = (g_2^{\overrightarrow{b_1^*}}, (g_2^{\overrightarrow{b_1^*}})^{\beta}, \alpha)$ according to Setup. When $\mathcal{A}$ makes its k^{th} distinct extraction oracle query for ID_k:

- If $k < \ell$, $\mathcal{B}$ calculates the normal key $\mathsf{sk}_{\mathsf{ID}_k}$ using msk.
- If $k > \ell$, $\mathcal{B}$ calculates the normal key $\mathsf{sk}_{\mathsf{ID}_k}$ using msk. $\mathcal{B}$ randomly picks $\gamma_3, \gamma_4 \in \mathbb{Z}_p$ and calculates the SF key $\mathsf{sk}'_{\mathsf{ID}_k} = \mathsf{sk}_{\mathsf{ID}_k} \cdot (g_2^{\overrightarrow{d_3^*}})^{\gamma_3} \cdot (g_2^{\overrightarrow{d_4^*}})^{\gamma_4}$.
- If $k = \ell$, $\mathcal{B}$ chooses random $X_3', X_3'' \in \mathbb{G}_{p_3}$ and calculates the key $\mathsf{sk}_{\mathsf{ID}_\ell}$:

$$(g_2^{\overrightarrow{b_1^*}})^{\frac{\beta}{\alpha - \mathsf{ID}}} \cdot (T_1)^{\frac{-1}{\alpha - \mathsf{ID}}} \cdot T_2.$$

If $T = (g_2^{\tau_1 \overrightarrow{b_1^*}}, g_2^{\tau_1 \overrightarrow{b_2^*}})$, it is a normal key with $r = \tau_1$. Hence $\mathcal{B}$ simulates $\text{Game}_{\ell-1}$. If $T = (g_2^{\tau_1 \overrightarrow{b_1^*} + \tau_2 \overrightarrow{b_3^*}}, g_2^{\tau_1 \overrightarrow{b_2^*} + \tau_2 \overrightarrow{b_4^*}})$, then $\mathsf{sk}_{\mathsf{ID}_\ell}$ has an additional term of $(\frac{-\tau_2}{\alpha - \mathsf{ID}} \overrightarrow{b_3^*} + \tau_2 \overrightarrow{b_4^*})$ in its exponent. Since $\mathbf{A}$ is random, these coefficients are also random in the basis of $\overrightarrow{d_3^*}, \overrightarrow{d_4^*}$. Therefore, $\mathcal{B}$ simulates Game_{ℓ}.

$\mathcal{B}$ returns the above $\mathsf{sk}_{\mathsf{ID}}$ as the response to the query.

In the $\mathsf{Challenge}$ phase, $\mathcal{A}$ sends $\mathcal{B}$ two messages M_0^*, M_1^*, and an identity ID^*. $\mathcal{B}$ chooses a random bit $b' \in \{0,1\}$ and calculates the SF challenge ciphertext:

$$C_0^* = M_{b'}^* \cdot \hat{e}(U_1, g_2^{\overrightarrow{b_1^*}})^{\beta}, \quad C_1^* = U_1^{(\alpha - \mathsf{ID}^*)} U_2.$$

Thus, $\mathcal{B}$ can break SXDH if $\mathcal{A}$ can distinguish $\text{Game}_{\ell-1}$ and Game_{ℓ}.

Proof of Lemma 3. We randomly pick $\zeta_1, \zeta_2 \in \mathbb{Z}_p$ and define new dual orthonormal bases $\mathbb{F} := (\overrightarrow{f_1}, \ldots, \overrightarrow{f_4})$ and $\mathbb{F}^* := (\overrightarrow{f_1^*}, \ldots, \overrightarrow{f_4^*})$ as follows:

$$\begin{bmatrix} \overrightarrow{f_1} \\ \overrightarrow{f_2} \\ \overrightarrow{f_3} \\ \overrightarrow{f_4} \end{bmatrix} := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \zeta_1 & 0 & 1 & 0 \\ 0 & \zeta_2 & 0 & 1 \end{bmatrix} \begin{bmatrix} \overrightarrow{d_1} \\ \overrightarrow{d_2} \\ \overrightarrow{d_3} \\ \overrightarrow{d_4} \end{bmatrix}, \qquad \begin{bmatrix} \overrightarrow{f_1^*} \\ \overrightarrow{f_2^*} \\ \overrightarrow{f_3^*} \\ \overrightarrow{f_4^*} \end{bmatrix} := \begin{bmatrix} 1 & 0 & -\zeta_1 & 0 \\ 0 & 1 & 0 & -\zeta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \overrightarrow{d_1^*} \\ \overrightarrow{d_2^*} \\ \overrightarrow{d_3^*} \\ \overrightarrow{d_4^*} \end{bmatrix}.$$

It is easy to verify that $\mathbb{F}$ and $\mathbb{F}^*$ are also dual orthonormal.

Then, the master public key (apart from public parameters $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$), challenge ciphertext, and queried secret keys in Game_q expressed over $\mathbb{D}$ and $\mathbb{D}^*$ are, respectively:

$$\begin{aligned} \mathsf{mpk} &= (g_1^{\overrightarrow{d_1}}, g_1^{\overrightarrow{d_2}}, g_1^{\alpha\overrightarrow{d_1}}, g_T^{\beta}), \\ C &= (C_0 = M \cdot g_T^{\beta \cdot s}, \quad \overrightarrow{C_1} = g_1^{s(\alpha - \mathsf{ID})\overrightarrow{d_1} + s\overrightarrow{d_2} + \delta_3\overrightarrow{d_3} + \delta_4\overrightarrow{d_4}}), \\ \overrightarrow{K} &= g_2^{(\frac{\beta - r}{\alpha - \mathsf{ID}})\overrightarrow{d_1^*} + r\overrightarrow{d_2^*} + \gamma_3\overrightarrow{d_3^*} + \gamma_4\overrightarrow{d_4^*}}, \end{aligned}$$

which can be expressed over $\mathbb{F}$ and $\mathbb{F}^*$ as:

$$\begin{aligned} \mathsf{mpk} &= (g_1^{\overrightarrow{f_1}}, g_1^{\overrightarrow{f_2}}, g_1^{\alpha\overrightarrow{f_1}}, g_T^{\beta}), \\ C &= (C_0 = M \cdot g_T^{\beta \cdot s}, \quad \overrightarrow{C_1} = g_1^{z\overrightarrow{f_1} + z'\overrightarrow{f_2} + \delta_3\overrightarrow{f_3} + \delta_4\overrightarrow{f_4}}), \\ \overrightarrow{K} &= g_2^{(\frac{\beta - r}{\alpha - \mathsf{ID}})\overrightarrow{f_1^*} + r\overrightarrow{f_2^*} + \gamma_3'\overrightarrow{f_3^*} + \gamma_4'\overrightarrow{f_4^*}}, \end{aligned}$$

where

$$\begin{aligned} z &= s(\alpha - \mathsf{ID}) - \zeta_1\delta_3, & z' &= s - \zeta_2\delta_4, \\ \gamma_3' &= \gamma_3 + (\frac{\beta - r}{\alpha - \mathsf{ID}})\zeta_1, & \gamma_4' &= \gamma_4 + r\zeta_2. \end{aligned}$$

They are uniformly random since $\zeta_1, \zeta_2, \delta_3, \delta_4$ are random. In other words, the coefficients of $(\mathbb{D}, \mathbb{D}^*)$ in the $\overrightarrow{C_1}$ term of the challenge ciphertext are changed to uniformly random coefficients of $(\mathbb{F}, \mathbb{F}^*)$, allowing the challenge ciphertext to be viewed as an SF encryption of a random message. Hence, the adversary cannot distinguish between Game_q and $\text{Game}_{\text{final}}$.

3.2 Accountable-Authority Identity-Based Encryption

In Gentry-IBE, the element r in an identity-based secret key cannot be re-randomized by the user. To extend this into A-IBE scheme, r can be jointly computed by the user and the PKG in such a way that the PKG does not know the final value of r in the secret key it helps issue, while still ensuring that the user cannot re-randomize it. This non-re-randomizability is crucial for tracing — a key is leaked by a user if and only if the embedded element r in the key is identical to that of the user's key [Goy07]. Our A-IBE construction adopts this approach to ensure accountability.

There are several modifications compared to our (prime-order) IBE scheme. Firstly, we turn the key extraction procedure into an interactive protocol between the PKG and user. To defend against dishonest PKG, we divide the secret key into two parts, allowing a simulator acting as a user to issue a key in the semi-functional form without letting the adversary (acting as PKG) know (*cf.*, Theorem 5). Next, we apply a strong one-time signature scheme to achieve CCA security [BCHK07]. To securely bind a verification key of the signature scheme to the ciphertext, we introduce two more vector pairs $(\overrightarrow{d_5}, \overrightarrow{d_5^*})$, $(\overrightarrow{d_6}, \overrightarrow{d_6^*})$ into our system. Note that these pairs do not introduce any additional SF components during the proof. IND-CCA security can be proven in much the same way as for our IBE scheme, except that decryption oracle can now be simulated [BCHK07].

Concrete Construction. We describe our (prime-order) A-IBE scheme as follows.

- $\mathsf{Setup}(1^\lambda)$: The PKG runs the prime-order bilinear group context generator $\mathcal{G}(1^\lambda)$ to get $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e})$ and generators $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$. It samples random dual orthonormal bases $(\mathbb{D}, \mathbb{D}^*) \leftarrow \mathsf{Dual}(\mathbb{Z}_p^6)$. We let $\overrightarrow{d_1}, \ldots, \overrightarrow{d_6}$ denote the elements of $\mathbb{D}$ and $\overrightarrow{d_1^*}, \ldots, \overrightarrow{d_6^*}$ denote the elements of $\mathbb{D}^*$. The PKG randomly picks $\alpha, \beta, v \in \mathbb{Z}_p$, computes $g_T = \hat{e}(g_1, g_2)^{\overrightarrow{d_1} \cdot \overrightarrow{d_1^*}}$. Let $(\mathsf{KG}, \mathsf{Sign}, \mathsf{Verify})$ be a strong one-time signature scheme. Let $(\mathsf{CRSGen}, \mathsf{P}, \mathsf{V})$ be an interactive concurrent zero-knowledge proof of knowledge system [PV08]. It computes

  $$\mathsf{param} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}, g_1^{\overrightarrow{d_1}}, g_1^{\overrightarrow{d_2}}, g_1^{\overrightarrow{d_5}}, g_1^{\overrightarrow{d_6}}, g_T, \mathsf{crs}_1), \quad \mathsf{mpk} = (g_1^{\alpha\overrightarrow{d_1}}, g_1^{v\overrightarrow{d_1}}, g_T^{\beta}),$$

  where $\mathsf{crs}_1 \leftarrow \mathsf{CRSGen}(1^\lambda)$. The master secret key is

  $$\mathsf{msk} = (g_2^{\overrightarrow{d_1^*}}, g_2^{\overrightarrow{d_2^*}}, g_2^{\overrightarrow{d_5^*}}, g_2^{\overrightarrow{d_6^*}}, g_2^{\beta\overrightarrow{d_1^*}}, g_2^{v\overrightarrow{d_2^*}}, \alpha).$$

- $\mathsf{Extract}(\mathsf{msk}, \mathsf{ID})$: The user and the PKG interact to obtain the secret key $\mathsf{sk}_{\mathsf{ID}}$ as

  $$\overrightarrow{K_1} = g_2^{(\frac{\beta - r}{\alpha - \mathsf{ID}})\overrightarrow{d_1^*} + t\overrightarrow{d_5^*} + v \cdot t\overrightarrow{d_6^*}}, \quad \overrightarrow{K_2} = g_2^{r\overrightarrow{d_2^*}}$$

  for random $r, t \in \mathbb{Z}_p$. The interaction is as follows (which implicitly sets $r = r_0 r_1$):

  1. The PKG picks $r_1, t \in \mathbb{Z}_p$, and sends (A_1, A_2) to the user, for $A_1 = g_2^{r_1\overrightarrow{d_1^*}}$, $A_2 = g_2^{r_1\overrightarrow{d_2^*}}$.
  2. The PKG runs the concurrent zero-knowledge proof of knowledge π_1 of r_1 using crs_1, such that A_1 and A_2 are properly formed. The user continues if proof π_1 is valid.
  3. The user chooses some random $r_0, \rho \in \mathbb{Z}_p$.
  4. The user (with committed, private input $(\rho, A_1^{r_0})$) and the PKG (with private input (msk, r_1, t)) engage in a secure two-party computation protocol, efficiently implemented via techniques such as garbled arithmetic circuits [BHR12, AIK14], two-party computation on committed inputs [JS07], or other secure methods for computing such arithmetics [DKL+23, WMC24]. If the committed input is invalid, *e.g.*, $A_1^{r_0}$ is not formed correctly using A_1 with the knowledge of r_0, it outputs $\perp$ and aborts. Otherwise, the PKG obtains the private outputs:

     $$\overrightarrow{K'} = (A_1^{r_0})^{(\frac{-\rho}{\alpha - \mathsf{ID}})} (g_2)^{(\frac{\beta}{\alpha - \mathsf{ID}})\overrightarrow{d_1^*}\rho + t\overrightarrow{d_5^*}\rho + v \cdot t\overrightarrow{d_6^*}\rho}.$$

     The PKG sends $\overrightarrow{K'}$ to the user.
  5. The user outputs $(\overrightarrow{K_1} = \overrightarrow{K'}^{1/\rho}, \overrightarrow{K_2} = A_2^{r_0})$ if it is a valid secret key.

- $\mathsf{Enc}(\mathsf{mpk}, \mathsf{ID}, M)$: To encrypt M to ID, the sender randomly picks $s \in \mathbb{Z}_p$ and runs $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{KG}(1^\lambda)$. It outputs $C = (C_0, C_1, \sigma, \mathsf{vk})$ where

$$C_0 = M \cdot g_T^{\beta \cdot s}, \quad \overrightarrow{C_1} = g_1^{s(\alpha - \mathsf{ID})\overrightarrow{d_1} + s\overrightarrow{d_2} + v \cdot \mathsf{vk}\overrightarrow{d_5}}, \quad \sigma = \mathsf{Sign}(\mathsf{sk}, C_0 || \overrightarrow{C_1}).$$

- $\mathsf{Dec}(\mathsf{mpk}, \mathsf{sk_{ID}}, C)$: Given a ciphertext $C = (C_0, \overrightarrow{C_1}, \sigma, \mathsf{vk})$ and a secret key $\mathsf{sk_{ID}} = (\overrightarrow{K_1}, \overrightarrow{K_2})$, it checks $\mathsf{Verify}(\mathsf{vk}, C_0 || \overrightarrow{C_1}, \sigma)$. If Verify returns that σ is invalid, it outputs $\perp$; else it outputs $M = C_0 / \hat{e}_6(\overrightarrow{C_1} \cdot (g_1^{\overrightarrow{d_6}})^{\mathsf{vk}}, \overrightarrow{K_1} \cdot \overrightarrow{K_2})$.

- $\mathtt{Trace}^{\mathbf{D}}(\mathsf{mpk}, \mathsf{sk_{ID}}, \epsilon)$: Given a valid $\mathsf{sk_{ID}} = (\overrightarrow{K_1}, \overrightarrow{K_2})$ for a user ID and an ϵ-useful decoder box $\mathbf{D}$, it checks by the following steps:
    1. Initialize ctr to 0 and repeat the following steps for $L = 16\lambda/\epsilon$ times:
        (a) Randomly choose $s, s' \in \mathbb{Z}_p$ such that $s \neq s'$ and run $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{KG}(1^\lambda)$. Set $\overrightarrow{C_1} = g_1^{s(\alpha - \mathsf{ID})\overrightarrow{d_1} + s'\overrightarrow{d_2} + v \cdot \mathsf{vk}\overrightarrow{d_5}}$.
        (b) Compute $C_0 = M \cdot \hat{e}_6(\overrightarrow{C_1} \cdot (g_1^{\overrightarrow{d_6}})^{\mathsf{vk}}, \overrightarrow{K_1} \cdot \overrightarrow{K_2})$ for a random message $M \in \mathbb{G}_T$ and $\sigma = \mathsf{Sign}(\mathsf{sk}, C_0 || \overrightarrow{C_1})$.
        (c) Provide $\mathbf{D}$ with $(C_0, \overrightarrow{C_1}, \sigma, \mathsf{vk})$. If $\mathbf{D}$ outputs the same M, increment ctr.
    2. If $\mathsf{ctr} = 0$, incriminate the PKG. Otherwise, incriminate the user.

The security proof is given in Appendix B.

4 Exponent-Inversion Signatures and More

Our dual form IBE scheme implies secure dual form signatures [GLOW12].

4.1 Dual Form Exponent-Inversion Signatures

Our dual form variant of Boneh-Boyen/Gentry signatures, referred to as DFEI, is as follows.

- $\mathsf{Setup}(1^\lambda)$: It runs $\mathcal{G}(1^\lambda)$ to get $(N = p_1 p_2 p_3, \mathbb{G}, \mathbb{G}_T, \hat{e})$ and generators $g_1 \in \mathbb{G}_{p_1}$, $g_3 \in \mathbb{G}_{p_3}$, and $g_{2,3} \in \mathbb{G}_{p_2 p_3}$. It randomly picks $\alpha \in \mathbb{Z}_N$, $u_1, h_1 \in \mathbb{G}_{p_1}$. The public key is $\mathsf{pk} = (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g_1, u_1, \hat{e}(g_1, h_1), g_1^\alpha)$. The secret key is $\mathsf{sk} = (h_1, \alpha, g_3, g_{2,3})$.

- $\mathsf{Sign}(\mathsf{sk}, M)$: To sign on a message $M \in \mathbb{Z}_N$, it randomly picks $X_3, X'_3 \in \mathbb{G}_{p_3}$, $r \in \mathbb{Z}_N$, and outputs $\sigma = (\sigma_1, \sigma_2)$, where $\sigma_1 = (h_1 u_1^{-r})^{\frac{1}{\alpha - M}} X_3$, $\sigma_2 = g_1^r X'_3$.

- $\mathsf{Verify}(\mathsf{pk}, \sigma, M)$: Given $\sigma = (\sigma_1, \sigma_2)$, it outputs $\hat{e}(g_1^\alpha \cdot g_1^{-M}, \sigma_1) \cdot \hat{e}(u_1, \sigma_2) \stackrel{?}{=} \hat{e}(g_1, h_1)$.

The security proof is provided in Appendix C.

Prime-Order Group Version. Like our IBE scheme, we can turn our signature scheme into one over prime-order groups using the methods of Chen *et al.* [CLL+12].

- $\mathsf{Setup}(1^\lambda)$: It runs the bilinear group context generator $\mathcal{G}(1^\lambda)$ to get $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e})$ and generators $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$. It samples random dual orthonormal bases $(\mathbb{D}, \mathbb{D}^*) \leftarrow \mathsf{Dual}(\mathbb{Z}_p^4)$. Let $\overrightarrow{d_1}, \ldots, \overrightarrow{d_4}$ be the elements of $\mathbb{D}$ and $\overrightarrow{d_1^*}, \ldots, \overrightarrow{d_4^*}$ represent the elements of $\mathbb{D}^*$. It randomly picks $\alpha, \gamma, z_1, z_3, y_1, y_3 \in \mathbb{Z}_p$, and let $g_T = \hat{e}(g_1, g_2)^{\overrightarrow{d_1} \cdot \overrightarrow{d_1^*}}$. The public key[5] is

$$\mathsf{pk} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}, g_1, g_2, g_1^{\overrightarrow{d_1}}, g_1^{\overrightarrow{d_2}}, g_1^{\alpha\overrightarrow{d_1}}, g_T^\gamma, Z = g_1^{z_1\overrightarrow{d_1} + z_3\overrightarrow{d_3}}, Y = g_1^{y_1\overrightarrow{d_1} + y_3\overrightarrow{d_3}}).$$

[5] The elements Y and Z are only used for the proof of unforgeability of the anonymous credentials application in Appendix D.

The secret key is $\mathsf{sk} = (g_2^{\overrightarrow{d_1^*}}, g_2^{\overrightarrow{d_2^*}}, \alpha, \gamma)$.

- $\mathsf{Sign}(\mathsf{sk}, M)$: To sign on $M \in \mathbb{Z}_p$, it picks $r \in \mathbb{Z}_p$, computes $\overrightarrow{\sigma_1} = g_2^{(\frac{\gamma-r}{\alpha-M})\overrightarrow{d_1^*}}$, $\overrightarrow{\sigma_2} = g_2^{r\overrightarrow{d_2^*}}$, and outputs $\sigma = (\overrightarrow{\sigma_1}, \overrightarrow{\sigma_2})$.
- $\mathsf{Verify}(\mathsf{pk}, \sigma, M)$: The verifier randomly picks $s \in \mathbb{Z}_p$, sets $\overrightarrow{C} = g_1^{s(\alpha-M)\overrightarrow{d_1}+s\overrightarrow{d_2}}$, and returns $\hat{e}_4(\overrightarrow{C}, \overrightarrow{\sigma_1} \circ \overrightarrow{\sigma_2}) \overset{?}{=} g_T^{\gamma s}$.

Optimized Version. The signature in our scheme includes two components designed for use in our anonymous credential system. We can make slight modifications to our algorithms to reduce the signature size:

$\mathsf{Sign}(\mathsf{sk}, M)$: It computes the signature as $\overrightarrow{\sigma} = g_2^{(\frac{\gamma-r}{\alpha-M})\overrightarrow{d_1^*}+r\overrightarrow{d_2^*}}$.

$\mathsf{Verify}(\mathsf{pk}, \sigma, M)$: The recipient returns $\hat{e}_4(\overrightarrow{C}, \overrightarrow{\sigma}) \overset{?}{=} g_T^{\gamma s}$.

4.2 Anonymous Credentials from P-Signatures

Anonymous credentials allow a prover to demonstrate to a verifier that the prover possesses a certificate from a credential issuer, while remaining unlinkable to both the registration instance with the issuer for obtaining the certificate and any such demonstrations with other verifiers. A formal definition has been detailed [BCKL08]. A P-signature scheme is a type of signature scheme with efficient protocols for non-interactive zero-knowledge proof of knowledge, both for requesting and showing signatures on a committed message. Belenkiy *et al.* [BCKL08] show that anonymous credentials are an immediate consequence of P-signatures. Thus, by presenting a P-signature scheme under the SXDH assumption, we obtain a non-interactive anonymous credential scheme.

We use the prime-order version of DFEI, along with the Groth-Sahai NIZK proof system $\mathsf{GS} = (\mathsf{Setup}, \mathsf{Commit}, \mathsf{Prove}, \mathsf{Verify})$, instantiated under the SXDH assumption [GS12]. Our P-signature scheme is presented as follows:

- $\mathsf{Setup}(1^\lambda)$: It runs $\mathcal{G}(1^\lambda)$ to obtain the parameters $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e})$ for λ-bit prime p. Using the same bilinear group parameters, it runs $\mathsf{param}_{\mathsf{GS}} \leftarrow \mathsf{GS.Setup}(1^\lambda)$ and picks a $\mathbb{G}_1$-generator Y and a $\mathbb{G}_2$-generator Z. Finally, it outputs the public system parameters $\mathsf{param} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}, Y, Z, \mathsf{param}_{\mathsf{GS}})$.
- $\mathsf{Sign}(\mathsf{param}, \mathsf{sk}, M)$: It returns $\sigma \leftarrow \mathsf{DFEI.Sign}(\mathsf{sk}, M)$.
- $\mathsf{Verify}(\mathsf{param}, \mathsf{pk}, \sigma, M)$: It returns $\{0, 1\} \leftarrow \mathsf{DFEI.Verify}(\mathsf{pk}, \sigma, M)$.
- $\mathsf{Commit}(\mathsf{param}, M, \mathsf{Open})$: It returns $C \leftarrow \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Z^M, \mathsf{Open})$.
- $\mathsf{ObtainSig}(\mathsf{param}, \mathsf{pk}, M, C, \mathsf{Open}) \leftrightarrow \mathsf{IssueSig}(\mathsf{param}, \mathsf{sk}, C)$:
  1. The user chooses random $\rho_1, \rho_2 \in \mathbb{Z}_p$.
  2. The issuer chooses a random $r \in \mathbb{Z}_p$.
  3. The user (with private input $(M, \rho_1, \rho_2, \mathsf{Open})$) and the issuer (with private input (sk, r)) engage in a secure two-party computation protocol to compute the private outputs below, using techniques such as those suggested in Section 3.2 (*e.g.*, by securely computing $(\alpha - M)\rho_1$). If all private inputs are honestly committed, in particular, $C = \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Z^M, \mathsf{Open})$, the issuer uses $\mathsf{sk} = (g_2^{\overrightarrow{d_1^*}}, g_2^{\overrightarrow{d_2^*}}, \alpha, \gamma)$ and obtains private outputs:

$$\overrightarrow{\sigma_1'} = g_2^{(\frac{\gamma-r}{(\alpha-M)\rho_1})\overrightarrow{d_1^*}}, \quad \overrightarrow{\sigma_2'} = g_2^{r\overrightarrow{d_2^*}\rho_2}.$$

Otherwise, the issuer aborts. Finally, the issuer sends $(\overrightarrow{\sigma_1'}, \overrightarrow{\sigma_2'})$ to the user.

4. The user outputs $(\overrightarrow{\sigma_1} = \overrightarrow{\sigma_1'}^{\rho_1}, \overrightarrow{\sigma_2} = \overrightarrow{\sigma_2'}^{1/\rho_2})$ if it is a valid signature.

- $\mathsf{Prove}(\mathsf{param}, \mathsf{pk}, M, \sigma)$: It forms the following GS commitments:

  $$M_Z = \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Z^M, \mathsf{Open}_Z), M_Y = \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Y^M, \mathsf{Open}_Y),$$
  $$\Sigma_{i,j} = \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, \sigma_{i,j}, \mathsf{Open}_{\Sigma_{i,j}}) \text{ for } i \in [1,2], j \in [1,4],$$

  where $\sigma_{i,j}$ is the j-th component of $\overrightarrow{\sigma_i}$, and all openings Open_Z, Open_Y, and $\{\mathsf{Open}_{\Sigma_{i,j}}\}$ are freshly generated. It then computes the proof π as follows:

  $$\mathsf{GS.Prove}\{(M, \sigma) : \hat{e}_4(g_1^{s(\alpha - M)\overrightarrow{d_1} + s\overrightarrow{d_2}}, \overrightarrow{\sigma_1} \circ \overrightarrow{\sigma_2}) = \hat{e}(g_1, g_2)^{\gamma \overrightarrow{d_1} \cdot \overrightarrow{d_1^*} s}\},$$

  using the commitments M_Z, M_Y, and $\overrightarrow{\Sigma}_i = (\Sigma_{i,1}, \ldots, \Sigma_{i,4})$ for some randomly chosen $s \in \mathbb{Z}_p$. It outputs $\mathsf{comm} = (M_Z, M_Y, \overrightarrow{\Sigma}_1, \overrightarrow{\Sigma}_2)$ and π.

- $\mathsf{VerifyPf}(\mathsf{param}, \mathsf{pk}, \mathsf{comm}, \pi)$: It outputs 1 if π is a valid NIZK proof for comm and the language above; 0 otherwise.

- $\mathsf{EqCommProve}(\mathsf{param}, M, \mathsf{Open}, \mathsf{Open}')$: It forms the GS commitments below:

  $$\mathsf{comm}_1 = \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Z^M, \mathsf{Open}),$$
  $$\mathsf{comm}_2 = \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Z^M, \mathsf{Open}'),$$

  and compute the following proof:

  $$\pi = \mathsf{GS.Prove}\{(M, \mathsf{Open}, \mathsf{Open}') : \ M_1 = \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Z^M, \mathsf{Open})$$
  $$\wedge\ M_2 = \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Z^M, \mathsf{Open}')\},$$

  which can be done by the existing zero-knowledge proof of equality of committed exponents [BCKL08]. The final outputs contain $\mathsf{comm}_1, \mathsf{comm}_2$, and π.

- $\mathsf{EqCommVerify}(\mathsf{param}, \mathsf{Open}, \mathsf{Open}', \pi)$: It outputs 1 if π is a valid NIZK proof for comm_1 and comm_2 above; 0 otherwise.

Theorem 3. *Our P-signature scheme is secure under the SXDH assumption and the security of the two-party computation.*

The proof is provided in Appendix D.

5 Conclusion

We present dual form exponent-inversion signature schemes in the standard model under static assumptions. By extending them via P-signatures, we obtain an anonymous credential scheme that is more efficient than upgrading previous "simple-assumption-relying" works. We also provide dual system encryption variants of Gentry-IBE [Gen06] under static assumptions, from which we build a fully secure black-box accountable-authority IBE scheme with constant key and ciphertext sizes.

Our work offers an orthogonal approach to the Déjà Q frameworks [CM14, CMM16]. While their focus is on assumption-level transformations, our study emphasizes the scheme level, covering a representative class of schemes built upon exponent inversion.

A conceptual contribution of our study is the insights it provides into transforming exponent-inversion-based schemes to rely on static assumptions. This exploration also raises the question of whether the distinction between the exponent-inversion and commutative blinding families is as significant as perceived, suggesting avenues for further research.

References

[ABB10] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In *CRYPTO*, pages 98–115, 2010. doi:10.1007/978-3-642-14623-7_6.

[ACD$^+$16] Masayuki Abe, Melissa Chase, Bernardo David, Markulf Kohlweiss, Ryo Nishimaki, and Miyako Ohkubo. Constant-size structure-preserving signatures: Generic constructions and simple assumptions. *J. Cryptol.*, 29(4):833–878, 2016. doi:10.1007/S00145-015-9211-7.

[ACN13] Tolga Acar, Sherman S. M. Chow, and Lan Nguyen. Accumulators and U-Prove revocation. In *FC*, pages 189–196, 2013. doi:10.1007/978-3-642-39884-1_15.

[AFG$^+$10] Masayuki Abe, Georg Fuchsbauer, Jens Groth, Kristiyan Haralambiev, and Miyako Ohkubo. Structure-preserving signatures and commitments to group elements. In *CRYPTO*, pages 209–236, 2010. doi:10.1007/978-3-642-14623-7_12.

[AIK14] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. How to garble arithmetic circuits. *SIAM J. Comput.*, 43(2):905–929, 2014. doi:10.1137/120875193.

[ASMC13] Man Ho Au, Willy Susilo, Yi Mu, and Sherman S. M. Chow. Constant-size dynamic k-times anonymous authentication. *IEEE Syst. J.*, 7(2):249–261, 2013. doi:10.1109/JSYST.2012.2221931.

[BB08] Dan Boneh and Xavier Boyen. Short signatures without random oracles and the SDH assumption in bilinear groups. *J. Cryptol.*, 21(2):149–177, 2008. doi:10.1007/S00145-007-9005-7.

[BBS04] Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In *CRYPTO*, pages 41–55, 2004. doi:10.1007/978-3-540-28628-8_3.

[BCHK07] Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. *SIAM J. Comput.*, 36(5):1301–1328, 2007. doi:10.1137/S009753970544713X.

[BCKL08] Mira Belenkiy, Melissa Chase, Markulf Kohlweiss, and Anna Lysyanskaya. P-signatures and noninteractive anonymous credentials. In *TCC*, pages 356–374, 2008. doi:10.1007/978-3-540-78524-8_20.

[BF03] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil pairing. *SIAM J. Comput.*, 32(3):586–615, 2003. doi:10.1137/S0097539701398521.

[BHR12] Mihir Bellare, Viet Tung Hoang, and Phillip Rogaway. Foundations of garbled circuits. In *CCS*, pages 784–796, 2012. doi:10.1145/2382196.2382279.

[BLSV18] Zvika Brakerski, Alex Lombardi, Gil Segev, and Vinod Vaikuntanathan. Anonymous IBE, leakage resilience and circular security from new assumptions. In *EUROCRYPT Part I*, pages 535–564, 2018. doi:10.1007/978-3-319-78381-9_20.

[Boy07] Xavier Boyen. General ad hoc encryption from exponent inversion IBE. In *EUROCRYPT*, pages 394–411, 2007. doi:10.1007/978-3-540-72540-4_23.

[CCH+12] Sherman S. M. Chow, Cheng-Kang Chu, Xinyi Huang, Jianying Zhou, and Robert H. Deng. Dynamic secure cloud storage with provenance. In *Cryptography and Security: From Theory to Applications - Essays Dedicated to Jean-Jacques Quisquater on the Occasion of His 65th Birthday*, pages 442–464, 2012. doi:10.1007/978-3-642-28368-0_28.

[CDRW10] Sherman S. M. Chow, Yevgeniy Dodis, Yannis Rouselakis, and Brent Waters. Practical leakage-resilient identity-based encryption from simple assumptions. In *CCS*, pages 152–161, 2010. doi:10.1145/1866307.1866325.

[CHKP12] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. *J. Cryptol.*, 25(4):601–639, 2012. doi:10.1007/S00145-011-9105-2.

[Cho09] Sherman S. M. Chow. Removing escrow from identity-based encryption. In *PKC*, pages 256–276, 2009. doi:10.1007/978-3-642-00468-1_15.

[Cho10] Sherman S. M. Chow. *New Privacy-Preserving Architectures for Identity-/Attribute-based Encryption.* PhD thesis, New York University, USA, 2010. URL: https://dl.acm.org/doi/10.5555/2049343.

[CK18] Sanjit Chatterjee and R. Kabaleeshwaran. Towards static assumption based cryptosystem in pairing setting: Further applications of DéjàQ and dual-form signature (extended abstract). In *PROVSEC*, pages 220–238, 2018. doi:10.1007/978-3-030-01446-9_13.

[CK19] Sanjit Chatterjee and R. Kabaleeshwaran. Rerandomizable signatures under standard assumption. In *INDOCRYPT*, pages 45–67, 2019. doi:10.1007/978-3-030-35423-7_3.

[CL04] Jan Camenisch and Anna Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In *CRYPTO*, pages 56–72, 2004. doi:10.1007/978-3-540-28628-8_4.

[CLL+12] Jie Chen, Hoon Wei Lim, San Ling, Huaxiong Wang, and Hoeteck Wee. Shorter IBE and signatures via asymmetric pairings. In *PAIRING*, pages 122–140, 2012. doi:10.1007/978-3-642-36334-4_8.

[CM14] Melissa Chase and Sarah Meiklejohn. Déjà Q: using dual systems to revisit q-type assumptions. In *EUROCRYPT*, pages 622–639, 2014. doi:10.1007/978-3-642-55220-5_34.

[CMM16] Melissa Chase, Mary Maller, and Sarah Meiklejohn. Déjà Q all over again: Tighter and broader reductions of q-type assumptions. In *ASIACRYPT Part II*, pages 655–681, 2016. doi:10.1007/978-3-662-53890-6_22.

[CY11] Sherman S. M. Chow and Siu-Ming Yiu. Exclusion-intersection encryption. *Int. J. Secur. Networks*, 6(2/3):136–146, 2011. doi:10.1504/IJSN.2011.043672.

[CZZ17] Sherman S. M. Chow, Haibin Zhang, and Tao Zhang. Real hidden identity-based signatures. In *FC*, pages 21–38, 2017. doi:10.1007/978-3-319-70972-7_2.

[Del07] Cécile Delerablée. Identity-based broadcast encryption with constant size ciphertexts and private keys. In *ASIACRYPT*, pages 200–215, 2007. doi:10.1007/978-3-540-76900-2_12.

[DG21] Nico Döttling and Sanjam Garg. Identity-based encryption from the Diffie-Hellman assumption. *J. ACM*, 68(3):14:1–14:46, 2021. doi:10.1145/3422370.

[DKL$^+$23] Jack Doerner, Yashvanth Kondi, Eysa Lee, abhi shelat, and LaKyah Tyner. Threshold BBS+ signatures for distributed anonymous credential issuance. In *SP*, pages 773–789, 2023. doi:10.1109/SP46215.2023.10179470.

[DP08] Cécile Delerablée and David Pointcheval. Dynamic threshold public-key encryption. In *CRYPTO*, pages 317–334, 2008. doi:10.1007/978-3-540-85174-5_18.

[DY05] Yevgeniy Dodis and Aleksandr Yampolskiy. A verifiable random function with short proofs and keys. In *PKC*, pages 416–431, 2005. doi:10.1007/978-3-540-30580-4_28.

[Gen06] Craig Gentry. Practical identity-based encryption without random oracles. In *EUROCRYPT*, pages 445–464, 2006. doi:10.1007/11761679_27.

[GH09] Craig Gentry and Shai Halevi. Hierarchical identity based encryption with polynomially many levels. In *TCC*, pages 437–456, 2009. doi:10.1007/978-3-642-00457-5_26.

[GLOW12] Michael Gerbush, Allison B. Lewko, Adam O'Neill, and Brent Waters. Dual form signatures: An approach for proving security from static assumptions. In *ASIACRYPT*, pages 25–42, 2012. doi:10.1007/978-3-642-34961-4_4.

[Goy07] Vipul Goyal. Reducing trust in the PKG in identity based cryptosystems. In *CRYPTO*, pages 430–447, 2007. doi:10.1007/978-3-540-74143-5_24.

[Gro07] Jens Groth. Fully anonymous group signatures without random oracles. In *ASIACRYPT*, pages 164–180, 2007. doi:10.1007/978-3-540-76900-2_10.

[GS12] Jens Groth and Amit Sahai. Efficient noninteractive proof systems for bilinear groups. *SIAM J. Comput.*, 41(5):1193–1232, 2012. doi:10.1137/080725386.

[JS07] Stanislaw Jarecki and Vitaly Shmatikov. Efficient two-party secure computation on committed inputs. In *EUROCRYPT*, pages 97–114, 2007. doi:10.1007/978-3-540-72540-4_6.

[JY09] David Jao and Kayo Yoshida. Boneh-Boyen signatures and the strong Diffie-Hellman problem. In *PAIRING*, pages 1–16, 2009. doi:10.1007/978-3-642-03298-1_1.

[KT15] Aggelos Kiayias and Qiang Tang. Making any identity-based encryption accountable, efficiently. In *ESORICS Part I*, pages 326–346, 2015. doi:10.1007/978-3-319-24174-6_17.

[LDZW13] Junzuo Lai, Robert H. Deng, Yunlei Zhao, and Jian Weng. Accountable authority identity-based encryption with public traceability. In *CTRSA*, pages 326–342, 2013. doi:10.1007/978-3-642-36095-4_21.

[LMPY16] Benoît Libert, Fabrice Mouhartem, Thomas Peters, and Moti Yung. Practical "signatures with efficient protocols" from simple assumptions. In *AsiaCCS*, pages 511–522, 2016. doi:10.1145/2897845.2897898.

[LV11] Benoît Libert and Damien Vergnaud. Towards practical black-box accountable authority IBE: Weak black-box traceability with short ciphertexts and private keys. *IEEE Trans. Inf. Theory*, 57(10):7189–7204, 2011. doi:10.1109/TIT.2011.2161958.

[LW10] Allison B. Lewko and Brent Waters. New techniques for dual system encryption and fully secure HIBE with short ciphertexts. In *TCC*, pages 455–479, 2010. doi:10.1007/978-3-642-11799-2_27.

[Ngu05] Lan Nguyen. Accumulators from bilinear pairings and applications. In *CTRSA*, pages 275–292, 2005. doi:10.1007/978-3-540-30574-3_19.

[OT08] Tatsuaki Okamoto and Katsuyuki Takashima. Homomorphic encryption and signatures from vector decomposition. In *PAIRING*, pages 57–74, 2008. doi:10.1007/978-3-540-85538-5_4.

[PS16] David Pointcheval and Olivier Sanders. Short randomizable signatures. In *CTRSA*, pages 111–126, 2016. doi:10.1007/978-3-319-29485-8_7.

[PS18] David Pointcheval and Olivier Sanders. Reassessing security of randomizable signatures. In *CTRSA*, pages 319–338, 2018. doi:10.1007/978-3-319-76953-0_17.

[PV08] Rafael Pass and Muthuramakrishnan Venkitasubramaniam. On constant-round concurrent zero-knowledge. In *TCC*, pages 553–570, 2008. doi:10.1007/978-3-540-78524-8_30.

[SK03] Ryuichi Sakai and Masao Kasahara. Cryptosystems based on pairing over elliptic curve. In *Symposium on Cryptography and Information Security*, 2003.

[SS11] Amit Sahai and Hakan Seyalioglu. Fully secure accountable-authority identity-based encryption. In *PKC*, pages 296–316, 2011. doi:10.1007/978-3-642-19379-8_19.

[Wat09] Brent Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In *CRYPTO*, pages 619–636, 2009. doi:10.1007/978-3-642-03356-8_36.

[WC23] Huangting Wu and Sherman S. M. Chow. Anonymous (hierarchical) identity-based encryption from broader assumptions. In *ACNS Part II*, pages 366–395, 2023. doi:10.1007/978-3-031-33491-7_14.

[Wee16] Hoeteck Wee. Déjà Q: encore! un petit IBE. In *TCC-A Part II*, pages 237–258, 2016. doi:10.1007/978-3-662-49099-0_9.

[WMC24] Harry W. H. Wong, Jack P. K. Ma, and Sherman S. M. Chow. Secure multiparty computation of threshold signatures made more efficient. In *NDSS*, 2024. doi:10.14722/ndss.2024.24601.

[YCZY14] Tsz Hon Yuen, Sherman S. M. Chow, Cong Zhang, and Siu-Ming Yiu. Exponent-inversion signatures and IBE under static assumptions. IACR Cryptol. ePrint Arch. 2014/311, 2014.

[YZC22] Tsz Hon Yuen, Cong Zhang, and Sherman S. M. Chow. Don't tamper with dual system encryption - beyond polynomial related-key security of IBE. In *ACNS*, pages 419–439, 2022. doi:10.1007/978-3-031-09234-3_21.

[YZCL13] Tsz Hon Yuen, Cong Zhang, Sherman S. M. Chow, and Joseph K. Liu. Towards anonymous ciphertext indistinguishability with identity leakage. In *PROVSEC*, pages 139–153, 2013. doi:10.1007/978-3-642-41227-1_8.

[ZSS04] Fangguo Zhang, Reihaneh Safavi-Naini, and Willy Susilo. An efficient signature scheme from bilinear pairings and its applications. In *PKC*, pages 277–290, 2004. doi:10.1007/978-3-540-24632-9_20.

A Security Proof for Our Identity-Based Encryption

A.1 Complexity Assumptions

We first review three complexity assumptions [LW10].

Assumption 1. Given a bilinear group context generator $\mathcal{G}$, which picks λ-bit primes p_1, p_2 and p_3 and outputs $(\mathbb{G}, \mathbb{G}_T)$ of order $N = p_1p_2p_3$, we define the distribution below:

$$(N = p_1p_2p_3, \mathbb{G}, \mathbb{G}_T, \hat{e}) \overset{R}{\leftarrow} \mathcal{G}(1^\lambda), \quad g \overset{R}{\leftarrow} \mathbb{G}_{p_1}, \quad X_3 \overset{R}{\leftarrow} \mathbb{G}_{p_3},$$
$$T_0 \overset{R}{\leftarrow} \mathbb{G}_{p_1p_2}, \quad T_1 \overset{R}{\leftarrow} \mathbb{G}_{p_1}, \quad D := (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g, X_3).$$

For any PPT algorithm $\mathcal{A}_1$ with output in $\{0, 1\}$, the advantage

$$\mathsf{Adv}_{\mathcal{G},\mathcal{A}_1} := |\Pr[(D, T_0) = 1] - \Pr[(D, T_1) = 1]| = \mathsf{negl}(\lambda),$$

where $\mathsf{negl}(\lambda)$ denotes the class of negligible function in λ.

Assumption 2. Given a bilinear group context generator $\mathcal{G}$, we define the distribution:

$$(N, \mathbb{G}, \mathbb{G}_T, \hat{e}) \overset{R}{\leftarrow} \mathcal{G}(1^\lambda), \quad g, X_1, Z_1 \overset{R}{\leftarrow} \mathbb{G}_{p_1}, \quad X_i, Y_i, Z_i \overset{R}{\leftarrow} \mathbb{G}_{p_i}(i = 2, 3),$$
$$T_0 = Z_1Z_3, \quad T_1 = Z_1Z_2Z_3, \quad D := (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g, X_1X_2, X_3, Y_2Y_3).$$

For any PPT algorithm $\mathcal{A}_2$ with output in $\{0, 1\}$, the advantage

$$\mathsf{Adv}_{\mathcal{G},\mathcal{A}_2} := |\Pr[(D, T_0) = 1] - \Pr[(D, T_1) = 1]| = \mathsf{negl}(\lambda).$$

Assumption 3. Given $\mathcal{G}$, we define the following distribution:

$$(N, \mathbb{G}, \mathbb{G}_T, \hat{e}) \overset{R}{\leftarrow} \mathcal{G}(1^\lambda), \alpha, s \overset{R}{\leftarrow} \mathbb{Z}_N, g \overset{R}{\leftarrow} \mathbb{G}_{p_1}, X_2, Y_2, Z_2 \overset{R}{\leftarrow} \mathbb{G}_{p_2}, X_3 \overset{R}{\leftarrow} \mathbb{G}_{p_3},$$
$$T_0 := \hat{e}(g, g)^{\alpha s}, \quad T_1 \overset{R}{\leftarrow} \mathbb{G}_T. \quad D := (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g, g^\alpha X_2, g^s Y_2, Z_2, X_3).$$

For any PPT algorithm $\mathcal{A}_3$ with output in $\{0, 1\}$, the advantage

$$\mathsf{Adv}_{\mathcal{G},\mathcal{A}_3} := |\Pr[(D, T_0) = 1] - \Pr[(D, T_1) = 1]| = \mathsf{negl}(\lambda).$$

A.2 Security Proof

Under the dual system encryption paradigm [Wat09], we define the semi-functional (SF) structures used only in the security proof. These SF structures are like their normal version in the actual scheme but "perturbed" by a $\mathbb{G}_{p_2}$ generator, denoted by either $\bar{g}_2$ or $\hat{g}_2$ below.

An *SF secret key* (or simply *SF key*) takes the form of $(K_1' = K_1 \cdot \bar{g}_2^\gamma, K_2' = K_2 \cdot \bar{g}_2)$, where $\gamma \in \mathbb{Z}_N$, and (K_1, K_2) is a normal secret key.

An *SF ciphertext* is in the form of $(C_0' = C_0, C_1' = C_1 \cdot \hat{g}_2, C_2' = C_2 \cdot \hat{g}_2^\delta)$, where $\delta \in \mathbb{Z}_N$ and (C_0, C_1, C_2) is a normal ciphertext.

Decryption involving SF structures succeeds if an SF key is used to decrypt a normal ciphertext, or a normal key is used to decrypt an SF ciphertext. However, decrypting an SF ciphertext using an SF secret key will result in a message "blinded" by $\hat{e}(\bar{g}_2, \hat{g}_2)^{-(\gamma+\delta)}$.

In case the exponents in these extra blinding factors are zeros, decryption still works, leading to the notion of *nominally semi-functional (NSF) secret keys.* An NSF secret key is a special kind of SF key that can successfully decrypt the corresponding SF ciphertext, namely, $\gamma + \delta = 0$. If an SF secret key is not nominally semi-functional, it is *truly semi-functional.*

Theorem 1. *Our IBE scheme is IND-ID-CPA secure under Assumptions 1, 2, and 3.*

Proof. We prove this through a hybrid argument involving a sequence of games. The first game $\mathrm{Game}_{\mathrm{real}}$ is the IND-ID-CPA game. Let the challenge identity be ID^*. The second game $\mathrm{Game}_{\mathrm{res}}$ is the same as $\mathrm{Game}_{\mathrm{real}}$, except that the adversary cannot ask for the secret key of identity $\mathsf{ID} = \mathsf{ID}^* \bmod p_2$. Subsequent games will retain this restriction. Let q be the number of extraction oracle queries. For $k \in \{0, \ldots, q\}$, we define Game_k as follows.

Game$_k$: It is the same as $\mathrm{Game}_{\mathrm{res}}$, except that the challenge ciphertext is semi-functional and the keys used to answer first k^{th} oracle queries are semi-functional. The keys for the rest of the queries are normal. Thus, in Game_0, all keys are normal, and the challenge ciphertext is semi-functional. In Game_q, everything is semi-functional.

The last game is $\mathrm{Game}_{\mathrm{final}}$, identical to Game_q, except that the challenge ciphertext is a semi-functional encryption of a random message instead of a challenge message.

We proceed by proving the indistinguishability between these games.

Lemma 4. *We can construct an algorithm $\mathcal{B}$ with a non-negligible advantage in breaking Assumption 1 or 2 given $\mathcal{A}$ such that $\mathsf{Adv}_{\mathcal{A}}(Game_{real}) - \mathsf{Adv}_{\mathcal{A}}(Game_{res})$ is non-negligible*

The proof of Lemma 4 is easy and similar to the one in [LW10] and hence omitted.

Lemma 5. *We can construct an algorithm $\mathcal{B}$ with advantage ϵ in breaking Assumption 1 if there exists an adversary $\mathcal{A}$ such that $\mathsf{Adv}_{\mathcal{A}}(Game_{res}) - \mathsf{Adv}_{\mathcal{A}}(Game_0) = \epsilon$.*

Lemma 6. *We can construct an algorithm $\mathcal{B}$ with advantage ϵ in breaking Assumption 2 if there exists $\mathcal{A}$ such that $\mathsf{Adv}_{\mathcal{A}}(Game_{\ell-1}) - \mathsf{Adv}_{\mathcal{A}}(Game_\ell) = \epsilon$.*

Lemma 7. *We can construct an algorithm $\mathcal{B}$ with advantage ϵ in breaking Assumption 3 if there exists $\mathcal{A}$ such that $\mathsf{Adv}_{\mathcal{A}}(Game_q) - \mathsf{Adv}_{\mathcal{A}}(Game_{final}) = \epsilon$.*

In $\mathrm{Game}_{\mathrm{final}}$, the value of b' is information-theoretically hidden from $\mathcal{A}$. Thus, $\mathcal{A}$ has no advantage in winning $\mathrm{Game}_{\mathrm{final}}$. If Assumptions 1, 2, and 3 hold, given the proofs of the above lemmas, $\mathrm{Game}_{\mathrm{real}}$ is indistinguishable from $\mathrm{Game}_{\mathrm{final}}$. Hence, the attacker has a negligible advantage in winning $\mathrm{Game}_{\mathrm{real}}$. So, our scheme is IND-ID-CPA secure. □

Proof of Lemma 5. Given (g, X_3, T) from Assumption 1, $\mathcal{B}$ can simulate $\mathrm{Game}_{\mathrm{res}}$ or Game_0 with $\mathcal{A}$. $\mathcal{B}$ chooses random $\beta \in \mathbb{Z}_N$ and $h_1 \in \mathbb{G}_{p_1}$; sets $g_1 = g$, $u_1 = g^\beta$, $g_3 = X_3$; and generates the rest of mpk and $\mathsf{msk} = (\alpha, h_1, g_3)$ according to Setup. For extraction oracle queries, $\mathcal{B}$ answers by computing $\mathsf{sk}_{\mathsf{ID}}$ using msk.

When $\mathcal{A}$ sends $\mathcal{B}$ two messages M_0^*, M_1^*, and an identity ID^*, $\mathcal{B}$ randomly picks a bit $b' \in \{0, 1\}$. $\mathcal{B}$ computes the challenge ciphertext as:

$$C_0^* = M_{b'}^* \cdot \hat{e}(T, h_1), \quad C_1^* = T^{\alpha - \mathsf{ID}^*}, \quad C_2^* = T^\beta.$$

If $T = g^s$, this is a normal ciphertext and hence $\mathcal{B}$ simulates $\mathrm{Game}_{\mathrm{res}}$. If $T = g^s Y_2$, this is an SF ciphertext with $\hat{g}_2 = Y_2^{\alpha - \mathsf{ID}^*}, \hat{g}_2^\delta = Y_2^\beta$; and hence $\mathcal{B}$ simulates Game_0 with $\delta = \beta/(\alpha - \mathsf{ID}^*)$. By the Chinese remainder theorem, the values of $\alpha, \beta \bmod p_2$ are not correlated with the values of α and β modulo p_1. If $\mathcal{A}$ can distinguish between $\mathrm{Game}_{\mathrm{res}}$ and Game_0, $\mathcal{B}$ can then break Assumption 1.

Proof of Lemma 6. Given $(g, X_1X_2, X_3, Y_2Y_3, T)$ from Assumption 2, $\mathcal{B}$ can simulate $\mathrm{Game}_{\ell-1}$ or Game_ℓ with $\mathcal{A}$. $\mathcal{B}$ picks random $\alpha, \beta \in \mathbb{Z}_N$, $h_1 \in \mathbb{G}_{p_1}$, sets $g_1 = g$, $u_1 = g^\beta$, and $g_3 = X_3$; and generates the rest of mpk and $\mathsf{msk} = (\alpha, h_1, g_3)$ as Setup. For the k^{th} distinct extraction oracle query on ID_k:

- If $k < \ell$, $\mathcal{B}$ computes the normal key $\mathsf{sk}_{\mathsf{ID}_k}$ using msk.

- If $k > \ell$, $\mathcal{B}$ computes the normal key $\mathsf{sk}_{\mathsf{ID}_k} = (K_1, K_2)$ using msk. $\mathcal{B}$ randomly picks $\gamma_1, \gamma_2 \in \mathbb{Z}_N$ and returns the SF key:

  $$K_1' = K_1 \cdot (Y_2 Y_3)^{\gamma_1}, \quad K_2' = K_2 \cdot (Y_2 Y_3)^{\gamma_2}.$$

  This is a semi-functional key. By the Chinese remainder theorem, the values of γ_1, γ_2 modulo p_2 and those modulo p_3 are not correlated.

- If $k = \ell$, $\mathcal{B}$ chooses random $X_3', X_3'' \in \mathbb{G}_{p_3}$ and computes the key $\mathsf{sk}_{\mathsf{ID}_\ell}$:

  $$K_1 = h_1^{\frac{1}{\alpha - \mathsf{ID}_\ell}} \cdot T^{\frac{\beta}{\alpha - \mathsf{ID}_\ell}} \cdot X_3', \quad K_2 = T \cdot X_3''.$$

  If $T = Z_1 Z_3$, it is a normal key (for $Z_1 = g^r$). Hence, $\mathcal{B}$ simulates Game$_{\ell-1}$. If $T = Z_1 Z_2 Z_3$, it is an SF key with $\bar{g}_2^{\gamma} = Z_2^{\frac{\beta}{\alpha - \mathsf{ID}_\ell}}$ and $\bar{g}_2 = Z_2$. Thus, $\mathcal{B}$ simulates Game$_\ell$. The value of γ mod p_2 is not correlated with the values of α and β modulo p_1.

In the $\mathsf{Challenge}$ phase, $\mathcal{A}$ sends $\mathcal{B}$ two messages M_0^*, M_1^*, and an identity ID^*. $\mathcal{B}$ chooses a random bit $b' \in \{0, 1\}$ and computes the challenge ciphertext:

$$C_0^* = M_{b'}^* \cdot \hat{e}(X_1 X_2, h_1), \quad C_1^* = (X_1 X_2)^{\alpha - \mathsf{ID}^*}, \quad C_2^* = (X_1 X_2)^{\beta}.$$

It is an SF ciphertext with $\hat{g}_2 = X_2^{\alpha - \mathsf{ID}^*}$ and $\hat{g}_2^{\delta} = X_2^{\beta}$. Note that $f(\mathsf{ID}) = \beta/(\alpha - \mathsf{ID})$ is a pairwise independent function modulo p_2. As long as $\mathsf{ID}^* \neq \mathsf{ID}_\ell$, δ and $\gamma = \beta/(\alpha - \mathsf{ID}_\ell)$ mod p_2 will seem randomly distributed to $\mathcal{A}$ (again, the values of α and β modulo p_2 are uncorrelated with their values modulo p_1 by the Chinese remainder theorem). The case that $\mathsf{ID}^* = \mathsf{ID}_\ell$ mod p_2 in which $\mathcal{A}$ has made an invalid key request has been excluded.

So $\mathcal{B}$ can break Assumption 2 if $\mathcal{A}$ can distinguish Game$_{\ell-1}$ and Game$_\ell$.

Proof of Lemma 7. Given $(g, g^a X_2, g^s Y_2, Z_2, X_3, T)$ from Assumption 3, $\mathcal{B}$ chooses random $\alpha, \beta \in \mathbb{Z}_N$ and sets $g_1 = g, u_1 = g^{\beta}, \hat{e}(g_1, h_1) = \hat{e}(g, g^a X_2)$. $\mathcal{B}$ implicitly sets $h_1 = g^a$. $\mathcal{B}$ generates the rest of the master public key mpk honestly and sends mpk to $\mathcal{A}$.

$\mathcal{B}$ can compute the semi-functional secret key for ID as follows. $\mathcal{B}$ randomly picks $r \in \mathbb{Z}_N$, $R_2, R_2' \in \mathbb{G}_{p_3}$and $R_3, R_3' \in \mathbb{G}_{p_3}$ and computes:

$$K_1' = (g^a X_2 \cdot u_1^{-r})^{\frac{1}{\alpha - \mathsf{ID}}} \cdot R_2 \cdot R_3, \quad K_2' = g^r \cdot R_2' \cdot R_3',$$

Therefore $\mathcal{B}$ can answer all extraction oracle queries.

In the $\mathsf{Challenge}$ phase, $\mathcal{B}$ randomly chooses $b' \in \{0, 1\}$ and computes the SF ciphertext:

$$C_0' = M_{b'}^* \cdot T, \quad C_1' = (g^s Y_2)^{\alpha - \mathsf{ID}^*}, \quad C_2' = (g^s Y_2)^{\beta}.$$

If $T = \hat{e}(g, g)^{as}$, then $\mathcal{B}$ simulates Game$_q$. Otherwise, $\mathcal{B}$ simulates Game$_{\text{final}}$. If $\mathcal{A}$ can distinguish between these two games, $\mathcal{B}$ can break Assumption 3. Thus, no PPT adversary $\mathcal{A}$ can distinguish between Game$_q$ and Game$_{\text{final}}$.

B Security Proof for Accountable-Authority IBE

The IND-ID-CCA security of our A-IBE scheme can be proven analogously to our prime-order IBE scheme (under the $(2, 6)$-decisional subspace assumption) and hence omitted. In particular, the decryption oracle is simulated by using either the normal or semi-functional key to decrypt normal ciphertext. The semi-functional challenge ciphertext cannot be queried to the decryption oracle, and modifying the challenge ciphertext will not give a valid ciphertext (by the unforgeability of the strong one-time signature scheme).

Dishonest User Security.

Theorem 4. *If SXDH holds, then no PPT adversary has a non-negligible advantage in the ComputeNewKey-CCA game.*

Proof of Theorem 4. We consider the probability the tracing algorithm increases ctr.

Lemma 8. *If SXDH holds, in the ComputeNewKey-CCA game, if* $\mathbf{D}^*$ *correctly opens well-formed ciphertexts with non-negligible probability, then the probability of the tracing algorithm increasing* ctr *is also non-negligible.*

Proof. First, we define the semi-functional key and ciphertext.

An *SF secret key* (or just *SF key*) is in the form of

$$\overrightarrow{K_1'} = \overrightarrow{K_1} \cdot g_2^{\gamma_3 \overrightarrow{d_3^*}}, \quad \overrightarrow{K_2'} = \overrightarrow{K_2} \cdot g_2^{\gamma_4 \overrightarrow{d_4^*}},$$

where $\gamma_3, \gamma_4 \in \mathbb{Z}_p$, and $\mathsf{sk} = (\overrightarrow{K_1}, \overrightarrow{K_2})$ is a normal secret key.

An *SF ciphertext* can be computed by randomly picking $s, v \in \mathbb{Z}_p$ and running $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{KG}(1^\lambda)$. It outputs $C = (C_0', \overrightarrow{C_1'}, \sigma, \mathsf{vk})$ by

$$C_0' = M \cdot g_T^{\beta \cdot s}, \quad \overrightarrow{C_1'} = g_1^{s(\alpha - \mathsf{ID})\overrightarrow{d_1} + s\overrightarrow{d_2} + v \cdot \mathsf{vk}\overrightarrow{d_5}} \cdot g_1^{\delta_3 \overrightarrow{d_3} + \delta_4 \overrightarrow{d_4}}, \quad \sigma = \mathsf{Sign}(\mathsf{sk}, C_0 || \overrightarrow{C_1}),$$

where $\delta_3, \delta_4 \in \mathbb{Z}_p$.

Let Game$_{\text{real}}$ be the original adaptive-ID ComputeNewKey-CCA game. Let Game$_q$ be the same as Game$_{\text{real}}$, except that the ciphertext used to feed into the decoder box in Trace and the keys used to answer queries are changed from normal to semi-functional. Let Game$_{q'}$ be the same as Game$_q$, except that the ciphertext used to feed into the decoder box in Trace is changed to a valid ciphertext s.t. $s' = s$ (while retaining the semi-functional components). Let Game$_{\text{final}}$ be the same as Game$_{q'}$, except that the ciphertext used to feed into the decoder box in Trace and the keys used to answer queries are changed from semi-functional to normal.

Similar to the IND-ID-CCA proof, (Game$_{\text{real}}$, Game$_q$) and (Game$_{q'}$, Game$_{\text{final}}$) are indistinguishable under SXDH. In Game$_{\text{final}}$, the adversary is provided normal keys and the decoder box is fed with a well-formed ciphertext. Therefore, if $\mathbf{D}^*$ is ϵ-useful, then it increases ctr with non-negligible probability.

Next, we show that all adversaries cannot distinguish between Game$_q$ and Game$_{q'}$. We randomly pick $\zeta_1, \zeta_2 \in \mathbb{Z}_p$ and define new dual orthonormal bases $\mathbb{F} := (\overrightarrow{f_1}, \dots, \overrightarrow{f_6})$ and $\mathbb{F}^* := (\overrightarrow{f_1^*}, \dots, \overrightarrow{f_6^*})$ as follows

$$\begin{bmatrix} \overrightarrow{f_1} \\ \overrightarrow{f_2} \\ \overrightarrow{f_3} \\ \overrightarrow{f_4} \\ \overrightarrow{f_5} \\ \overrightarrow{f_6} \end{bmatrix} := \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \zeta_1 & 0 & 1 & 0 & 0 & 0 \\ 0 & \zeta_2 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \overrightarrow{d_1} \\ \overrightarrow{d_2} \\ \overrightarrow{d_3} \\ \overrightarrow{d_4} \\ \overrightarrow{d_5} \\ \overrightarrow{d_6} \end{bmatrix}, \quad \begin{bmatrix} \overrightarrow{f_1^*} \\ \overrightarrow{f_2^*} \\ \overrightarrow{f_3^*} \\ \overrightarrow{f_4^*} \\ \overrightarrow{f_5^*} \\ \overrightarrow{f_6^*} \end{bmatrix} := \begin{bmatrix} 1 & 0 & -\zeta_1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -\zeta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \overrightarrow{d_1^*} \\ \overrightarrow{d_2^*} \\ \overrightarrow{d_3^*} \\ \overrightarrow{d_4^*} \\ \overrightarrow{d_5^*} \\ \overrightarrow{d_6^*} \end{bmatrix}.$$

It is easy to verify that $\mathbb{F}$ and $\mathbb{F}^*$ are also dual orthonormal.

Then the public parameters, master public key, queried secret keys in Game$_q$, and challenge ciphertext expressed over $\mathbb{D}$ and $\mathbb{D}^*$ are, respectively,

$$(g_1^{\overrightarrow{d_1}}, g_1^{\overrightarrow{d_2}}, g_1^{\overrightarrow{d_5}}, g_1^{\overrightarrow{d_6}}, g_1^{\alpha\overrightarrow{d_1}}, g_1^{v\overrightarrow{d_1}}, g_T^{\beta}),$$

$$K = (\overrightarrow{K_1} = g_2^{(\frac{\beta - r}{\alpha - \mathsf{ID}})\overrightarrow{d_1^*} + t\overrightarrow{d_5^*} + v \cdot t\overrightarrow{d_6^*} + \gamma_3 \overrightarrow{d_3^*}}, \quad \overrightarrow{K_2} = g_2^{r\overrightarrow{d_2^*} + \gamma_4 \overrightarrow{d_4^*}}),$$

$$C = (\overrightarrow{C_1} = g_1^{s(\alpha - \mathsf{ID})\overrightarrow{d_1} + s'\overrightarrow{d_2} + v \cdot \mathsf{vk}\overrightarrow{d_5} + \delta_3 \overrightarrow{d_3} + \delta_4 \overrightarrow{d_4}}, \quad C_0 = M \cdot \hat{e}_6(\overrightarrow{C_1} \cdot (g_1^{\overrightarrow{d_6}})^{\mathsf{vk}}, \overrightarrow{K_1} \cdot \overrightarrow{K_2}), \sigma),$$

which can be expressed over $\mathbb{F}$ and $\mathbb{F}^*$ as

$$
\begin{aligned}
&(g_1^{\vec{f_1}}, g_1^{\vec{f_2}}, g_1^{\vec{f_5}}, g_1^{\vec{f_6}}, g_1^{\alpha\vec{f_1}}, g_1^{v\vec{f_1}}, g_T^{\beta}),\\
&K = (\overrightarrow{K_1} = g_2^{(\frac{\beta-r}{\alpha-\mathsf{ID}})\vec{f_1^*}+t\vec{f_5^*}+v\cdot t\vec{f_6^*}+\gamma_3'\vec{f_3^*}}, \quad \overrightarrow{K_2} = g_2^{r\vec{f_2^*}+\gamma_4'\vec{f_4^*}}),\\
&C = (\overrightarrow{C_1} = g_1^{z\vec{f_1}+z'\vec{f_2}+v\cdot\mathsf{vk}\vec{f_5}+\delta_3\vec{f_3}+\delta_4\vec{f_4}}, \quad C_0 = M\cdot\hat{e}_6(\overrightarrow{C_1}\cdot(g_1^{\vec{f_6}})^{\mathsf{vk}}, \overrightarrow{K_1}\cdot\overrightarrow{K_2}), \sigma),
\end{aligned}
$$

where

$$
\begin{aligned}
\gamma_3' &= \gamma_3 + (\frac{\beta-r}{\alpha-\mathsf{ID}})\zeta_1, &\quad \gamma_4' &= \gamma_4 + r\zeta_2,\\
z &= s(\alpha-\mathsf{ID}) - \zeta_1\delta_3, &\quad z' &= s' - \zeta_2\delta_4,
\end{aligned}
$$

They are uniformly random since $\zeta_1, \zeta_2, \delta_3, \delta_4$ are random. Similarly, the parameters in Game$_{q'}$ can be expressed over $\mathbb{F}$ and $\mathbb{F}^*$ with uniformly random coefficients. Thus, the adversary cannot distinguish between Game$_q$ and Game$_{q'}$.

Lastly, if $\mathbf{D}^*$ correctly opens well-formed ciphertexts with non-negligible probability in Game$_{\text{final}}$, the probability that the tracing algorithm increases ctr is also non-negligible. □

Dishonest PKG Security.

Theorem 5. *If SXDH holds, then no PPT adversary has a non-negligible advantage in the FindNewKey-CCA game.*

Proof of Theorem 5. We show that the probability of increasing ctr is negligible.

Lemma 9. *If SXDH holds, in the black-box FindNewKey-CCA game, one iteration of the tracing algorithm increases* ctr *with negligible probability.*

Proof. Given $D = (g_2^{\vec{d_1^*}}, g_2^{\vec{d_2^*}}, g_2^{\vec{d_5^*}}, g_2^{\vec{d_6^*}}, g_1^{\vec{d_1}}, \ldots, g_1^{\vec{d_6}}, U_1, U_2, \mu_2)$ along with $T = (T_1, T_2)$ being either (V_1, V_2) or (W_1, W_2) from $(2,6)$-decisional subspace assumption in $\mathbb{G}_1$, $\mathcal{B}$ generates crs_1 of the concurrent zero knowledge proof of knowledge with a knowledge extractor; and sends to $\mathcal{A}$ the parameters $(g_2^{\vec{d_1^*}}, g_2^{\vec{d_2^*}}, g_2^{\vec{d_5^*}}, g_2^{\vec{d_6^*}}, g_1^{\vec{d_1}}, g_1^{\vec{d_2}}, g_1^{\vec{d_5}}, g_1^{\vec{d_6}}, \mathsf{crs}_1)$ in the $\mathsf{Initialize}$ phase. $\mathcal{A}$ chooses the master secret key and sends the master public key mpk and a challenge identity ID^* to $\mathcal{B}$.

During the $\mathsf{Extract}$ phase for ID^*, $\mathcal{A}$ sends (A_1, A_2) and a proof π_1 to $\mathcal{B}$ in Steps 1 and 2. $\mathcal{B}$ uses the knowledge extractor with respect to crs_1 to get r_1 s.t. $A_1 = g_2^{r_1\vec{d_1^*}}$, $A_2 = g_2^{r_1\vec{d_2^*}}$. In Step 3, $\mathcal{B}$ picks some random r_0', ρ and implicitly sets $r_0 = r_0' \cdot \mu_1$ and (with private input $\rho, A_1, U_1^{r_0' r_1}$) interacts with $\mathcal{A}$ in a secure two-party computation protocol. $\mathcal{A}$ obtains the output $\overrightarrow{K'}$ and sends $\overrightarrow{K'}$ to $\mathcal{B}$. $\mathcal{B}$ then computes $\overrightarrow{\tilde{K}_1} = \overrightarrow{K'}^{1/\rho}, \overrightarrow{\tilde{K}_2} = U_2^{r_0' r_1}$. Note that the correct values of $\overrightarrow{K_1}$ and $\overrightarrow{K_2}$ are not known by $\mathcal{B}$. Instead, $\mathcal{B}$ only has an SF key

$$
(\overrightarrow{\tilde{K}_1} = g_2^{(\frac{\beta - r_0' r_1 \mu_1}{\alpha-\mathsf{ID}})\vec{d_1^*} - \frac{r_0' r_1 \mu_2}{\alpha-\mathsf{ID}}\vec{d_3^*} + t\vec{d_5^*} + v\cdot t\vec{d_6^*}}, \quad \overrightarrow{\tilde{K}_2} = g_2^{r_0' r_1 \mu_1 \vec{d_2^*} + r_0' r_1 \mu_2 \vec{d_4^*}}),
$$

where $U_1 = g_2^{\mu_1\vec{d_1^*}+\mu_2\vec{d_3^*}}$, $U_2 = g_2^{\mu_1\vec{d_2^*}+\mu_2\vec{d_4^*}}$. Nevertheless, $\mathcal{B}$ can still answer all decryption oracle queries using the SF key. It is because $\mathcal{A}$ has not seen any $g_1^{\vec{d_3}}, g_1^{\vec{d_4}}$ elements so far and cannot produce an SF ciphertext.

Finally, $\mathcal{A}$ outputs an ϵ-useful decoder box $\mathbf{D}^*$. If the tracing algorithm increases ctr, $\mathbf{D}^*$ must decrypt any ciphertext as if using the real $\mathsf{sk}_{\mathsf{ID}^*}$. $\mathcal{B}$ randomly picks $s, v \in \mathbb{Z}_p$, $M \in \mathbb{G}_T$; runs $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{KG}(1^\lambda)$; and computes

$$
C_0 = M \cdot g_T^{\beta s}, \quad \overrightarrow{C_1} = g_1^{s(\alpha-\mathsf{ID}^*)\vec{d_1}+v\cdot\mathsf{vk}\vec{d_5}} \cdot T_2,
$$

and $\sigma = \mathsf{Sign}(\mathsf{sk}, C_0 || \overrightarrow{C_1})$. $\mathcal{B}$ submits $C = (C_0, \overrightarrow{C_1}, \sigma, \mathsf{vk})$ to the decoder box $\mathbf{D}^*$. Note that $\mathbf{D}^*$ (generated by the dishonest PKG) may be able to recognize invalid ciphertexts in the tracing stage. However, assuming $\mathbf{D}^*$ is stateless, it cannot shut down or self-destruct when detecting a tracing attempt. $\mathbf{D}^*$ tries to decrypt such invalid ciphertexts in the same way as the owner of the identity-based secret key $\mathsf{sk}_{\mathsf{ID}^*}$, and outputs a decrypted message M^*. Observe that $\mathbf{D}^*$ should not output $\perp$ since σ is a valid strong one-time signature. If this iteration of the tracing algorithm increases ctr with probability ϵ, then with the same probability,

$$M^* = C_0/[\hat{e}_6(\overrightarrow{C_1} \cdot (g_1^{\overrightarrow{d_6}})^{\mathsf{vk}}, \overrightarrow{K_1} \cdot \overrightarrow{K_2})]$$

corresponding to the correct key $(\overrightarrow{K_1}, \overrightarrow{K_2})$. With M^* stated above, $\mathcal{B}$ can break the $(2,6)$-decisional subspace assumption. Specifically, $\mathcal{B}$ computes

$$T' \quad = \quad \frac{C_0}{M^* \cdot [\hat{e}_6(\overrightarrow{C_1} \cdot (g_1^{\overrightarrow{d_6}})^{\mathsf{vk}}, \overrightarrow{\tilde{K}_1} \cdot \overrightarrow{\tilde{K}_2})]}.$$

If $T_2 = g_2^{\tau_1 \overrightarrow{b_2^*}}$, then the above equation equals 1. If $T_2 = g_2^{\tau_1 \overrightarrow{b_2^*} + \tau_2 \overrightarrow{b_4^*}}$, then the above equation gives a non-one value $\hat{e}(g_1, g_2)^{s_0' s_1 \mu_2 \tau_2 \overrightarrow{b_4} \overrightarrow{b_4^*}}$. Therefore, if $\mathbf{D}^*$ successfully decrypt with non-negligible probability, the $(2,6)$-decisional subspace assumption can be broken. □

C Proof for Our Exponent-Inversion Signatures

We first review the definitions of dual form signatures:

- Setup: Given a parameter 1^λ, generate a public/private key pair $(\mathsf{pk}, \mathsf{sk})$.
- Sign_A: Given sk and a message M, output a signature σ. This is a signing algorithm used in the real construction.
- Sign_B: Given sk and a message M, output a signature σ. This is a signing algorithm used only in the proofs.
- Verify: Given pk, a signature σ, and a message M, output 1 or 0.

Forgery Class. Let $\mathcal{V}$ denote the set of signature-message pairs for which the Verify algorithm outputs 1. Let $\mathcal{V}_I$ and $\mathcal{V}_{II}$ be two disjoint subsets of $\mathcal{V}$, such that $\mathcal{V} = \mathcal{V}_I \cup \mathcal{V}_{II}$. Signatures from these sets are referred to as Type I and Type II forgeries, corresponding to two types of forgeries received from an adversary in our security proof. Type I forgeries will be related to signatures output by the Sign_A algorithm, and Type II forgeries will be related to those by the Sign_B algorithm. The precise relationships between the forgery types and the signing algorithms are explicitly defined by the following set of security properties for the dual form system.

Security Properties. We briefly review the following properties of dual form signatures [GLOW12], where the adversary is given only pk for $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda)$.

- $\mathsf{A\text{-}I\ Matching}$. If an attacker is only given a signing oracle that returns outputs from Sign_A, then it is hard to create anything but a Type I forgery.
- $\mathsf{B\text{-}II\ Matching}$. If an attacker is only given a signing oracle that returns outputs from Sign_B, then it is hard to create anything but a Type II forgery.

- Dual Oracle Invariance. The attacker $\mathcal{A}$ is given oracle accesses to Sign_A and Sign_B. At some point, $\mathcal{A}$ outputs a challenge message M. The challenger returns a challenge signature on M from either Sign_A or Sign_B with equal probability. Finally, $\mathcal{A}$ outputs a forgery pair (M^*, σ^*), where M^* was not asked to any oracle. The probability that $\mathcal{A}$ produces a Type I forgery when the challenge signature is from Sign_A is approximately the same as when the challenge signature is from Sign_B.

A dual form signature scheme is secure if it satisfies all these properties.

Theorem 6 ([GLOW12])**.** *If* $(\mathsf{Setup}, \mathsf{Sign}_A, \mathsf{Sign}_B, \mathsf{Verify})$ *is a dual form signature scheme,* $(\mathsf{Setup}, \mathsf{Sign}_A, \mathsf{Verify})$ *is existentially unforgeable under adaptive chosen message attacks.*

Theorem 7. *Our dual form Boneh-Boyen signature is existentially unforgeable under an adaptive chosen message attack if Assumptions 1, 2, and 3 hold.*

Let Sign_A be the original signing algorithm, and define its dual form, Sign_B, as follows.

$\mathsf{Sign}_B(\mathsf{sk}, M)$: The signer randomly picks $r \in \mathbb{Z}_N$, $X_{2,3}, X'_{2,3} \in \mathbb{G}_{p_2 p_3}$ and computes the signature $\sigma = (\sigma_1, \sigma_2)$, where:

$$\sigma_1 = (h_1 u_1^{-r})^{\frac{1}{\alpha - M}} X_{2,3}, \quad \sigma_2 = g_1^r X'_{2,3}.$$

The A-I and B-II matching properties can be proven similarly to the security proof of the IBE scheme. Specifically, the challenger can compute some verifying parameters (counterparts to the challenge ciphertexts in IBE proofs) with the underlying assumptions. Once a PPT adversary $\mathcal{A}$ creates a Type-II/I forgery in A-I/B-II matching, the verification result of the forgery with the verifying parameters allows the challenger to break the assumptions. We show that our scheme has the dual oracle invariance property with Lemma 10. Above all, our scheme is existentially unforgeable with Theorem 6.

Lemma 10. *If Assumption 2 holds, our scheme satisfies dual oracle invariance.*

Proof. Given $(g, X_1 X_2, X_3, Y_2 Y_3, T)$ from Assumption 2, $\mathcal{B}$ chooses random $b, \alpha \in \mathbb{Z}_N$, $h_1 \in \mathbb{G}_{p_1}$. $\mathcal{B}$ sets $g_1 = g$, $u_1 = g^b$, $g_{2,3} = Y_2 Y_3$, and $g_3 = X_3$. $\mathcal{B}$ generates the rest of pk and the secret key $\mathsf{sk} = (h_1, \alpha, g_3, g_{2,3})$ according to Setup.

For the oracle queries to Sign_A, $\mathcal{B}$ randomly picks $r, w, v \in \mathbb{Z}_N$ and uses sk to compute the signature $\sigma = (\sigma_1 = (h_1 u_1^{-r})^{1/(\alpha - M)} X_3^w, \sigma_2 = g_1^r X_3^v)$.

For the oracle queries to Sign_B, $\mathcal{B}$ randomly picks $r, w, v \in \mathbb{Z}_N$ and computes the signature $\sigma = (\sigma_1, \sigma_2)$, where:

$$\sigma_1 = (h_1 u_1^{-r})^{\frac{1}{\alpha - M}} (Y_2 Y_3)^w, \quad \sigma_2 = g_1^r (Y_2 Y_3)^v.$$

By the Chinese remainder theorem, the values of v and w modulo p_2 and those modulo p_3 are uncorrelated. Finally, $\mathcal{A}$ queries $\mathcal{B}$ with a challenge message M. $\mathcal{B}$ chooses some random $w, v \in \mathbb{Z}_N$, and computes the signature:

$$\sigma_1 = h_1^{\frac{1}{\alpha - M}} \cdot T^{\frac{-b}{\alpha - M}} \cdot X_3^w, \quad \sigma_2 = T \cdot X_3^v.$$

If $T = Z_1 Z_3$, it is a signature from Sign_A. If $T = Z_1 Z_2 Z_3$, it is from Sign_B. Note that b modulo p_2 is not revealed at any point during the Query phase. So the $\mathbb{G}_{p_2}$ part of σ_1^* is randomly distributed from the view of $\mathcal{A}$.

Once A returns the forgery, (σ^*, M^*), $\mathcal{B}$ must first check that $\mathcal{A}$ has not previously seen a signature for M^* before and that (σ^*, M^*) passes verification. If either of these checks fails, $\mathcal{B}$ will guess randomly. If both pass, $\mathcal{B}$ determines what forgery class (σ^*, M^*) belongs to in order to determine what subgroup T is in, via a backdoor verification test similar to that in the previous proof. $\mathcal{B}$ sets: $C_0^* = \hat{e}(X_1 X_2, h_1)$, $C_1^* = (X_1 X_2)^{\alpha - M^*}$,

$C_2^* = (X_1X_2)^b$. Parse $\sigma^* = (\sigma_1^*, \sigma_2^*)$. $\mathcal{B}$ proceeds with a backdoor verification test by returning:

$$C_0^* \overset{?}{=} \hat{e}(C_1^*, \sigma_1^*) \cdot \hat{e}(C_2^*, \sigma_2^*).$$

If it holds, $\mathcal{B}$ flips a coin $b' \in \{0,1\}$ and returns b'. Otherwise, $\mathcal{B}$ outputs 1.

If $\mathcal{A}$ returns a Type I forgery, it also passes the backdoor verification test since it passes the real signature verification. If $\mathcal{A}$ returns a Type II forgery, suppose the $\mathbb{G}_{p_2}$ parts of σ_1^* and σ_2^* are $\hat{g}_2^{\delta_1}$ and $\hat{g}_2^{\delta_2}$ respectively, for some $\hat{g}_2 \in \mathbb{G}_{p_2}$, $\delta_1, \delta_2 \in \mathbb{Z}_N$ and either δ_1 or δ_2 is non-zero modulus p_2. Then the backdoor verification equation proceeds by returning:

$$\hat{e}(C_1^*, \sigma_1^*)\hat{e}(C_2^*, \sigma_2^*) = C_0^*\hat{e}(\hat{g}_2^{\delta_1}, X_2^{\alpha - M^*})\hat{e}(\hat{g}_2^{\delta_2}, X_2^b) = C_0^*\hat{e}(\hat{g}_2, X_2)^{\delta_1(\alpha - M^*) + \delta_2 b} \overset{?}{=} C_0^*.$$

Thus, if the forgery fails the test, it is a Type II forgery with probability 1. Any forgery passing the test can be either Type I or Type II. A Type II forgery might also pass the additional verification test, but only with negligible probability. To see, for such a Type II forgery, we have $\delta_1(\alpha - M^*) + \delta_2 b = 0 \bmod p_2$. Consider:

1. If $\delta_1 = 0 \bmod p_2$ and $\delta_2 \neq 0 \bmod p_2$, it implies $b = 0 \bmod p_2$. It happens with negligible probability since b is randomly chosen by $\mathcal{B}$ from $\mathbb{Z}_N$.
2. If $\delta_1 \neq 0 \bmod p_2$, we rewrite the equation as $(\alpha - M^*) + \delta b = 0 \bmod p_2$, where $\delta = \delta_2/\delta_1$. To create such a Type II forgery, an adversary must implicitly determine $(\alpha - M^*)/b$ modulo p_2. The adversary only knows $b/(\alpha - M)$ modulo p_2 from the challenge signature if $T = Z_1Z_2Z_3$. As long as $M \neq M^*$ modulo p_2, the adversary has only a negligible probability of achieving the correct value of δ modulo p_2.

We now consider the information obtained by adversary $\mathcal{A}$. In the challenge signature, α and b modulo p_2 are only included in the first element. Thus, $\mathcal{A}$ can only derive the single value $\frac{b}{\alpha - M}$ modulo p_2. However, this single equation has two unknowns α and b modulo p_2, and it is not possible to determine their unique values. Moreover, $\frac{b}{\alpha - M}$ is a pairwise independent function of M modulo p_2 (except with negligible probability that $\alpha = M \bmod p_2$). Therefore, $\mathcal{A}$ cannot achieve the correct value of $\frac{b}{\alpha - M^*} \bmod p_2$ as long as $M \neq M^* \bmod p_2$, except with negligible probability. It is possible that $M = M^*$ modulo p_2, but $M \neq M^*$ modulo N. If this occurs with non-negligible probability, $\mathcal{B}$ can extract a non-trivial factor of N as the greatest common divisor of N and $M - M^*$, and use it to break Assumption 2 with non-negligible advantage. So, if a forgery passes the additional verification test, with a high probability, it is a Type I forgery. □

Security Proof for Our Prime-Order Version and its F-unforgeability. The proof for our prime-order signature scheme largely follows by adapting the changes from our composite-order dual form Gentry-IBE to its prime-order counterpart, but starting from our composite-order dual form Boneh-Boyen signatures. Namely, the forgery classes Type-I and Type-II are based on whether the signature has the $\overrightarrow{d_3}$ and $\overrightarrow{d_4}$ components.

Additionally, to construct a P-signature scheme from our signature scheme, we require an F-unforgeability property, where the adversary is only asked to output $(\sigma^*, F(M^*))$ instead of (σ^*, M^*) as the forgery for some bijective function F.

Theorem 8. *Let $F(x) = (Y^x, Z^x)$, where Y, Z are in the span of $\overrightarrow{d_1}, \overrightarrow{d_3}$ in the exponent. Our prime-order signature scheme is F-unforgeable under an adaptive chosen-message attack if SXDH holds.*

The proof largely follows the proof of standard unforgeability. The additional elements Y and Z in the public key enable the backdoor verification test when the adversary returns $(\sigma^*, F(M^*))$ as the F-forgery for $F(x) = (Y^x, Z^x)$.

D Security Proof for Our Anonymous Credentials

Theorem 3. *Our P-signature scheme is secure under the SXDH assumption and the security of the two-party computation.*

Proof. **Signer Privacy.** We construct the algorithm $\mathsf{SimIssue}(\mathsf{param}, C, \overrightarrow{\sigma_1}, \overrightarrow{\sigma_2})$ to simulate the adversary's view. Firstly, SimIssue invokes the simulator of the two-party computation protocol, and extracts the input of the adversary, which is $(\rho_1, \rho_2, M, \mathsf{Open})$ in this case. SimIssue checks if $C \overset{?}{=} \mathsf{GS.Commit}(\mathsf{param}_{\mathsf{GS}}, Z^M, \mathsf{Open})$; if it is not, it terminates. Otherwise, it sends $\overrightarrow{\sigma_1'} = \overrightarrow{\sigma_1}^{1/\rho_1}, \overrightarrow{\sigma_2'} = \overrightarrow{\sigma_2}^{\rho_2}$ to the adversary $\mathcal{A}$. If $\mathcal{A}$ can detect that it is interacting with a simulator, it breaks the security of the two-party computation.

User Privacy. We construct the algorithm $\mathsf{SimObtain}(\mathsf{param}, \mathsf{pk}, C)$ to simulate the adversary's view. Firstly, SimObtain invokes the simulator of the two-party computation protocol, and extracts the input of the adversary, which is (sk', r) in this case (where sk' is not necessarily the valid secret key used). The simulator picks random $\rho_1, \rho_2 \in \mathbb{Z}_p$ and computes $\overrightarrow{\sigma_1'}, \overrightarrow{\sigma_2'}$ using (sk', r) and M. It proceeds to interact with the adversary such that if the adversary completes the protocol, its output is $(\overrightarrow{\sigma_1'}, \overrightarrow{\sigma_2'}, C)$. If the adversary can detect that it is interacting with a simulator, it breaks the security of the two-party computation.

Zero-knowledge/Witness Indistinguishability. From the security of the Groth-Sahai proof system, the simulator can run a setup simulation for $\mathsf{param}'_{\mathsf{GS}}$. The distribution of $\mathsf{param}'_{\mathsf{GS}}$ is computationally indistinguishable from the real $\mathsf{param}_{\mathsf{GS}}$. Using $\mathsf{param}'_{\mathsf{GS}}$, commitments are perfectly hiding. The simulator can compute the output (comm, π) using a simulation algorithm SimProve on $\mathsf{param}'_{\mathsf{GS}}$. It is witness indistinguishable from the real proof of the Prove algorithm with input (M, σ). Similarly, the output of EqCommProve can also be simulated by SimProve via the composable zero-knowledge property.

Unforgeability. Suppose an adversary $\mathcal{A}$ can break the unforgeability of our P-signatures, then we construct an algorithm $\mathcal{B}$ to break the F-unforgeability of the underlying DFEI signature, where $F(x) = (Y^x, Z^x)$. Firstly, $\mathcal{B}$ obtains pk' from the challenger of the DFEI signature. By the security of the Groth-Sahai proof system, $\mathcal{B}$ can run a setup simulation for $\mathsf{param}'_{\mathsf{GS}}$ and obtain an extraction trapdoor td. The distribution of $\mathsf{param}'_{\mathsf{GS}}$ is identical to the real $\mathsf{param}_{\mathsf{GS}}$. $\mathcal{B}$ gives param and pk to $\mathcal{A}$ using pk' and $\mathsf{param}'_{\mathsf{GS}}$. For all signing oracle queries on message M, $\mathcal{B}$ forwards the query to its challenger to answer it.

Finally, if $\mathcal{A}$ can output a proof π that VerifyPf outputs 1, then the simulator can use td to extract (Z^M, Y^M, σ) from the commitments comm. If $\mathcal{A}$ wins the security game by:

1. σ is not a valid signature on M, or comm is not a commitment to M, it breaks the soundness of the Groth-Sahai proof system.

2. $\mathcal{A}$ has never queried the signing oracle on M, then $\mathcal{B}$ returns $(\sigma, F(M) = (Y^M, Z^M))$ as the forgery to the DFEI signature.

Therefore, if $\mathcal{A}$ can break the unforgeability of our P-signature scheme, we can break SXDH. □