Hindawi
IET Information Security
Volume 2023, Article ID 8829580, 17 pages
https://doi.org/10.1049/2023/8829580

Research Article

Inner-Product Matchmaking Encryption: Bilateral Access Control and Beyond Equality

Qiaohan Chu,[1] **Anmin Fu**,[2] **Haifeng Qian**,[1] **and Jie Chen**[1]

[1]*Shanghai Key Laboratory of Trustworthy Computing, Software Engineering Institute, East China Normal University, Shanghai 200062, China*
[2]*School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China*

Correspondence should be addressed to Jie Chen; s080001@e.ntu.edu.sg

Received 18 July 2023; Revised 26 September 2023; Accepted 6 October 2023; Published 2 November 2023

Academic Editor: Helena Rifà-Pous

We present an inner-product matchmaking encryption (IP-ME) scheme achieving weak privacy and authenticity in prime-order groups under symmetric external Diffie–Hellman (SXDH) assumption in the standard model. We further present an IP-ME with Monotone Span Program Authenticity (IP-ME with MSP Auth) scheme, where the chosen sender policy is upgraded to MSP, and the scheme also achieves weak privacy and authenticity in prime-order groups under SXDH assumption in the standard model. Both of the schemes have more expressive functionalities than identity-based matchmaking encryption (IB-ME) scheme, and are simpler than Ateniese et al.'s modular ME scheme (Crypto' 19). But our schemes only achieve a very limited flavor of security, which is reflected in the privacy.

1. Introduction

1.1. Background

1.1.1. Matchmaking Encryption. Matchmaking Encryption (ME) is a cryptographic primitive introduced by Ateniese et al.'s [1] work. It is motivated by trying to work out a noninteractive version of secret handshake (SH) protocol [2] in order to get rid of real-time interactions, and further enhance the privacy of participants. Except noninteractivity and strong privacy, the definition of ME proposed by Ateniese et al. [1] also provides the property of authenticity, so that elminating the "not credible" problem in anonymous communication.

Specifically, an ME scheme works as follows: the authority generates sender's key ek_σ with sender's attributes σ, and receiver's key dk_ρ with receiver's attributes ρ, and sends them to the sender and the receiver, respectively. When the sender wants to send a secret message, he specifies a policy $\mathbb{R}$, and encrypts the message with ek_σ and $\mathbb{R}$, so that only the receiver whose attributes ρ match the policy $\mathbb{R}$ has the right for decryption. On the other hand, the receiver can also specify a policy $\mathbb{S}$, and make a query of $\mathsf{dk}_\mathbb{S}$ to the authority, so that the receiver can identify the information source.

Based on the functionality of ME, there are several applications for ME in the real world. For example, by Ateniese et al. [1], there says that the sender can specify the receiver who is an FBI agent and lives in NYC, and the receiver can also specify the sender who is a CIA agent. If the decryption fails, no private information will leak. Another example by Ateniese et al. [1] is encryption bids. Bidders send private bids to a collector encrypted with their chosen conditions, and the collector opens the bids that match specific requirements. Also, if the decryption fails, the collector does not know the reason and gains no information about the actual bids. Ateniese et al. [1] also presents an implementation of privacy-preserving bulletin board combining Tor hidden services with ME that allowing parties to collect information from anonymous but authentic sources.

1.1.2. Identity-Based Matchmaking Encryption. A special case of ME is identity-based ME (IB-ME), where the two policies are both equality. And since its policy is simple, IB-ME removes the algorithm PolGen (ref. Section 2.4), so that it can eliminate the process of sending the decryption key $\mathsf{dk}_\mathbb{S}$ from the authority to the receiver. IB-ME is well-suited for the application of spy communication that the spy can

Table 1: Comparison for currently nontheoretical ME schemes.

Scheme	Functionality	Assumption	Model	Privacy
AFNV19 [1]	ME	FE, SS, NIZK	—	Full
AFNV19 [1]	IB-ME	BDH	RO	Full (weak)
FGRV21 [3]	IB-ME	q-Type, NIZK	Standard	Enhanced
CLWW22 [4]	IB-ME	SXDH	Standard	Full (weak)
Π_{IP_C}	IP-ME	SD	Standard	Weak
Π_{IP_P}	IP-ME	SXDH	Standard	Weak
Π_{IPMSP_C}	IP-ME with MSP Auth	SD	Standard	Weak
Π_{IPMSP_P}	IP-ME with MSP Auth	SXDH	Standard	Weak

[1]FE denotes functional encryption. [2]SS denotes signature scheme. [3]NIZK denotes noninteractive zero-knowledge proofs. [4]For IB-ME, full privacy is equivalent to weak privacy. [5]For IP-ME with MSP Auth, the chosen receiver policy, $\mathbb{R}$ is inner-product and the chosen sender policy $\mathbb{S}$ is MSP. [6]Π_{IP_C}, Π_{IP_P}, Π_{IPMSP_C} and Π_{IPMSP_P} are the schemes presented in this work.

encrypt and decrypt the messages simply in the light of identities.

There have been several works about IB-ME. The first proposed IB-ME scheme is from Ateniese et al.'s [1] work, which is comparatively simple and concrete, and based on Bilinear Diffie–Hellman (BDH) assumption in the random oracle model. Then, Francati et al. [3] improve the random oracle model into the standard model, but under a nonstandard q-type assumption. Subsequently, Chen et al. [4] accomplish IB-ME under standard Symmetric External Diffie–Hellman (SXDH) assumption in the standard model and with a more direct construction.

1.1.3. Inner-Product Matchmaking Encryption. When the two policies are restricted to inner-product, we can obtain another special case of ME, i.e., inner-product ME (IP-ME). The inner-product policy demands that only the attributes, whose inner product with the vector of policy is zero, can match it. This policy can be adopted into some real scenarios, especially statistics related scenarios. For example, a company S, playing the role of sender, specifies a weight vector as the policy, and he wants to tell the company R, playing the role of receiver, a secret (e.g., "We can cooperate against the company A"), whose weighted sum of attributes (e.g., scores) equal to the target value. When company R receives the ciphertext, he tries to decrypt it with his chosen weight vector. If the decryption succeeds, it implies that company R is willing to cooperate with company S, and otherwise, there would not be any cooperation between company S and company R.

1.1.4. Inner-Product Matchmaking Encryption with Monotone Span Program Authenticity. We can further upgrade IP-ME to IPME with Monotone Span Program Authenticity (IP-ME with MSP Auth), where the chosen sender policy $\mathbb{S}$ is changed intoMSP [5–7]. This provides more power for the receiver, since the policy $\mathbb{S}$ is more expressive. When it is in the above "cooperation" scenario, company R can specify his cooperator more precisely by setting more precise policy.

1.2. Contributions. In this work, we mainly present an IP-ME scheme and an IP-ME with MSP Auth scheme, which are more expressive than IB-ME [1, 3, 4] and of simpler constructions than the modular ME [1], both in prime-order groups under standard SXDH assumption in the standard model. Our schemes are both with reasonable $O(n)$ sized parameters, where n denotes the size of each user's attributes, and both achieve authenticity but only weak privacy (ref. Def 4). As preparations for the prime-order versions, we also present the corresponding composite-order versions for our IP-ME and IP-ME with MSP Auth schemes. Our composite-order schemes are under subgroup decision (SD) assumption in the standard model, also with $O(n)$ sized parameters and achieve weak privacy and authenticity.

More specifically, our schemes are of the following advantages:

(i) *More Expressive Functionalities*: Compared to the current works of IB-ME with concrete constructions [1, 3, 4], our IP-ME and IP-ME with MSP Auth are of more expressive functionalities.

(ii) *Simpler and More Concrete Constructions*: Compared to the modular ME scheme Ateniese et al. [1], which is constructed of FE, Signature, and NIZK in a black-box manner, our schemes are directly constructed from a combination of two encryption instances, so that our schemes are simpler and more concrete than [1].

(iii) *Standard and Efficient*: Our schemes are under standard assumptions, SXDH and SD assumptions, and are in the standard model. Besides, our main schemes are in prime-order groups [8], and of $O(n)$ sized parameters, which is fairly reasonable since it is linear in the size of each user's attributes, not of a higher order of magnitude.

We would like to clarify that our schemes only achieve a very limited flavor of security notion compared with the original security notion of ME, since we cut down some possible cases.

We present a detailed comparison with currently related works in Table 1, and a detailed cost of our prime-order schemes in Table 2.

1.3. Technical Overview

1.3.1. Starting Point. Our goal is to construct simpler ME schemes than the modular one by Ateniese et al. [1], and

TABLE 2: The cost of our schemes in prime-order groups.

Scheme	$\lvert\mathsf{mpk}\rvert$	$\lvert\mathsf{ek}_\sigma\rvert$	$\lvert\mathsf{dk}_\rho\rvert$	$\lvert\mathsf{dk}_\mathbb{S}\rvert$	$\lvert\mathsf{ct}\rvert$
Π_{IP_p}	$3(n+2)\lvert G\rvert + \lvert G_T\rvert$	$6\lvert G\rvert$	$6\lvert H\rvert$	$3(n+1)\lvert H\rvert$	$3(n+1)\lvert G\rvert + \lvert G_T\rvert$
Π_{IPMSP_p}	$3(n+2)\lvert G\rvert + \lvert G_T\rvert$	$3(n+2)\lvert G\rvert$	$6\lvert H\rvert$	$3(n+1)\lvert H\rvert$	$3(n+2)\lvert G\rvert + \lvert G_T\rvert$

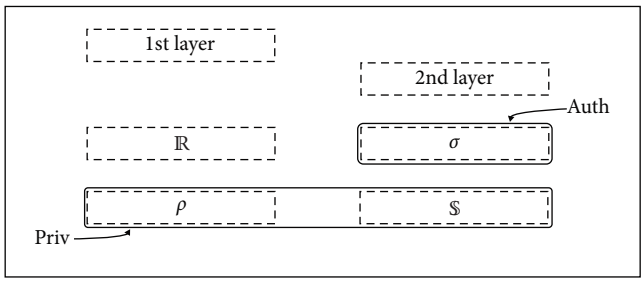

FIGURE 1: The two-layer structure from Chen et al.'s [4] work.

meanwhile extend the functionality of IB-ME, which has already been of several concrete constructions. Following [4], we start with the two-layer structure, which is actually a non-black-box combination of two instances of ABE schemes. Since compared with the study of Ateniese et al. [1], the two-layer structure only requires ABE as a building block, thus it might lead to simpler constructions. What makes the two-layer structure work is thanks to the fact that we can take the first layer instance as a weakly attribute-hiding ABE [9, 10], and take the second layer instance as a Signature with fine-grained control. We present an illustration for two-layer structure in Figure 1. And thus, thereinafter, our main task is trying to work out a way for combining the two instances.

1.3.2. Overview of Challenges. We would like to say ahead that such a combination is not trivial, since we need to guarantee the correctness and avoid the independence of the two instances simultaneously. And different from IB-ME, the more expressive ME requires the algorithm PolGen. This means that the design idea is very different from IB-ME [4], although the basic frameworks are both the two-layer structure. What is more, for the second signature layer, attribute-based signature (ABS) is a more complex primitive than identity-based signature (IBS), so the combination is more challenging.

1.3.3. IP-ME. As a first try, we consider how to combine two IPE instances. Before going to the details, we need to first select which IPE construction is our basic construction for each layer. Here, we use the modular framework by Chen et al. [11] and Wee [12] and the predicate encodings summarized Wee [12] to obtain our basic construction, and it is as below:

$$\begin{aligned}\mathsf{mpk} &= (\mathbb{G}, g^{w_1}, \ldots, g^{w_{n+1}}, e(g,h)^\alpha)\\ \mathsf{msk} &= (w_1, \ldots, w_{n+1}, \alpha),\end{aligned} \tag{1}$$

$$\begin{aligned}\mathsf{sk} &= (K_0 = h^r, K_1 = h^{\alpha + r w_2 y_1 + \cdots + r w_{n+1} y_n})\\ \mathsf{ct} &= (C_0 = g^s, C_1 = g^{s w_1 x_1 + s w_2}, \ldots,\\ &\quad C_n = g^{s w_1 x_n + s w_{n+1}}, C' = e(g,h)^{s\alpha} \cdot m)\\ \mathsf{Dec}: m &= C' / \left(e(C_0, K_1) / \left(e\left(C_1^{y_1}, K_0\right) \cdots e\left(C_n^{y_n}, K_0\right)\right)\right).\end{aligned} \tag{2}$$

When combining the two instances, we observe that the two instances need to be orthogonal with each other. That is, for example, for $K_0^{(1)}$ and $C_1^{(2)}$ (the superscripts (1) and (2) denote the instances in the first layer and in the second layer, respectively), it requires that $e\left(C_1^{(2)}, K_0^{(1)}\right) = [1]_T$, otherwise, there will be terms like $g_T^{s w_i' r'} / g_T^{s' w_i r}$ in decryption phase, which cannot be canceled out due to the different randomness s, s' and r, r' picked in the different instances. To obtain the orthogonality, we think about the technique used in Lewko and Waters's [13] work. Following the study of Lewko and Waters [13], we make the two instances in different subgroups. Then it comes to the challenge that how to combine the two instances validly. From a high-level, it seems that we can set dk_ρ and ek_σ as sk of IPE just with different randomness, set $\mathsf{dk}_\mathbb{S}$ as ct of IPE corresponding to ek_σ, and set $\mathsf{ct}_{\sigma,\mathbb{R}}$ as a combination of ek_σ and ct of IPE. However, this will make the two instances totally independent. That is, if we design the scheme as above, the decryptor will actually not need $\mathsf{dk}_\mathbb{S}$, and thus the sender can arbitrarily change ek_σ. This invalidates the second layer instance. To tackle this issue, we attach an element $h_3^{\alpha_2}$, which is in the subgroup of the second instance, to dk_ρ, so that if the decryptor does not use $\mathsf{dk}_\mathbb{S}$, he would not be able to decrypt the ciphertext successfully. Meanwhile, to guarantee the correctness of the scheme, we also need to attach some other components to some places, so that we can cancel out the extra element in dk_ρ. Our idea is to attach the same element to $\mathsf{dk}_\mathbb{S}$, then we can leverage the decryption process of IPE to remove this extra element. Notably, this design requires the first element of the sender's attribute vector to be 1. This can be easily achieved in inner-product setting, since we can assume the first element as 1 without loss of generality.

For security analysis, we observe that the two-layer structure prevents us from setting exactly the same mismatch conditions and match conditions as by Ateniese et al. [1]. For mismatch conditions, we can only set that ρ does not match $\mathbb{R}_0$ and ρ does not match $\mathbb{R}_1$. For match conditions, it actually corresponds to the fully attribute-hiding property, however, our basic IPE only achieves weakly attribute-hiding. Therefore, we relax the full privacy by Ateniese et al. [1] to a weak version here (weaker mismatch conditions and without match conditions). This is a weaker and very limited

security notion. As for authenticity, it can be directly reduced to the security of the second layer IPE scheme.

Next, we need to transform the composite-order version into prime-order version. By now, there has been a line of research on the techniques for simulating composite-order groups into prime-order groups [7, 11, 14–21], which can be divided into two categories: dual system group (DSG) [22] and dual pairing vector spaces (DPVS) [23, 24]. For DSG, it seems to be more efficient and simpler, however, it crucially relies on the property of associativity saying that the terms with "w" can be canceled out by the fraction. But such a cancellation requires the coefficients of the randomness to be the same, which our construction cannot achieve (this is exactly why the two instances of our construction must be orthogonal with each other). Therefore, we choose to use DPVS, which satisfies our "orthogonal" requirement well, to simulate our composite-order scheme. More specifically, we first use DPVS to simulate our composite-order scheme into prime-order scheme, and relies on decisional subspace (DS) assumption [25, 26], which is further based on SXDH assumption, to prove the security.

1.3.4. IP-ME with MSP Auth. We then upgrade IP-ME to IP-ME with MSP Auth, where the second layer is changed to ABE for MSP. One notable point is that for ABE, there are two types according to where the policy embedded, Ciphertext-Policy ABE (CP-ABE) and key-policy ABE (KP-ABE). To transform an ABE to ABS, only CP-ABE is feasible [27–30]. This is determined by the functionalities of CP-ABE and ABS. However, to the best of our knowledge, based on the framework in [11, 12], there is no predicate encoding of CP-ABE for MSP that the encoding of sk is random. Thus, we cannot obtain an ABS for MSP from a CP-ABE when following the modular framework [11, 12]. Lack of the second layer Signature, it seems that our IP-ME with MSP Auth scheme has to be terminated. Fortunately, we notice that the "non-random" of sk of the second layer ABE won't break the authenticity (also unforgeability) property, since the authenticity is derived from the security of ABE.

For the concrete constructions, we also start from the composite-order groups, and simulate it into prime-order groups. As our IP-ME, we first select our basic constructions for each layer. Here, for the first layer, the basic construction is exactly the basic IPE in our IP-ME, and for the second layer, the basic construction is an ABE scheme for MSP, which can also be obtained from the study of Chen et al. [11] and Wee [12]. The ABE for MSP is as follows:

$$\begin{aligned}
\mathsf{mpk} &= (\mathbb{G}, g^{w_1}, \ldots, g^{w_n}, g^{v}, e(g,h)^{\alpha}) \\
\mathsf{msk} &= (w_1, \ldots, w_n, v, \alpha) \\
\mathsf{sk} &= (K_0 = h^{r}, K_1 = h^{rw_1}, \ldots, K_n = h^{rw_n}, K_{n+1} = h^{\alpha + rv}) \\
\mathsf{ct} &= \left(C_0 = g^{s}, C_1 = g^{s(w_1+v_1)}, \ldots, C_n = g^{s(w_n+v_n)}, C' = e(g,h)^{s\alpha} \cdot m\right) \\
\mathsf{Dec}: m &= C' \Big/ \left(\left(e(C_0, K_1) \cdot \prod_j e\left(C_0, K_j\right)^{\eta_j} \right) \Big/ \prod_j e\left(C_j, K_0\right)^{\eta_j} \right),
\end{aligned} \tag{3}$$

where $v_i = \mathbb{M}_i \begin{pmatrix} v \\ \mathbf{u} \end{pmatrix}$, $\mathbf{u} \leftarrow_R \mathbb{Z}_p^{n'-1}$.

For composite-order version, we can adopt similar idea of our IP-ME to obtain the final construction. Specifically, we use an extra α_2 as our IP-ME, to combine the two instances validly. But different from our IP-ME, where we attach the entire α_2 to only one component in $\mathsf{dk}_{\mathbb{S}}$, we secretly share α_2 as v, and attach each share α_{2_i} to the corresponding component in $\mathsf{dk}_{\mathbb{S}}$, so that we can leverage the reconstruction process for v to reconstruct α_2 too. Then, we adopt the same technique as used in our IP-ME to simulate our IP-ME with MSP Auth in composite-order groups into one in prime-order groups.

1.4. Related Works. The first modular ME scheme is proposed by Ateniese et al. [1], and it is constructed from functional encryption (FE), signature, and noninteractive zero-knowledge proofs (NIZK), in a black-box manner. Ateniese et al. [1] also present an IB-ME scheme based on BDH assumption, whose structure is more direct than the proposed modular ME, but is in the random oracle model. In the journal version of Ateniese et al.'s [31] work, they show several other theoretical constructions of ME. Subsequently, Francati et al. [3] present an IB-ME scheme without random oracle and achieving enhanced privacy, which is constructed from reusable computational extractors, Signature and NIZK, but is based on q-type assumption. Then, Chen et al. [4] present the first IB-ME scheme based on standard assumption and in the standard model. Their scheme is directly derived from a two-layer structure of anonymous IBE-based on SXDH assumption. Recently, Francati et al. [32] present the first ME scheme that supports general policies from LWE at the price of having security only in case of a mismatch.

Following the study by Ateniese et al. [1], Xu et al. [33] present a new primitive called matchmaking attribute-based encryption (MABE), which offers secure fine-grained bilateral access control, but different from ME, their MABE seems to only hide the challenge m_0 and m_1, thus it does not provide anonymity. Subsequently, to tackle the issue in ME and MABE that the data decryption process costs a lot, which restricts them to be applied in resource-constrained IoT devices, Xu et al. [34] introduce another new primitive

called lightweight matchmaking encryption (LME) and give a concrete construction.

2. Preliminaries

2.1. Notations. We use $\leftarrow_R$ to denote random sampling, and use PPT to denote probabilistic polynomial time. We use negl to denote a negligible function in security parameter λ. And we use boldface uppercase letter to denote matrix, use boldface lowercase letter to denote vector. We use $||$ to denote concatenation, and use $\langle \cdot, \cdot \rangle$ to denote inner product.

2.2. Dual Pairing Vector Spaces. In cryptography, dual pairing vector spaces mainly relates to the algorithm $\mathsf{Dual}(\mathbb{Z}_p^n)$ as follows [7, 23, 25]:

(i) Sample random bases $\mathbb{D} := (\mathbf{d}_1, \ldots, \mathbf{d}_n)$ and $\mathbb{D}^* := (\mathbf{d}_1^*, \ldots, \mathbf{d}_n^*)$ over $\mathbb{Z}_p$, where p is a prime.
(ii) Output $\mathbb{D}$ and $\mathbb{D}^*$.

And such bases subject to the constraint, which is called "dual orthonormal", as follows:

$$\left\langle \mathbf{d}_i, \mathbf{d}_j^* \right\rangle = 0 \bmod q, \tag{4}$$

whenever $i \neq j$, and

$$\langle \mathbf{d}_i, \mathbf{d}_i^* \rangle = \phi \bmod q, \tag{5}$$

for all i, where ϕ is a random element over $\mathbb{Z}_p$.

Then let $e: G \times H \to G_T$ be a nondegenerated asymmetric bilinear group mapping generated from group generator $\mathcal{G}(1^\lambda)$, where G, H, and G_T are of prime order p. We have

$$e\left(g^{\mathbf{d}_i}, h^{\mathbf{d}_j^*}\right) = 1_T, \tag{6}$$

whenever $i \neq j$.

2.3. Assumptions

Definition 1. (Subgroup decision problem). [13, 35] Let $e: G \times H \to G_T$ be a nondegenerated asymmetric bilinear group mapping generated from group generator $\mathcal{G}(1^\lambda)$, where G, H, and G_T are of order $N = p_1 p_2 p_3$, and p_1, p_2, p_3 are primes. For $i, j \in [3], i \neq j$, let G_{p_i} denote the corresponding subgroup whose order is p_i, and $G_{p_{ij}}$ denote the corresponding subgroup whose order is $p_i p_j$. Let g_i denote the generator in subgroup G_{p_i}, and g_{ij} denote the generator (of arbitrary choice) in subgroup $G_{p_i p_j}$. Similar for group H.

Subgroup decision problem says that given $\mathbb{G} = (N, G, H, G_T, e)$, (g_1, g_2, g_3), and (h_1, h_{12}, h_3), for any PPT adversary $\mathcal{A}$, distinguishing $T_1 \leftarrow_R G_{p_1}$ and $T_2 \leftarrow_R G_{p_1 p_2}$ is hard.

In math language, it says that

$$\begin{aligned} \mathsf{Adv}_{\mathcal{A}}^{SD}(\lambda) :=\ & |\mathsf{Pr}[\mathcal{A}(\mathbb{G}, g_1, g_2, g_3, h_1, h_{12}, h_3, T_1) = 1] \\ & - \mathsf{Pr}[\mathcal{A}(\mathbb{G}, g_1, g_2, g_3, h_1, h_{12}, h_3, T_2) \\ & = 1]| \leq |\mathsf{negl}(\lambda). \end{aligned} \tag{7}$$

Remark 1. The problem also holds when the subscripts are permuted.

Remark 2. We would like to explain that when writing g_{12} as $g_{12} = g_1^{\gamma_1} \cdot g_2^{\gamma_2}$, γ_1, γ_2 should be restricted to $\gamma_1 \leftarrow_R \mathbb{Z}_N / \{k_1 \cdot p_1\}_{k_1 \in [p_2 p_3]}, \gamma_2 \leftarrow_R \mathbb{Z}_N / \{k_2 \cdot p_2\}_{k_2 \in [p_1 p_3]}$. This will lead to a negligible difference of $\frac{1}{p_1} + \frac{1}{p_2}$. For simplicity, we omit this negligible probability below, and simply write it as $g_{12} = g_1^{\gamma_1} \cdot g_2^{\gamma_2}$, where $\gamma_1, \gamma_2 \leftarrow_R \mathbb{Z}_N$.

Definition 2. (Decisional subspace problem). [12, 13] Let $e: G \times H \to G_T$ be a nondegenerated asymmetric bilinear group mapping generated from group generator $\mathcal{G}(1^\lambda)$, where G, H, and G_T are of prime order p. Let $(\mathbb{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_n), \mathbb{D}^* = (\mathbf{d}_1^*, \ldots, \mathbf{d}_n^*)) \leftarrow_R \mathsf{Dual}(\mathbb{Z}_p^n)$ be two random bases that are dual orthonormal. Pick $\tau_1, \tau_2, \mu_1, \mu_2 \leftarrow_R \mathbb{Z}_p$.

Decisional subspace problem in G (DS1) says that, given

$$\begin{aligned} \mathbb{G} = (& p, G, H, G_T, e, g, h), \\ & \left(h^{\mathbf{d}_1^*}, h^{\mathbf{d}_2^*}, \ldots, h^{\mathbf{d}_k^*}, h^{\mathbf{d}_{2k+1}^*}, \ldots, h^{\mathbf{d}_n^*}\right), \\ & \left(g^{\mathbf{d}_1}, \ldots, g^{\mathbf{d}_n}\right), \\ & \left(h^{\mu_1 \mathbf{d}_1^* + \mu_2 \mathbf{d}_{k+1}^*}, h^{\mu_1 \mathbf{d}_2^* + \mu_2 \mathbf{d}_{k+2}^*}, \ldots, h^{\mu_1 \mathbf{d}_k^* + \mu_2 \mathbf{d}_{2k}^*}\right), \\ & \mu_2, \end{aligned} \tag{8}$$

where k and n are positive integers satisfying $2k \leq n$, for any PPT adversary $\mathcal{A}$, distinguishing $(V_1 = g^{\tau_1 \mathbf{d}_1}, \ldots, V_k = g^{\tau_1 \mathbf{d}_k})$ and $(W_1 = g^{\tau_1 \mathbf{d}_1 + \tau_2 \mathbf{d}_{k+1}}, \ldots, W_k = g^{\tau_1 \mathbf{d}_k + \tau_2 \mathbf{d}_{2k}})$ is hard.

In math language, it says that

$$\begin{aligned} \mathsf{Adv}_{\mathcal{A}}^{DS1}(\lambda) :=& |\mathsf{Pr}[\mathcal{A}(\mathcal{D}, (V_1, \ldots, V_k)) = 1] \\ & - \mathsf{Pr}[\mathcal{A}(\mathcal{D}, (W_1, \ldots, W_k)) = 1]| \leq \mathsf{negl}(\lambda), \end{aligned} \tag{9}$$

where

$$\begin{aligned} \mathcal{D} = \big(& \mathbb{G}, \left(h^{\mathbf{d}_1^*}, h^{\mathbf{d}_2^*}, \ldots, h^{\mathbf{d}_k^*}, h^{\mathbf{d}_{2k+1}^*}, \ldots, h^{\mathbf{d}_n^*}\right), \left(g^{\mathbf{d}_1}, \ldots, g^{\mathbf{d}_n}\right), \\ & \left(h^{\mu_1 \mathbf{d}_1^* + \mu_2 \mathbf{d}_{k+1}^*}, h^{\mu_1 \mathbf{d}_2^* + \mu_2 \mathbf{d}_{k+2}^*}, \ldots, h^{\mu_1 \mathbf{d}_k^* + \mu_2 \mathbf{d}_{2k}^*}\right), \mu_2\big). \end{aligned} \tag{10}$$

Remark 3. Decisional subspace problem in H (DS2) is almost the same as decisional subspace problem in G, except the roles of G and H are exchanged.

Remark 4. Decisional subspace problem can be tightly reduced to symmetric external Diffie–Hellman problem in each group [25].

2.4. Matchmaking Encryption. This section is mainly modified from [1].

2.4.1. Syntax. An ME consists of the following polynomial-time algorithms, all the algorithms are probabilistic except Dec, which is deterministic:

$\text{Game}^{\text{Priv}}_{\Pi,\ A}(\lambda)$	$\text{Game}^{\text{Auth}}_{\Pi,\ A}(\lambda)$
$(\text{mpk}, \text{kpol}, \text{msk}) \leftarrow_R \text{Setup}(1^\lambda)$	$(\text{mpk}, \text{kpol}, \text{msk}) \leftarrow_R \text{Setup}(1^\lambda)$
$(m_0,\ m_1,\ \mathbb{R}_0,\ \mathbb{R}_1,\ \sigma_0,\ \sigma_1,\ st) \leftarrow_R A_1^{O_1,O_2,O_3}(\lambda, \text{mpk})$	$(ct,\ \rho,\ \mathbb{S}) \leftarrow_R A^{O_1,O_2,O_3}(\lambda, \text{mpk})$
$\beta \leftarrow_R \{0, 1\}$	$dk_\rho \leftarrow \text{RKGen}(\text{mpk}, \text{msk}, \rho)$
$ek_{\sigma_\beta} \leftarrow \text{SKGen}(\text{mpk}, \text{msk}, \sigma_\beta)$	$dk_{\mathbb{S}} \leftarrow \text{PolGen}(\text{mpk}, \text{kpol}, \mathbb{S})$
$\text{ct}_\beta \leftarrow \text{Enc}(\text{mpk}, ek_{\sigma_\beta}, \mathbb{R}_\beta, m_\beta)$	$m = \text{Dec}(\text{mpk}, dk_\rho,\ dk_{\mathbb{S}},\ ct)$
$\beta' \leftarrow A_1^{O_1,O_2,O_3}(\lambda,\ ct_\beta,\ st)$	If $\forall\sigma \in Q_{O_1}$: (σ mismatches $\mathbb{S}$) $\wedge$ ($m \neq \perp$) return 1
If ($\beta' = \beta$) return 1	Else return 0
Else return 0	

FIGURE 2: Games for privacy and authenticity of ME; Oracles $\mathcal{O}_1, \mathcal{O}_2, \mathcal{O}_3$ are implemented by $\mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \cdot), \mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \cdot), \mathsf{PolGen}(\mathsf{mpk}, \mathsf{kpol}, \cdot)$.

(i) $\mathsf{Setup}(1^\lambda)$: Take as input the security parameter λ, then output the master public key mpk, the master policy key kpol and the master secret key msk.

(ii) $\mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma)$: Take as input the master public key mpk, the master secret key msk, and the attributes σ, then output a secret encryption key ek_σ associated with σ for the sender.

(iii) $\mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho)$: Take as input the master pubic key mpk, the master secret key msk, and the attributes ρ, then output a secret decryption key dk_ρ associated with ρ for the receiver.

(iv) $\mathsf{PolGen}(\mathsf{mpk}, \mathsf{kpol}, \mathbb{S})$: Take as input the master public key, the master policy key kpol, and the policy $\mathbb{S}$, then output a secret decryption key $\mathsf{dk}_\mathbb{S}$ for the receiver.

(v) $\mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathbb{R}, m)$: Take as input the master public key mpk, the secret encryption key ek_σ, the policy $\mathbb{R}$ and the message m, then output a ciphertext ct associated with σ and $\mathbb{R}$.

(vi) $\mathsf{Dec}(\mathsf{mpk}, \mathsf{dk}_\rho, \mathsf{dk}_\mathbb{S}, \mathsf{ct})$: Take as input the master public key mpk, the secret decryption key dk_ρ, the secret decryption key $\mathsf{dk}_\mathbb{S}$ and the ciphertext ct, then output either a message m or $\perp$.

Definition 3. (Correctness of ME). We say an ME scheme is correct, if we have

$$\Pr\left[\mathsf{Dec} = m \;\middle|\; \begin{array}{l} \mathsf{mpk}, \mathsf{kpol}, \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ek}_\sigma \leftarrow \mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma) \\ \mathsf{dk}_\rho \leftarrow \mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho) \\ \mathsf{dk}_\mathbb{S} \leftarrow \mathsf{PolGen}(\mathsf{mpk}, \mathsf{kpol}, \mathbb{S}) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathbb{R}, m) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda), \tag{11}$$

whenever σ matches $\mathbb{S}$ and ρ matches $\mathbb{R}$, and otherwise

$$\Pr\left[\mathsf{Dec} = \perp \;\middle|\; \begin{array}{l} \mathsf{mpk}, \mathsf{kpol}, \mathsf{msk} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{ek}_\sigma \leftarrow \mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma) \\ \mathsf{dk}_\rho \leftarrow \mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho) \\ \mathsf{dk}_\mathbb{S} \leftarrow \mathsf{PolGen}(\mathsf{mpk}, \mathsf{kpol}, \mathbb{S}) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathbb{R}, m) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda). \tag{12}$$

2.4.2. Security

Definition 4. (Weak privacy of ME). We say an ME scheme Π satisfies weak privacy, if for any valid PPT adversary $\mathcal{A}$, we have

$$\left|\Pr\left[\mathsf{Game}^{Priv}_{\Pi,\mathcal{A}}(\lambda) = 1\right] - \frac{1}{2}\right| \leq \mathsf{negl}(\lambda), \tag{13}$$

where $\mathsf{Game}^{Priv}_{\Pi,\mathcal{A}}(\lambda)$ is defined in Figure 2. Adversary $\mathcal{A}$ is called valid if $\forall\rho \in \mathcal{Q}_{\mathcal{O}_2}$, it satisfies the following condition:

(i) **(Mismatch Condition)**. ρ does not match $\mathbb{R}_0$ and ρ does not match $\mathbb{R}_1$.

Definition 5. (Authenticity of ME). We say an ME scheme Π satisfies authenticity, if for any PPT adversary $\mathcal{A}$, we have

$$\Pr\left[\mathsf{Game}^{\text{Auth}}_{\Pi,\mathcal{A}}(\lambda) = 1\right] \leq \mathsf{negl}(\lambda), \tag{14}$$

where $\mathsf{Game}^{\text{Auth}}_{\Pi,\mathcal{A}}(\lambda)$ is defined in Figure 2.

Definition 6. (Weak security of ME). We say that an ME scheme Π satisfies weak security, if it satisfies weak privacy and authenticity.

3. Our IP-ME in Composite-Order Groups

We first present an IP-ME in composite-order groups, whose order is a product of three primes. And without loss of generality, we assume $y_1 = 1$ in $\mathbf{y}$.

3.1. Construction Π_{IP_C}

(i) $\mathsf{Setup}(1^\lambda)$:

(1) Run the group generator $\mathbb{G} = (N = p_1p_2p_3, G, H, G_T, e, g, h) \leftarrow \mathcal{G}(1^\lambda)$, then output $\mathsf{pp} = \mathbb{G}$.

(2) Pick $w_1, \ldots, w_{n+1}, w'_1, \ldots, w'_{n+1}, \alpha_1, \alpha_2 \leftarrow_R \mathbb{Z}_N$, then output

$$\mathsf{mpk} = \left(g_1, g_1^{w'_1}, \ldots, g_1^{w'_{n+1}}, e(g_1, h_1)^{\alpha_1}\right). \tag{15}$$

(3) Store secretly

$$\mathsf{msk} = \left(h_1, h_3, g_3, w_1, \ldots, w_{n+1}, w'_1, \ldots, w'_{n+1}, \alpha_1, \alpha_2\right). \tag{16}$$

(ii) $\mathsf{SKGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{y})$:

(1) Pick $s_2 \leftarrow_R \mathbb{Z}_N$, then output

$$\mathsf{ek}_\mathbf{y} = \left(K_0^{(2)} = g_3^{s_2}, K_1^{(2)} = \prod_i g_3^{s_2 w_{i+1} y_i}\right). \tag{17}$$

(iii) $\mathsf{RKGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{v})$:

(1) Pick $r_1 \leftarrow_R \mathbb{Z}_N$, then output

$$\mathsf{dk}_\mathbf{v} = \left(K_0^{(1)} = h_1^{r_1}, K_1^{(1)} = h_1^{\alpha_1} \cdot h_3^{\alpha_2} \cdot \prod_i h_1^{r_1 w'_{i+1} v_i}\right). \tag{18}$$

(iv) $\mathsf{PolGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{t})$:

(1) Pick $r_2 \leftarrow_R \mathbb{Z}_N$, then output

$$\begin{aligned}\mathsf{dk}_\mathbf{t} = \Big(&K_0^{(3)} = h_3^{r_2},\\ &K_1^{(3)} = h_3^{r_2 w_1 t_1} h_3^{r_2 w_2} \cdot h_3^{-\alpha_2},\\ &K_2^{(3)} = h_3^{r_2 w_1 t_2} h_3^{r_2 w_3},\\ &\ldots,\\ &K_n^{(3)} = h_3^{r_2 w_1 t_n} h_3^{r_2 w_{n+1}}\Big).\end{aligned} \tag{19}$$

(v) $\mathsf{Enc}(\mathsf{pp}, \mathsf{mpk}, \mathsf{ek}_\mathbf{y}, \mathbf{x}, m)$:

(1) Pick $s_1 \leftarrow_R \mathbb{Z}_N$, then output

$$\begin{aligned}\mathsf{ct} = \Big(&C_0 = g_1^{s_1} \cdot K_0^{(2)},\\ &C_1 = g_1^{s_1 w'_1 x_1} g_1^{s_1 w'_2} \cdot K_1^{(2)},\\ &C_2 = g_1^{s_1 w'_1 x_2} g_1^{s_1 w'_3},\\ &\ldots,\\ &C_n = g_1^{s_1 w'_1 x_n} g_1^{s_1 w'_{n+1}},\\ &C_{n+1} = e(g_1, h_1)^{\alpha_1 s_1} \cdot m\Big).\end{aligned} \tag{20}$$

(vi) $\mathsf{Dec}(\mathsf{pp}, \mathsf{dk}_\mathbf{v}, \mathsf{dk}_\mathbf{t}, \mathsf{ct})$:

(1) Compute

$$\begin{aligned}m = C_{n+1} / \Big(&\Big(e\Big(C_0, K_1^{(1)}\Big) \cdot e\Big(C_0, K_1^{(3)y_1}\Big) \cdots e\Big(C_0, K_n^{(3)y_n}\Big)\Big) /\\ &\Big(e\Big(C_1^{v_1}, K_0^{(1)}\Big) \cdots e\Big(C_n^{v_n}, K_0^{(1)}\Big) \cdot e\Big(C_1, K_0^{(3)}\Big)\Big)\Big).\end{aligned} \tag{21}$$

3.1.1. Correctness. The correctness follows from

$$\begin{aligned}&\Big(e\Big(C_0, K_1^{(1)}\Big) \cdot e\Big(C_0, K_1^{(3)y_1}\Big) \cdots e\Big(C_0, K_n^{(3)y_n}\Big)\Big) /\\ &\Big(e\Big(C_1^{v_1}, K_0^{(1)}\Big) \cdots e\Big(C_n^{v_n}, K_0^{(1)}\Big) \cdot e\Big(C_1, K_0^{(3)}\Big)\Big)\\ =\ &\left(e\left(g_1^{s_1} \cdot g_3^{s_2}, h_1^{\alpha_1} \cdot h_3^{\alpha_2} \cdot \prod_i h_1^{r_1 w'_{i+1} v_i}\right) \cdot e\left(g_1^{s_1} \cdot g_3^{s_2}, h_3^{r_2 w_1 t_1 y_1} h_3^{r_2 w_2 y_1} h_3^{-\alpha_2 y_1}\right)\right)\\ \cdot\ &\prod_{i=2,\ldots,n} e\left(g_1^{s_1} \cdot g_3^{s_2}, h_3^{r_2 w_1 t_i y_i} h_3^{r_2 w_{i+1} y_i}\right) / \left(e\left(g_1^{s_1 w'_1 x_1 v_1} g_1^{s_1 w'_2 v_1} \cdot \prod_i g_3^{s_2 w_{i+1} y_i v_1}, h_1^{r_1}\right) \cdot\right.\\ &\left.\prod_{i=2,\ldots,n} e\left(g_1^{s_1 w'_1 x_i v_i} g_1^{s_1 w'_{i+1} v_i}, h_1^{r_1}\right) \cdot e\left(g_1^{s_1 w'_1 x_1} g_1^{s_1 w'_2} \cdot \prod_i g_3^{s_2 w_{i+1} y_i}, h_3^{r_2}\right)\right.\\ =\ &\left(e\left(g_1^{s_1}, h_1^{\alpha_1} \prod_i h_1^{r_1 w'_{i+1} v_i}\right) \cdot e(g_3^{s_2}, h_3^{\alpha_2}) \cdot e\left(g_3^{s_2}, h_3^{r_2 w_1 t_1 y_1} h_3^{r_2 w_2 y_1} h_3^{-\alpha_2}\right) \cdot\right.\\ &\left.\prod_{i=2,\ldots,n} e\left(g_3^{s_2}, h_3^{r_2 w_1 t_i y_i} h_3^{r_2 w_{i+1} y_i}\right)\right) / \left(e\left(g_1^{s_1 w'_1 x_1 v_1} g_1^{s_1 w'_2 v_1}, h_1^{r_1}\right) \cdot\right.\\ &\left.\prod_{i=2,\ldots,n} e\left(g_1^{s_1 w'_1 x_i v_i} g_1^{s_1 w'_{i+1} v_i}, h_1^{r_1}\right) \cdot e\left(\prod_i g_3^{s_2 w_{i+1} y_i}, h_3^{r_2}\right)\right.\\ =\ &e(g_1^{s_1}, h_1^{\alpha_1}).\end{aligned} \tag{22}$$

Remark 5. When the subscript of product sign is a single i, it refers to $i = 1, \ldots, n$.

3.2. Security Analysis

Theorem 1. *The IP-ME scheme Π_{IP_C} satisfies weak privacy and authenticity under SD assumptions.*

Since the proof is similar to Theorem 4, which follows the dual system encryption methodology (turning the normal ciphertext and secret key into semifunctional forms and leading to unconditionally failed decryption, and achieving attribute-hiding via the attribute-hiding encoding), thus we omit it here.

4. Our IP-ME in Prime-Order Groups

We transform our composite-order IP-ME into prime-order version in this section. With DPVS, our substitutions are as below:

$$g_i \to g^{\mathbf{d}_i},\ h_i \to h^{\mathbf{d}_i^*}. \tag{23}$$

We also assume $y_1 = 1$ in $\mathbf{y}$ without loss of generality.

4.1. Construction Π_{IP_P}

(i) $\mathsf{Setup}(1^\lambda)$:

(1) Run the group generator $\mathbb{G} = (p, G, H, G_T, e, g, h) \leftarrow \mathcal{G}(1^\lambda)$, then output $\mathsf{pp} = \mathbb{G}$.

(2) Sample random dual orthonormal bases $(\mathbb{D}, \mathbb{D}^*) \leftarrow_R \mathsf{Dual}(\mathbb{Z}_p^3)$. Let $\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$ denote the elements of $\mathbb{D}$ and $\mathbf{d}_1^*, \mathbf{d}_2^*, \mathbf{d}_3^*$ denote the elements of $\mathbb{D}^*$. Let $g_T = e(g, h)^{\langle \mathbf{d}_i, \mathbf{d}_i^* \rangle}$.

(3) Pick $w_1, \ldots, w_{n+1}, w_1', \ldots, w_{n+1}', \alpha_1, \alpha_2 \leftarrow_R \mathbb{Z}_p$, then output

$$\mathsf{mpk} = \left(g^{\mathbf{d}_1}, g^{w_1'\mathbf{d}_1}, \ldots, g^{w_{n+1}'\mathbf{d}_1}, e\left(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*}\right)^{\alpha_1}\right). \tag{24}$$

(4) Store secretly

$$\mathsf{msk} = \left(h^{\mathbf{d}_1^*}, h^{\mathbf{d}_3^*}, g^{\mathbf{d}_3}, w_1, \ldots, w_{n+1}, w_1', \ldots, w_{n+1}', \alpha_1, \alpha_2\right). \tag{25}$$

(ii) $\mathsf{SKGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{y})$:

(1) Pick $s_2 \leftarrow_R \mathbb{Z}_N$, then output

$$\mathsf{ek}_\mathbf{y} = \left(K_0^{(2)} = g^{s_2\mathbf{d}_3}, K_1^{(2)} = \prod_i g^{s_2 w_{i+1} y_i \mathbf{d}_3}\right). \tag{26}$$

(iii) $\mathsf{RKGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{v})$:

(1) Pick $r_1 \leftarrow_R \mathbb{Z}_N$, then output

$$\mathsf{dk}_\mathbf{v} = \left(K_0^{(1)} = h^{r_1\mathbf{d}_1^*}, K_1^{(1)} = h^{\alpha_1\mathbf{d}_1^*} \cdot h^{\alpha_2\mathbf{d}_3^*} \cdot \prod_i h^{r_1 w_{i+1}' v_i \mathbf{d}_1^*}\right). \tag{27}$$

(iv) $\mathsf{PolGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{t})$:

(1) Pick $r_2 \leftarrow_R \mathbb{Z}_N$, then output

$$\begin{aligned} \mathsf{dk}_\mathbf{t} = \Big(&K_0^{(3)} = h^{r_2\mathbf{d}_3^*}, \\ &K_1^{(3)} = h^{(r_2 w_1 t_1 + r_2 w_2 - \alpha_2)\mathbf{d}_3^*}, \\ &K_2^{(3)} = h^{(r_2 w_1 t_2 + r_2 w_3)\mathbf{d}_3^*}, \\ &\ldots, \\ &K_n^{(3)} = h^{(r_2 w_1 t_n + r_2 w_{n+1})\mathbf{d}_3^*)}. \end{aligned} \tag{28}$$

(v) $\mathsf{Enc}(\mathsf{pp}, \mathsf{mpk}, \mathsf{ek}_\mathbf{y}, \mathbf{x}, m)$:

(1) Pick $s_1 \leftarrow_R \mathbb{Z}_N$, then output

$$\begin{aligned} \mathsf{ct} = \Big(&C_0 = g^{s_1\mathbf{d}_1} \cdot K_0^{(2)}, \\ &C_1 = g^{\left(s_1 w_1' x_1 + s_1 w_2'\right)\mathbf{d}_1} \cdot K_1^{(2)}, \\ &C_2 = g^{\left(s_1 w_1' x_2 + s_1 w_3'\right)\mathbf{d}_1}, \\ &\ldots, \\ &C_n = g^{\left(s_1 w_1' x_n + s_1 w_{n+1}'\right)\mathbf{d}_1}, \\ &C_{n+1} = e\left(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*}\right)^{\alpha_1 s_1} \cdot m). \end{aligned} \tag{29}$$

(vi) $\mathsf{Dec}(\mathsf{pp}, \mathsf{dk}_\mathbf{v}, \mathsf{dk}_\mathbf{t}, \mathsf{ct})$:

(1) Compute

$$\begin{aligned} m = C_{n+1} / \Big(&\left(e\left(C_0, K_1^{(1)}\right) \cdot e\left(C_0, K_1^{(3)y_1}\right) \cdots e\left(C_0, K_n^{(3)y_n}\right)\right) / \\ &\left(e\left(C_1^{v_1}, K_0^{(1)}\right) \cdots e\left(C_n^{v_n}, K_0^{(1)}\right) \cdot e\left(C_1, K_0^{(3)}\right)\right)\Big). \end{aligned} \tag{30}$$

4.1.1. Correctness. The correctness follows from

$$
\begin{aligned}
&\left(e\left(C_0, K_1^{(1)}\right)\cdot e\left(C_0, K_1^{(3)y_1}\right)\cdots e\left(C_0, K_n^{(3)y_n}\right)\right)/\left(e\left(C_1^{v_1}, K_0^{(1)}\right)\cdots e\left(C_n^{v_n}, K_0^{(1)}\right)\cdot e\left(C_1, K_0^{(3)}\right)\right)\\
=&\left(e\left(g^{s_1\mathbf{d}_1+s_2\mathbf{d}_3}, h^{\alpha_1\mathbf{d}_1^*}\cdot h^{\alpha_2\mathbf{d}_3^*}\cdot\prod_i h^{r_1w'_{i+1}v_i\mathbf{d}_1^*}\right)\cdot e\left(g^{s_1\mathbf{d}_1+s_2\mathbf{d}_3}, h^{y_1(r_2w_1t_1+r_2w_2-\alpha_2)\mathbf{d}_3^*}\right)\cdot\right.\\
&\left.\prod_{i=2,\ldots,n} e\left(g^{s_1\mathbf{d}_1+s_2\mathbf{d}_3}, h^{y_i(r_2w_1t_i+r_2w_{i+1})\mathbf{d}_3^{*)}}\right)\right)/\left(e\left(g^{v_1\left(s_1w'_1x_1+s_1w'_2\right)\mathbf{d}_1+v_1\sum_i s_2w_{i+1}y_i\mathbf{d}_3}, h^{r_1\mathbf{d}_1^*}\right)\cdot\right.\\
&\left.\prod_{i=2,\ldots,n} e\left(g^{v_i\left(s_1w'_1x_i+s_1w'_{i+1}\right)\mathbf{d}_1}, h^{r_1\mathbf{d}_1^*}\right)\cdot e\left(g^{\left(s_1w'_1x_1+s_1w'_2\right)\mathbf{d}_1+\sum_i s_2w_{i+1}y_i\mathbf{d}_3}, h^{r_2\mathbf{d}_3^*}\right)\right)\\
=&\left(g_T^{\left(s_1\alpha_1+s_1\sum_i r_1w'_{i+1}v_i\right)}\cdot g_T^{s_2\alpha_2}\cdot g_T^{s_2(y_1r_2w_1t_1+y_1r_2w_2)-s_2\alpha_2}\cdot g_T^{\sum_{i=2,\ldots,n}^{s_2} y_i(r_2w_1t_i+r_2w_{i+1})}\right)\\
/&\left(g_T^{v_1\left(s_1w'_1x_1+s_1w'_2\right)r_1}\cdot g_T^{\sum_{i=2,\ldots,n}^{r_1} v_i\left(s_1w'_1x_i+s_1w'_{i+1}\right)}\cdot g_T^{r_2\sum_i s_2w_{i+1}y_i}\right)\\
=&\, g_T^{s_1\alpha_1}\\
=&\, e\left(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*}\right)^{s_1\alpha_1}.
\end{aligned}
\tag{31}
$$

Remark 6. When the subscripts of product sign and summation sign are a single i, it refers to $i=1,\ldots,n$.

4.2. Security Analysis

Theorem 2. *The IP-ME scheme Π_{IP_P} satisfies weak privacy and authenticity under DS assumptions.*

Since the proof is similar to Theorem 4, which follows the dual system encryption methodology (turning the normal ciphertext and secret key in to semifunctional forms and leading to unconditionally failed decryption, and achieving attribute-hiding via the attribute-hiding encoding), thus we omit it here.

5. Our IP-ME with MSP Auth in Composite-Order Groups

In this section we present our IP-ME with MSP Auth in composite-order groups, whose order is a product of three primes. And note that here, we assume sender's attributes $\mathbf{y}\in\{0,1\}^n$.

5.1. Construction Π_{IPMSP_C}

(i) $\mathsf{Setup}(1^\lambda)$:

(1) Run the group generator $\mathbb{G}=(N=p_1p_2p_3, G, H, G_T, e, g, h)\leftarrow\mathcal{G}(1^\lambda)$, then output $\mathsf{pp}=\mathbb{G}$.

(2) Pick $w_1,\ldots,w_n,w'_1,\ldots,w'_n,w'_{n+1},v,\alpha_1,\alpha_2\leftarrow_R\mathbb{Z}_N$, then output

$$
\mathsf{mpk}=\left(g_1, g_1^{w'_1},\ldots,g_1^{w'_n}, g_1^{w'_{n+1}}, e(g_1,h_1)^{\alpha_1}\right).
\tag{32}
$$

(3) Store

$$
\mathsf{msk}=\left(h_1,h_3,g_3,w_1,\ldots,w_n,w'_1,\ldots,w'_n,w'_{n+1},v,\alpha_1,\alpha_2\right).
\tag{33}
$$

(ii) $\mathsf{SKGen}(\mathsf{pp},\mathsf{msk},\mathbf{y})$:

(1) Pick $s_2\leftarrow_R\mathbb{Z}_N$, then output

$$
\begin{aligned}
\mathsf{ek}_{\mathbf{y}}=\Big(&K_0^{(2)}=g_3^{s_2},\\
&K_1^{(2)}=g_3^{s_2w_1y_1},\\
&\ldots,\\
&K_n^{(2)}=g_3^{s_2w_ny_n},\\
&K_{n+1}^{(2)}=g_3^{s_2v)}.
\end{aligned}
\tag{34}
$$

(iii) $\mathsf{RKGen}(\mathsf{pp},\mathsf{msk},\mathbf{t})$:

(1) Pick $r_1\leftarrow_R\mathbb{Z}_N$, then output

$$
\mathsf{dk}_{\mathbf{t}}=\left(K_0^{(1)}=h_1^{r_1}, K_1^{(1)}=h_1^{\alpha_1}\cdot h_3^{\alpha_2}\cdot\prod_i h_1^{r_1w'_{i+1}t_i}\right).
\tag{35}
$$

(iv) $\mathsf{PolGen}(\mathsf{pp},\mathsf{msk},\mathbb{M}_{n\times n'})$:

(1) Pick $r_2 \leftarrow_R \mathbb{Z}_N, \mathbf{u}_1, \mathbf{u}_2 \leftarrow_R \mathbb{Z}_N^{n'-1}$, and set $v_i = \mathbb{M}_i \begin{pmatrix} v \\ \mathbf{u}_1 \end{pmatrix}, \alpha_{2_i} = \mathbb{M}_i \begin{pmatrix} \alpha_2 \\ \mathbf{u}_2 \end{pmatrix}$. Then output

$$\begin{aligned} \mathsf{dk}_{\mathbb{M}} = \Big(& K_0^{(3)} = h_3^{r_2}, \\ & K_1^{(3)} = h_3^{r_2 w_1 + r_2 v_1} h_3^{\alpha_{2_1}}, \\ & \ldots, \\ & K_n^{(3)} = h_3^{r_2 w_n + r_2 v_n} h_3^{\alpha_{2_n)}}. \end{aligned} \tag{36}$$

(v) $\mathsf{Enc}(\mathsf{pp}, \mathsf{mpk}, \mathsf{ek}_{\mathbf{y}}, \mathbf{x}, m)$:

(1) Pick $s_1 \leftarrow_R \mathbb{Z}_N$, then output

$$\begin{aligned} \mathsf{ct} = \Big(& C_0 = g_1^{s_1} \cdot K_0^{(2)}, \\ & C_1 = g_1^{s_1 w_1' x_1} g_1^{s_1 w_2'} \cdot K_1^{(2)}, \\ & \ldots, \\ & C_n = g_1^{s_1 w_1' x_n} g_1^{s_1 w_{n+1}'} \cdot K_n^{(2)}, \\ & C_{n+1} = K_{n+1}^{(2)}, \\ & C_{n+2} = e(g_1, h_1)^{\alpha_1 s_1} \cdot m). \end{aligned} \tag{37}$$

(vi) $\mathsf{Dec}(\mathsf{pp}, \mathsf{dk}_{\mathbf{t}}, \mathsf{dk}_{\mathbb{M}}, \mathsf{ct})$:

(1) Compute $\eta_j \in \mathbb{Z}_N$, such that

$$\sum_j \eta_j \mathbb{M}_{j|y_j=1|} = (1, 0, \ldots, 0). \tag{38}$$

Then

$$m = C_{n+2} / \left(\left(e\left(C_0, K_1^{(1)}\right) \cdot e\left(C_{n+1}, K_0^{(3)}\right) \prod_j e\left(C_j, K_0^{(3)}\right)^{\eta_j} \right) / \left(\prod_i e\left(C_i^{t_i}, K_0^{(1)}\right) \cdot \prod_j e\left(C_0, K_j^{(3)}\right)^{\eta_j} \right) \right). \tag{39}$$

5.1.1. Correctness. The correctness follows from

$$\begin{aligned} & \left(e\left(C_0, K_1^{(1)}\right) \cdot e\left(C_{n+1}, K_0^{(3)}\right) \cdot \prod_j e\left(C_j, K_0^{(3)}\right)^{\eta_j} \right) / \left(\prod_i e\left(C_i^{t_i}, K_0^{(1)}\right) \cdot \prod_j e\left(C_0, K_j^{(3)}\right)^{\eta_j} \right) \\ & = \left(e\left(g_1^{s_1} \cdot g_3^{s_2}, h_1^{\alpha_1} \cdot h_3^{\alpha_2} \cdot \prod_i h_1^{r_1 w_{i+1}' t_i} \right) \cdot e(g_3^{s_2 v}, h_3^{r_2}) \cdot \prod_j e\left(g_1^{s_1 w_1' x_j} g_1^{s_1 w_{j+1}'} \cdot g_3^{s_2 w_j y_j}, h_3^{r_2} \right)^{\eta_j} \right) \\ & / \left(\prod_i e\left(g_1^{s_1 w_1' x_i t_i} g_1^{s_1 w_{i+1}' t_i} \cdot g_3^{s_2 w_i y_i t_i}, h_1^{r_1} \right) \cdot \prod_j e\left(g_1^{s_1} \cdot g_3^{s_2}, h_3^{r_2 w_j + r_2 v_j} h_3^{\alpha_{2_j}} \right)^{\eta_j} \right) \\ & = \left(e\left(g_1^{s_1}, h_1^{\alpha_1} \cdot \prod_i h_1^{r_1 w_{i+1}' t_i} \right) \cdot e(g_3^{s_2}, h_3^{\alpha_2}) \cdot e(g_3^{s_2 v}, h_3^{r_2}) \cdot \prod_j e\left(g_3^{s_2 w_j}, h_3^{r_2} \right)^{\eta_j} \right) \\ & / \left(\prod_i e\left(g_1^{s_1 w_1' x_i t_i} g_1^{s_1 w_{i+1}' t_i}, h_1^{r_1} \right) \cdot \prod_j e\left(g_3^{s_2}, h_3^{r_2 w_j + r_2 v_j} h_3^{\alpha_{2_j}} \right)^{\eta_j} \right) \\ & = e(g_1, h_1)^{\alpha_1 s_1}. \end{aligned} \tag{40}$$

Remark 7. When the subscript of product sign is a single i, it refers to $i = 1, \ldots, n$. And when the subscript of product sign is a single j, it refers to j traversing the set $\{\psi | y_\psi | = 1\}$.

5.2. Security Analysis

Theorem 3. *The IP-ME with MSP Auth scheme Π_{IPMSP_C} satisfies weak privacy and authenticity under SD assumptions.*

Since the proof is similar to Theorem 4, which follows the dual system encryption methodology (turning the normal ciphertext and secret key in to semifunctional forms and leading to unconditionally failed decryption, and achieving attribute-hiding via the attribute-hiding encoding), thus we omit it here.

6. Our IP-ME with MSP Auth in Prime-Order Groups

We also transform our composite-order IP-ME with MSP Auth into prime-order version like in section 4. And we also assume sender's attributes $\mathbf{y} \in \{0, 1\}^n$.

6.1. Construction Π_{IPMSP_P}

(i) $\mathsf{Setup}(1^\lambda)$:

(1) Run the group generator $\mathbb{G} = (p, G, H, G_T, e, g, h) \leftarrow \mathcal{G}(1^\lambda)$, then output $\mathsf{pp} = \mathbb{G}$.

(2) Sample random dual orthonormal bases $(\mathbb{D}, \mathbb{D}^*) \leftarrow_R \mathsf{Dual}(\mathbb{Z}_p^3)$. Let $\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$ denote the elements of $\mathbb{D}$ and $\mathbf{d}_1^*, \mathbf{d}_2^*, \mathbf{d}_3^*$ denote the elements of $\mathbb{D}^*$. Let $g_T = e(g, h)^{\langle \mathbf{d}_i, \mathbf{d}_i^* \rangle}$.

(3) Pick $w_1, \ldots, w_n, w_1', \ldots, w_n', w_{n+1}', v, \alpha_1, \alpha_2 \leftarrow_R \mathbb{Z}_p$, then output

$$\mathsf{mpk} = \left(g^{\mathbf{d}_1}, g^{w_1' \mathbf{d}_1}, \ldots, g^{w_n' \mathbf{d}_1}, g^{w_{n+1}' \mathbf{d}_1}, e\left(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*}\right)^{\alpha_1}\right). \tag{41}$$

(4) Store secretly

$$\mathsf{msk} = \left(h^{\mathbf{d}_1^*}, h^{\mathbf{d}_3^*}, g^{\mathbf{d}_3}, w_1, \ldots, w_n, w_1', \ldots, w_n', w_{n+1}', v, \alpha_1, \alpha_2\right). \tag{42}$$

(ii) $\mathsf{SKGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{y})$:

(1) Pick $s_2 \leftarrow_R \mathbb{Z}_p$, then output

$$\begin{aligned} \mathsf{ek}_{\mathbf{y}} = \Big(& K_0^{(2)} = g^{s_2 \mathbf{d}_3}, \\ & K_1^{(2)} = g^{s_2 w_1 y_1 \mathbf{d}_3}, \\ & \ldots, \\ & K_n^{(2)} = g^{s_2 w_n y_n \mathbf{d}_3}, \\ & K_{n+1}^{(2)} = g^{s_2 v \mathbf{d}_3}). \end{aligned} \tag{43}$$

(iii) $\mathsf{RKGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{t})$:

(1) Pick $r_1 \leftarrow_R \mathbb{Z}_p$, then output

$$\mathsf{dk}_{\mathbf{t}} = \left(K_0^{(1)} = h^{r_1 \mathbf{d}_1^*}, K_1^{(1)} = h^{\alpha_1 \mathbf{d}_1^* + \alpha_2 \mathbf{d}_3^* + \sum_i r_1 w_{i+1}' t_i \mathbf{d}_1^*} \right). \tag{44}$$

(iv) $\mathsf{PolGen}(\mathsf{pp}, \mathsf{msk}, \mathbb{M}_{n \times n'})$:

(1) Pick $r_2 \leftarrow_R \mathbb{Z}_p, \mathbf{u}_1, \mathbf{u}_2 \leftarrow_R \mathbb{Z}_p^{n'-1}$, and set $v_i = \mathbb{M}_i \begin{pmatrix} v \\ \mathbf{u}_1 \end{pmatrix}, \alpha_{2_i} = \mathbb{M}_i \begin{pmatrix} \alpha_2 \\ \mathbf{u}_2 \end{pmatrix}$. Then output

$$\begin{aligned} \mathsf{dk}_{\mathbb{M}} = \Big(& K_0^{(3)} = h^{r_2 \mathbf{d}_3^*}, \\ & K_1^{(3)} = h^{\left(r_2 w_1 + r_2 v_i + \alpha_{2_i}\right) \mathbf{d}_3^*}, \\ & \ldots, \\ & K_n^{(3)} = h^{\left(r_2 w_n + r_2 v_n + \alpha_{2_n}\right) \mathbf{d}_3^*}). \end{aligned} \tag{45}$$

(v) $\mathsf{Enc}(\mathsf{pp}, \mathsf{mpk}, \mathsf{ek}_{\mathbf{y}}, \mathbf{x}, m)$:

(1) Pick $\mathbf{s}_1 \leftarrow_R \mathbb{Z}_p^k$, then output

$$\begin{aligned} \mathsf{ct} = \Big(& C_0 = g^{s_1 \mathbf{d}_1} \cdot K_0^{(2)}, \\ & C_1 = g^{\left(s_1 w_1' x_1 + s_1 w_2'\right) \mathbf{d}_1} \cdot K_1^{(2)}, \\ & \ldots, \\ & C_n = g^{\left(s_1 w_1' x_n + s_1 w_{n+1}'\right) \mathbf{d}_1} \cdot K_n^{(2)}, \\ & C_{n+1} = K_{n+1}^{(2)}, \\ & C_{n+2} = e\left(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*}\right)^{\alpha_1 s_1} \cdot m). \end{aligned} \tag{46}$$

(vi) $\mathsf{Dec}(\mathsf{pp}, \mathsf{dk}_{\mathbf{t}}, \mathsf{dk}_{\mathbb{M}}, \mathsf{ct})$:

(1) Compute $\eta_j \in \mathbb{Z}_p$, such that

$$\sum_j \eta_j \mathbb{M}_{j|y_j|=1} = (1, 0, \ldots, 0). \tag{47}$$

Then

$$m = C_{n+2} / \left(\left(e\left(C_0, K_1^{(1)}\right) \cdot e\left(C_{n+1}, K_0^{(3)}\right) \cdot \prod_j e\left(C_j, K_0^{(3)}\right)^{\eta_j} \right) / \left(\prod_i e\left(C_i^{t_i}, K_0^{(1)}\right) \cdot \prod_j e\left(C_0, K_j^{(3)}\right)^{\eta_j} \right) \right). \tag{48}$$

6.1.1. Correctness. The correctness follows from

$$
\begin{aligned}
&\left(e\left(C_0, K_1^{(1)}\right)\cdot e\left(C_{n+1}, K_0^{(3)}\right)\cdot \prod_j e\left(C_j, K_0^{(3)}\right)^{\eta_j}\right) / \left(\prod_i e\left(C_i^{t_i}, K_0^{(1)}\right)\cdot \prod_j e\left(C_0, K_j^{(3)}\right)^{\eta_j}\right)\\
&= \left(e\left(g^{s_1\mathbf{d}_1+s_2\mathbf{d}_3}, h^{\alpha_1\mathbf{d}_1^*+\alpha_2\mathbf{d}_3^*+\sum_i r_1 w'_{i+1}t_i\mathbf{d}_1^*}\right)\cdot e\left(g^{s_2 v\mathbf{d}_3}, h^{r_2\mathbf{d}_3^*}\right)\cdot \right.\\
&\left.\prod_j e\left(g^{\left(s_1 w'_1 x_j + s_1 w'_{j+1}\right)\mathbf{d}_1+s_2 w_j y_j\mathbf{d}_3}, h^{r_2\mathbf{d}_3^*}\right)^{\eta_j}\right) / \left(\prod_i e\left(g^{\left(s_1 w'_1 x_i t_i + s_1 w'_{i+1} t_i\right)\mathbf{d}_1+s_2 w_i y_i t_i\mathbf{d}_3}, h^{r_1\mathbf{d}_1^*}\right)\cdot \right.\\
&\left.\prod_j e\left(g^{s_1\mathbf{d}_1+s_2\mathbf{d}_3}, h^{\left(r_2 w_j + r_2 v_j + \alpha_{2_j}\right)\mathbf{d}_3^*}\right)^{\eta_j}\right)\\
&= \left(g_T^{s_1\left(\alpha_1+\sum_i r_1 w'_{i+1}t_i\right)}\cdot g_T^{s_2\alpha_2}\cdot g_T^{s_2 v r_2}\cdot g_T^{r_2\sum_j s_2 w_j\eta_j}\right) / \left(g_T^{r_1\sum_i\left(s_1 w'_1 x_i t_i + s_1 w'_{i+1} t_i\right)}\cdot \right.\\
&\left. g_T^{s_2\sum_j\left(r_2 w_j\eta_j + r_2 v_j\eta_j + \alpha_{2_j}\eta_j\right)}\right)\\
&= g_T^{\alpha_1 s_1}\\
&= e\left(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*}\right)^{\alpha_1 s_1}.
\end{aligned}
\tag{49}
$$

Remark 8. When the subscripts of product sign and summation sign are a single i, it refers to $i = 1, \ldots, n$. And when the subscript of product sign and summation sign are a single j, it refers to j traversing the set $\{\psi | y_\psi = 1\}$.

6.2. Security Analysis

Theorem 4. *The scheme Π_{IPMSP_p} satisfies weak privacy and authenticity under DS assumptions.*

6.2.1. Proof of Theorem 4. Proof of Privacy

Theorem 5. *For any* PPT *adversary $\mathcal{A}$, we have*

$$
\begin{aligned}
\mathsf{Adv}_{\mathcal{A}}^{\mathsf{Game}_{\text{priv}}}(\lambda) = \left|\Pr\left[\mathsf{Game}_{\mathcal{A}}^{\text{priv}}(\lambda) = 1\right] - \frac{1}{2}\right| &\le \mathsf{Adv}_{\mathcal{B}_1}^{DS1}(\lambda)\\
&+ D\cdot \mathsf{Adv}_{\mathcal{B}_2}^{DS2}(\lambda) + D\cdot \mathsf{Adv}_{\mathcal{B}_3}^{DS2}(\lambda).
\end{aligned}
\tag{50}
$$

where $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$ are defined in the following lemmas, and without loss of generality, we assume the upper bounds of the number of $\mathsf{dk}_{\mathbf{t}}$ and $\mathsf{dk}_{\mathbb{M}}$ are both equal to D.

Proof. We adopt the dual system encryption methodology to prove weak privacy [13, 36]. Roughly speaking, dual system encryption methodology is a proof strategy that utilizes another subgroup for increasing the entropy, so that we can finally achieve unconditionally failed decryption (and weak attribute-hiding). We first present the forms of $\mathsf{ek}_{\mathbf{y}}$, $\mathsf{dk}_{\mathbf{t}}, \mathsf{dk}_{\mathbb{M}}$ and ct used in our proof:

(i) Form of $\mathsf{ek}_{\mathbf{y}}$:

(i) *Normal*:

$$
\begin{aligned}
K_0^{(2)} &= g^{s_2\mathbf{d}_3}\\
K_1^{(2)} &= g^{s_2 w_1 y_1\mathbf{d}_3}\\
&\cdots\\
K_n^{(2)} &= g^{s_2 w_n y_n\mathbf{d}_3}\\
K_{n+1}^{(2)} &= g^{s_2 v\mathbf{d}_3}
\end{aligned}
\quad .
\tag{51}
$$

(ii) Forms of $\mathsf{dk}_{\mathbf{t}}$:

(i) *Normal*:

$$
\begin{aligned}
K_0^{(1)} &= h^{r_1\mathbf{d}_1^*}\\
K_1^{(1)} &= h^{\alpha_1\mathbf{d}_1^*+\alpha_2\mathbf{d}_3^*+\sum_i r_1 w'_{i+1}t_i\mathbf{d}_1^*}\cdot
\end{aligned}
\tag{52}
$$

(ii) *SF 1*:

$$
\begin{aligned}
K_0^{(1)} &= h^{r_1\mathbf{d}_1^*+r'_1\mathbf{d}_2^*}\\
K_1^{(1)} &= h^{\alpha_1\mathbf{d}_1^*+\alpha_2\mathbf{d}_3^*+\sum_i w'_{i+1}t_i\left(r_1\mathbf{d}_1^*+r'_1\mathbf{d}_2^*\right)}\cdot
\end{aligned}
\tag{53}
$$

where $r'_1 \leftarrow_R \mathbb{Z}_p$.

(iii) *SF 2*:

$$K_0^{(1)} = h^{r_1\mathbf{d}_1^* + r_1'\mathbf{d}_2^*}$$
$$K_1^{(1)} = h^{\alpha_1\mathbf{d}_1^* + \alpha'\mathbf{d}_2^* + \alpha_2\mathbf{d}_3^* + \sum_i w_{i+1}' t_i (r_1\mathbf{d}_1^* + r_1'\mathbf{d}_2^*)} \cdot \tag{54}$$

where $\alpha' \leftarrow_R \mathbb{Z}_p$.

(iv) *SF 3*:

$$K_0^{(1)} = h^{r_1\mathbf{d}_1^*}$$
$$K_1^{(1)} = h^{\alpha_1\mathbf{d}_1^* + \alpha'\mathbf{d}_2^* + \alpha_2\mathbf{d}_3^* + \sum_i r_1 w_{i+1}' t_i \mathbf{d}_1^*} \cdot \tag{55}$$

(iii) Form of $\mathsf{dk}_{\mathbb{M}}$:

(i) *Normal*:

$$\begin{aligned} K_0^{(3)} &= h^{r_2\mathbf{d}_3^*} \\ K_1^{(3)} &= h^{\left(r_2 w_1 + r_2 v_i + \alpha_{2_i}\right)\mathbf{d}_3^*} \\ &\cdots \\ K_n^{(3)} &= h^{\left(r_2 w_n + r_2 v_n + \alpha_{2_n}\right)\mathbf{d}_3^*} \end{aligned} \cdot \tag{56}$$

(iv) Forms of ct:

(i) *Normal*:

$$\begin{aligned} C_0 &= g^{s_1\mathbf{d}_1 + s_2\mathbf{d}_3} \\ C_1 &= g^{\left(s_1 w_1' x_1 + s_1 w_2'\right)\mathbf{d}_1 + s_2 w_1 y_1 \mathbf{d}_3} \\ &\cdots \\ C_n &= g^{\left(s_1 w_1' x_n + s_1 w_{n+1}'\right)\mathbf{d}_1 + s_2 w_n y_n \mathbf{d}_3} \\ C_{n+1} &= g^{s_2 v \mathbf{d}_3} \\ C_{n+2} &= e\left(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*}\right)^{\alpha_1 s_1} \cdot m \end{aligned} \cdot \tag{57}$$

(ii) *SF*:

$$\begin{aligned} C_0 &= g^{s_1\mathbf{d}_1 + s_1'\mathbf{d}_2 + s_2\mathbf{d}_3} \\ C_1 &= g^{\left(w_1' x_1 + w_2'\right)\left(s_1\mathbf{d}_1 + s_1'\mathbf{d}_2\right) + s_2 w_1 y_1 \mathbf{d}_3} \\ &\cdots \\ C_n &= g^{\left(w_1' x_n + w_{n+1}'\right)\left(s_1\mathbf{d}_1 + s_1'\mathbf{d}_2\right) + s_2 w_n y_n \mathbf{d}_3} \\ C_{n+1} &= g^{s_2 v \mathbf{d}_3} \\ C_{n+2} &= e\left(g^{s_1\mathbf{d}_1 + s_1'\mathbf{d}_2}, h^{\mathbf{d}_1^*}\right)^{\alpha_1} \cdot m \end{aligned} \cdot \tag{58}$$

where $s_1' \leftarrow_R \mathbb{Z}_p$.

We then list our games as follows:

(i) Game_0: This is the same as the real construction.

(ii) Game_1: This is the same as Game_0, except that we change ct from *Normal* to *SF*.

(iii) $\mathsf{Game}_{2,j,1}$: For $j \in [D]$, $\mathsf{Game}_{2,j,1}$ is the same as $\mathsf{Game}_{2,j-1,3}$, except that we change $\mathsf{dk}_{\mathbf{v}_j}$ from *Normal* to *SF 1*. Note that $\mathsf{Game}_{2,0,3}$ is exactly Game_1.

(iv) $\mathsf{Game}_{2,j,2}$: For $j \in [D]$, $\mathsf{Game}_{2,j,2}$ is the same as $\mathsf{Game}_{2,j,1}$, except that we change $\mathsf{dk}_{\mathbf{v}_j}$ from *SF 1* to *SF 2*.

(v) $\mathsf{Game}_{2,j,3}$: For $j \in [D]$, $\mathsf{Game}_{2,j,3}$ is the same as $\mathsf{Game}_{2,j,2}$, except that we change $\mathsf{dk}_{\mathbf{v}_j}$ from *SF 2* to *SF 3*.

(vi) Game_3: This is the same as $\mathsf{Game}_{2,D,3}$, except that we change the challenge $(m_\beta, \mathbf{x}_\beta, \mathbf{y}_\beta)$ to $(m_R, \mathbf{x}_R, \mathbf{y}_R)$, where $m_R \leftarrow_R G_T$ and $\mathbf{x}_R, \mathbf{y}_R \leftarrow_R \mathbb{Z}_p^n$. □

Lemma 1. *Under DS1 assumption, we have*

$$\left|\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_1}(\lambda) - \mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_0}(\lambda)\right| \le \mathsf{Adv}_{\mathscr{B}_1}^{DS1}(\lambda). \tag{59}$$

Proof. Suppose that we have

$$\left|\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_1}(\lambda) - \mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_0}(\lambda)\right| = \epsilon, \tag{60}$$

where ϵ is a non-negligible value.

Then we can build a PPT adversary $\mathscr{B}_1$ so that $\mathsf{Adv}_{\mathscr{B}_1}^{DS1}(\lambda) = \epsilon$ as follows:

$\mathscr{B}_1$ is given $(\mathbb{G}, h^{\mathbf{d}_1^*}, h^{\mathbf{d}_3^*}, g^{\mathbf{d}_1}, g^{\mathbf{d}_2}, g^{\mathbf{d}_3}, h^{\mu_1\mathbf{d}_1^* + \mu_2\mathbf{d}_2^*}, \mu_2)$, then pick $w_1, \ldots, w_n, w_1', \ldots, w_n', w_{n+1}', v, \alpha_1, \alpha_2 \leftarrow_R \mathbb{Z}_p$. $\mathscr{B}_1$ sends $\mathsf{mpk} = (g^{\mathbf{d}_1}, g^{w_1'\mathbf{d}_1}, \ldots, g^{w_n'\mathbf{d}_1}, g^{w_{n+1}'\mathbf{d}_1}, e\left(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*}\right)^{\alpha_1})$ to $\mathscr{A}$, and stores $\mathsf{msk} = (h^{\mathbf{d}_1^*}, h^{\mathbf{d}_3^*}, g^{\mathbf{d}_3}, w_1, \ldots, w_n, w_1', \ldots, w_n', w_{n+1}', v, \alpha_1, \alpha_2)$ secretly.

Upon $\mathscr{A}$ making ek queries for $\mathbf{y}_i, i \in [E]$, $\mathscr{B}_1$ simulates $\mathsf{ek}_{\mathbf{y}_i}$ as the real algorithm SKGen does, and sends the outputs back to $\mathscr{A}$.

Upon $\mathscr{A}$ making dk queries for $\mathbb{M}_\ell, \ell \in [D]$, $\mathscr{B}_1$ simulates $\mathsf{dk}_{\mathbb{M}_\ell}$ as the real algorithm RKGen does, and sends the outputs back to $\mathscr{A}$.

Upon $\mathscr{A}$ making the challenge $((m_0, \mathbf{x}_0, \mathbf{y}_0), (m_1, \mathbf{x}_1, \mathbf{y}_1))$, $\mathscr{B}_1$ chooses $\beta \leftarrow_R \{0, 1\}$, and simulates ct_β as follows:

Pick $s_2 \leftarrow_R \mathbb{Z}_p$, then generate ct_β with the challenge T of DS1 assumption as follows:

$$\begin{aligned} C_0 &= T \cdot g^{s_2\mathbf{d}_3} \\ C_1 &= T^{\left(w_1' x_{\beta_1} + w_2'\right)} \cdot g^{s_2 w_1 y_{\beta_1}\mathbf{d}_3} \\ &\cdots \\ C_n &= T^{\left(w_1' x_{\beta_n} + w_{n+1}'\right)} \cdot g^{s_2 w_n y_{\beta_n}\mathbf{d}_3} \\ C_{n+1} &= g^{s_2 v \mathbf{d}_3} \\ C_{n+2} &= e\left(T, h^{\mathbf{d}_1^*}\right)^{\alpha_1} \cdot m_\beta \end{aligned}, \tag{61}$$

$\mathscr{B}_1$ sends ct_β back to $\mathscr{A}$.

Observe that if $T = g^{s_1\mathbf{d}_1}$ where $s_1 \leftarrow_R \mathbb{Z}_p$, ct_β is the same as Game_0; if $T = g^{s_1\mathbf{d}_1 + s_1'\mathbf{d}_2}$ where $s_1, s_1' \leftarrow_R \mathbb{Z}_p$, ct_β is the same as Game_1. Then we can successfully build an adversary $\mathscr{B}_1$ to break DS1 assumption, which is contrary to the fact that breaking DS1 assumption is hard. □

Lemma 2. *Under DS2 assumption, we have*

$$\left|\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j,1}}(\lambda)-\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j-1,3}}(\lambda)\right|\leq \mathsf{Adv}_{\mathscr{B}_2}^{DS2}(\lambda). \tag{62}$$

Proof. Suppose that we have

$$\left|\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j,1}}(\lambda)-\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j-1,3}}(\lambda)\right|=\epsilon, \tag{63}$$

where ϵ is a non-negligible value.

Then we can build a PPT adversary $\mathscr{B}_2$ so that $\mathsf{Adv}_{\mathscr{B}_2}^{DS2}(\lambda)=\epsilon$ as follows:

$\mathscr{B}_2$ is given $(\mathbb{G}, g^{\mathbf{d}_1}, g^{\mathbf{d}_3}, h^{\mathbf{d}_1^*}, h^{\mathbf{d}_2^*}, h^{\mathbf{d}_3^*}, g^{\mu_1\mathbf{d}_1+\mu_2\mathbf{d}_2}, \mu_2)$, then pick $w_1,\ldots,w_n,\ w_1',\ldots,w_n', w_{n+1}', \nu, \alpha_1, \alpha_2 \leftarrow_R \mathbb{Z}_p$. $\mathscr{B}_2$ sends $\mathsf{mpk}=(g^{\mathbf{d}_1}, g^{w_1'\mathbf{d}_1},\ldots, g^{w_n'\mathbf{d}_1}, g^{w_{n+1}'\mathbf{d}_1}, \ e(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*})^{\alpha_1})$ to $\mathscr{A}$, and stores $\mathsf{msk}=(h^{\mathbf{d}_1^*}, h^{\mathbf{d}_3^*}, g^{\mathbf{d}_3}, w_1,\ldots,w_n, w_1',\ldots,w_n', w_{n+1}', \nu, \alpha_1, \alpha_2)$ secretly.

Upon $\mathscr{A}$ making ek queries for $\mathbf{y}_i, i\in[E]$, $\mathscr{B}_2$ simulates $\mathsf{ek}_{\mathbf{y}_i}$ as the real algorithm SKGen does, and sends the outputs back to $\mathscr{A}$.

Upon $\mathscr{A}$ making dk queries for $\mathbf{t}_\ell, \ell\in[D]$, $\mathscr{B}_2$ simulates $\mathsf{dk}_{\mathbf{t}_\ell}$ for $\ell\in[j-1]$ as follows:

Pick $r_{\ell,1}\leftarrow_R\mathbb{Z}_p$, then generate $\mathsf{dk}_{\mathbf{t}_\ell}$ as follows:

$$\begin{aligned} K_{\ell,0}^{(1)} &= h^{r_{\ell,1}\mathbf{d}_1^*} \\ K_{\ell,1}^{(1)} &= h^{\alpha_1\mathbf{d}_1^*+\alpha_\ell'\mathbf{d}_2^*+\alpha_2\mathbf{d}_3^*+\sum_i r_{\ell,1}w_{i+1}'t_{\ell_i}\mathbf{d}_1^*}. \end{aligned} \tag{64}$$

$\mathscr{B}_2$ simulates $\mathsf{dk}_{\mathbf{t}_\ell}$ for $\ell=j$ as follows:

Generate $\mathsf{dk}_{\mathbf{t}_\ell}$ with the challenge T of DS2 assumption as follows:

$$\begin{aligned} K_{j,0}^{(1)} &= T \\ K_{j,1}^{(1)} &= h^{\alpha_1\mathbf{d}_1^*+\alpha_2\mathbf{d}_3^*}\cdot T^{\sum_i w_{i+1}'t_{j,i}}. \end{aligned} \tag{65}$$

$\mathscr{B}_2$ simulates $\mathsf{dk}_{\mathbf{t}_\ell}$ for $\ell\in\{j+1,\ldots,D\}$ as the real algorithm does, and then $\mathscr{B}_2$ sends $\mathsf{dk}_{\mathbf{t}_\ell}, \ell\in[D]$ back to $\mathscr{A}$.

Upon $\mathscr{A}$ making dk queries for $\mathbb{M}_\ell, \ell\in[D]$, $\mathscr{B}_2$ simulates $\mathsf{dk}_{\mathbb{M}_\ell}$ as the real algorithm PolGen does, and sends the outputs back to $\mathscr{A}$.

Upon $\mathscr{A}$ making the challenge $((m_0,\mathbf{x}_0,\mathbf{y}_0),(m_1,\mathbf{x}_1,\mathbf{y}_1))$, $\mathscr{B}_2$ chooses $\beta\leftarrow_R\{0,1\}$, and simulates ct_β as follows:

Pick $s_2\leftarrow_R\mathbb{Z}_p$, then generate ct_β as follows:

$$\begin{aligned} C_0 &= g^{\mu_1\mathbf{d}_1+\mu_2\mathbf{d}_2+s_2\mathbf{d}_3} \\ C_1 &= g^{(w_1'x_{\beta_1}+w_2')(\mu_1\mathbf{d}_1+\mu_2\mathbf{d}_2)+s_2w_1y_{\beta_1}\mathbf{d}_3} \\ &\cdots \\ C_n &= g^{(w_1'x_{\beta_n}+w_{n+1}')(\mu_1\mathbf{d}_1+\mu_2\mathbf{d}_2)+s_2w_ny_{\beta_n}\mathbf{d}_3} \\ C_{n+1} &= g^{s_2\nu\mathbf{d}_3} \\ C_{n+2} &= e(g^{\mu_1\mathbf{d}_1+\mu_2\mathbf{d}_2}, h^{\mathbf{d}_1^*})^{\alpha_1}\cdot m_\beta. \end{aligned} \tag{66}$$

$\mathscr{B}_2$ sends ct_β back to $\mathscr{A}$.

Observe that if $T=h^{r_{j,1}\mathbf{d}_1^*}$ where $r_{j,1}\leftarrow_R\mathbb{Z}_p$, $\mathsf{dk}_{\mathbf{t}_\ell}$ is the same as $\mathsf{Game}_{2,j-1,3}$; if $T=h^{r_{j,1}\mathbf{d}_1^*+r_{j,1}'\mathbf{d}_2^*}$ where $r_{j,1}$, $r_{j,1}'\leftarrow_R\mathbb{Z}_p$, $\mathsf{dk}_{\mathbf{t}_\ell}$ is the same as $\mathsf{Game}_{2,j,1}$. Then we can successfully build an adversary $\mathscr{B}_2$ to break DS2 assumption, which is contrary to the fact that breaking DS2 assumption is hard. □

Lemma 3. *We have*

$$\left|\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j,2}}(\lambda)-\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j,1}}(\lambda)\right|=0. \tag{67}$$

That is, $\mathsf{Game}_{2,j,2}$ *and* $\mathsf{Game}_{2,j,1}$ *are identically distributed.*

Proof. Observe that the change occurs only in $\mathsf{dk}_{\mathbf{t}_j}$, which is from $K_{j,1}^{(1)}=h^{\alpha_1\mathbf{d}_1^*+\alpha_2\mathbf{d}_3^*+\sum_i w_{i+1}'t_{j,i}\left(r_{j,1}\mathbf{d}_1^*+r_{j,1}'\mathbf{d}_2^*\right)}$ to $K_{j,1}^{(1)}=h^{\alpha_1\mathbf{d}_1^*+\alpha_j'\mathbf{d}_2^*+\alpha_2\mathbf{d}_3^*+\sum_i w_{i+1}'t_{j,i}\left(r_{j,1}\mathbf{d}_1^*+r_{j,1}'\mathbf{d}_2^*\right)}$. The identical distribution follows from α-privacy property and the attribute-hiding encoding [11, 12]. □

Lemma 4. *Under DS2 assumption, we have*

$$\left|\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j,3}}(\lambda)-\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j,2}}(\lambda)\right|\leq \mathsf{Adv}_{\mathscr{B}_3}^{DS2}(\lambda). \tag{68}$$

Proof. Suppose that we have

$$\left|\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j,3}}(\lambda)-\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,j,2}}(\lambda)\right|=\epsilon, \tag{69}$$

where ϵ is a non-negligible value.

Then we can build a PPT adversary $\mathscr{B}_3$ so that $\mathsf{Adv}_{\mathscr{B}_3}^{DS2}(\lambda)=\epsilon$ as follows:

$\mathscr{B}_3$ is given $(\mathbb{G}, g^{\mathbf{d}_1}, g^{\mathbf{d}_3}, h^{\mathbf{d}_1^*}, h^{\mathbf{d}_2^*}, h^{\mathbf{d}_3^*}, g^{\mu_1\mathbf{d}_1+\mu_2\mathbf{d}_2}, \mu_2)$, then pick $w_1,\ldots,w_n,\ w_1',\ldots,w_n', w_{n+1}', \nu, \alpha_1, \alpha_2 \leftarrow_R \mathbb{Z}_p$. $\mathscr{B}_1$ sends $\mathsf{mpk}=(g^{\mathbf{d}_1}, g^{w_1'\mathbf{d}_1},\ldots, g^{w_n'\mathbf{d}_1}, g^{w_{n+1}'\mathbf{d}_1}, \ e(g^{\mathbf{d}_1}, h^{\mathbf{d}_1^*})^{\alpha_1})$ to $\mathscr{A}$, and stores $\mathsf{msk}=(h^{\mathbf{d}_1^*}, h^{\mathbf{d}_3^*}, g^{\mathbf{d}_3}, w_1,\ldots,w_n, w_1',\ldots,w_n', w_{n+1}', \nu, \alpha_1, \alpha_2)$ secretly.

Upon $\mathscr{A}$ making ek queries for $\mathbf{y}_i, i\in[E]$, $\mathscr{B}_3$ simulates $\mathsf{ek}_{\mathbf{y}_i}$ as the real algorithm SKGen does, and sends the outputs back to $\mathscr{A}$.

Upon $\mathscr{A}$ making dk queries for $\mathbf{t}_\ell, \ell\in[D]$, $\mathscr{B}_3$ simulates $\mathsf{dk}_{\mathbf{t}_\ell}$ for $\ell\in[j-1]$ as follows:

Pick $r_1\leftarrow_R\mathbb{Z}_p$, then generate $\mathsf{dk}_{\mathbf{t}_\ell}$ as follows:

$$\begin{aligned} K_{\ell,0}^{(1)} &= h^{r_{\ell,1}\mathbf{d}_1^*} \\ K_{\ell,1}^{(1)} &= h^{\alpha_1\mathbf{d}_1^*+\alpha_\ell'\mathbf{d}_2^*+\alpha_2\mathbf{d}_3^*+\sum_i r_{\ell,1}w_{i+1}'t_{\ell_i}\mathbf{d}_1^*}. \end{aligned} \tag{70}$$

$\mathscr{B}_3$ simulates $\mathsf{dk}_{\mathbf{t}_\ell}$ for $\ell=j$ as follows:

Generate $\mathsf{dk}_{\mathbf{t}_\ell}$ with the challenge T of DS2 assumption as follows:

$$K_{j,0}^{(1)} = T$$
$$K_{j,1}^{(1)} = h^{\alpha_1 \mathbf{d}_1^* + \alpha_j' \mathbf{d}_2^* + \alpha_2 \mathbf{d}_3^*} \cdot T^{\sum_i w_{i+1}' t_{\ell_i}} \cdot \tag{71}$$

$\mathscr{B}_3$ simulates $\mathsf{dk}_{\mathbf{t}_\ell}$ for $\ell \in \{j+1, \ldots, D\}$ as the real algorithm does, and then $\mathscr{B}_2$ sends $\mathsf{dk}_{\mathbf{t}_\ell}, \ell \in [D]$ back to $\mathscr{A}$.

Upon $\mathscr{A}$ making dk queries for $\mathbb{M}_\ell, \ell \in [D]$, $\mathscr{B}_3$ simulates $\mathsf{dk}_{\mathbb{M}_\ell}$ as the real algorithm PolGen does, and sends the outputs back to $\mathscr{A}$.

Upon $\mathscr{A}$ making the challenge $((m_0, \mathbf{x}_0, \mathbf{y}_0), (m_1, \mathbf{x}_1, \mathbf{y}_1))$, $\mathscr{B}_3$ chooses $\beta \leftarrow_R \{0,1\}$, and simulates ct_β as follows:

Pick $s_2 \leftarrow_R \mathbb{Z}_p$, then generate ct_β as follows:

$$\begin{aligned}
C_0 &= g^{\mu_1 \mathbf{d}_1 + \mu_2 \mathbf{d}_2 + s_2 \mathbf{d}_3} \\
C_1 &= g^{(w_1' x_{\beta_1} + w_2')(\mu_1 \mathbf{d}_1 + \mu_2 \mathbf{d}_2) + s_2 w_1 y_{\beta_1} \mathbf{d}_3} \\
&\cdots \\
C_n &= g^{(w_1' x_{\beta_n} + w_{n+1}')(\mu_1 \mathbf{d}_1 + \mu_2 \mathbf{d}_2) + s_2 w_n y_{\beta_n} \mathbf{d}_3} \\
C_{n+1} &= g^{s_2 v \mathbf{d}_3} \\
C_{n+2} &= e\left(g^{\mu_1 \mathbf{d}_1 + \mu_2 \mathbf{d}_2}, h^{\mathbf{d}_1^*}\right)^{\alpha_1} \cdot m_\beta.
\end{aligned} \tag{72}$$

$\mathscr{B}_3$ sends ct_β back to $\mathscr{A}$.

Observe that if $T = h^{r_{j,1} \mathbf{d}_1^* + r_{j,1}' \mathbf{d}_2^*}$ where $r_{j,1}, r_{j,1}' \leftarrow_R \mathbb{Z}_p$, $\mathsf{dk}_{\mathbf{t}_\ell}$ is the same as $\mathsf{Game}_{2,j,2}$; if $T = h^{r_{j,1} \mathbf{d}_1^*}$ where $r_{j,1} \leftarrow_R \mathbb{Z}_p$, $\mathsf{dk}_{\mathbf{t}_\ell}$ is the same as $\mathsf{Game}_{2,j,3}$. Then we can successfully build an adversary $\mathscr{B}_3$ to break DS2 assumption, which is contrary to the fact that breaking DS2 assumption is hard. □

Lemma 5. *We have*

$$\left| \mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_3}(\lambda) - \mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{2,D,3}}(\lambda) \right| = 0. \tag{73}$$

That is, Game_3 *and* $\mathsf{Game}_{2,D,3}$ *are identically distributed.*

Proof. Since the symmetric key is changed into random values and the predicate encoding of inner product satisfy the attribute-hiding encoding in [11], thus Game_3 and $\mathsf{Game}_{2,D,3}$ are identically distributed. □

Proof of Authenticity

Theorem 6. *For any* PPT *adversary* $\mathscr{A}$*, we have*

$$\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{auth}}(\lambda) = \mathsf{Pr}\left[\mathsf{Game}_{\mathscr{A}}^{auth}(\lambda) = 1\right] \leq \mathsf{Adv}_{\mathscr{B}}^{ABE_p}(\lambda), \tag{74}$$

where $\mathscr{B}$ *is defined in the following lemma.*

Proof. The authenticity can be reduced to the security of the ABE scheme corresponding to $\mathbf{y}$-$\mathbb{M}$ pair based on DS assumptions, which is embedded in our IP-ME with MSP Auth scheme.

Suppose that

$$\mathsf{Adv}_{\mathscr{A}}^{\mathsf{Game}_{auth}}(\lambda) = \epsilon, \tag{75}$$

where ϵ is a non-negligible value.

Then we can build an adversary $\mathscr{B}$ so that $\mathsf{Adv}_{\mathscr{B}}^{ABE_p}(\lambda) = \epsilon$ as follows:

Upon $\mathscr{A}$ making a query of $(\mathsf{ct}_{\mathbf{x},\mathbf{y}}, \mathbf{t}, \mathbb{M})$, $\mathscr{B}$ generates $\mathsf{dk}_{\mathbf{t}}$ and $\mathsf{dk}_{\mathbb{M}}$ as the real algorithms do, and sends $\mathsf{dk}_{\mathbf{t}}$ and $\mathsf{dk}_{\mathbb{M}}$ back to $\mathscr{A}$. Then $\mathscr{A}$ can find $\mathbf{y}^*$ satisfying $\mathbb{M}$ with ϵ probability such that $\mathbf{y}^*$ is also valid for generating $\mathsf{ct}_{\mathbf{x},\mathbf{y}}$ when decrypting with policy $\mathbb{M}$, and then sends $\mathbf{y}^*$ to $\mathscr{B}$. Note that the fact that $\mathbf{y}$ and $\mathbf{y}^*$ are both valid for $\mathsf{ct}_{\mathbf{x},\mathbf{y}}$ implies that for a ciphertext associated with $\mathbb{M}$ in the underlying ABE, there would be two valid secret keys associated with $\mathbf{y}$ and $\mathbf{y}^*$ respectively. Therefore, $\mathscr{B}$ can make secret key query of $\mathbf{y}^*$, and challenge $(m_0, \mathbf{y})$ and $(m_1, \mathbf{y}')$. Then $\mathscr{B}$ can distinguish the challenge ciphertext easily by using the secret key associated with $\mathbf{y}^*$. Thus, we obtain a contradiction. □

7. Conclusion

ME is a cryptographic primitive that supports fine-grained access control for both the sender and the receiver. It can be applied in scenarios that especially require anonymity, such as Tor network. Currently, there have existed a nontheoretically modular framework of ME, but it consist of more than one building blocks, thus its construction is not simple enough and might be under different assumptions or even not in the standard model. There have also existed some IB-ME schemes, which support only the equality policy, but are of comparatively simple constructions.

For cryptographic primitives, we are desirable for schemes under standard assumptions, since standard assumptions are well-studied so that they can guarantee the security better. We are also desirable for schemes in the standard model, since schemes in the standard model are more secure than those not in the standard model. For example, there have been some schemes secure in the random oracle model, but not secure in the standard model.

To explore simpler ME schemes for more expressive functionalities under standard assumptions in the standard model, we present an IP-ME scheme and an IP-ME with MSP scheme both under SXDH assumption in the standard model. The policies for access control of our schemes are beyond equality policy, and reach inner-product policy as well as MSP policy. Therefore, our schemes are more expressive than IB-ME schemes.

Data Availability

No underlying data was collected or produced in this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Anmin Fu, Haifeng Qian, Qiaohan Chu, and Jie Chen contributed in the methodology. Qiaohan Chu contributed in the writing. Anmin Fu, Jie Chen, and Haifeng Qian contributed in the reviewing. Haifeng Qian and Jie Chen contributed in the funding acquisition.

Acknowledgments

This research was supported by National Natural Science Foundation of China (61972156, 62372180), NSFC-ISF Joint Scientific Research Program (61961146004), National Key Research and Development Program of China (2018YFA0704701), and Innovation Program of Shanghai Municipal Education Commission (2021-01-07-00-08-E00101).

References

[1] G. Ateniese, D. Francati, D. Nuñez, and D. Venturi, "Match me if you can: matchmaking encryption and its applications," in *CRYPTO 2019*, A. Boldyreva and D. Micciancio, Eds., vol. 11693 of *Lecture Notes in Computer Science*, pp. 701–731, Springer, Cham, 2019.

[2] D. Balfanz, G. Durfee, N. Shankar, D. K. Smetters, J. Staddon, and H. Wong, "Secret handshakes from pairing-based key agreements," in *2003 Symposium on Security and Privacy, 2003.*, pp. 180–196, IEEE, Berkeley, CA, USA, 2003.

[3] D. Francati, A. Guidi, L. Russo, and D. Venturi, "Identity-based matchmaking encryption without random oracles," in *INDOCRYPT 2021*, A. Adhikari, R. Küsters, and B. Preneel, Eds., vol. 13143 of *Lecture Notes in Computer Science*, pp. 415–435, Springer, Cham, 2021.

[4] J. Chen, Y. Li, J. Wen, and J. Weng, "Identity-based matchmaking encryption from standard assumptions," in *Advances in Cryptology–ASIACRYPT 2022*, vol. 13793 of *Lecture Notes in Computer Science*, pp. 394–422, Springer, Cham, 2022.

[5] A. Beimel, *Secure schemes for secret sharing and key distribution*, phd thesis israel institute of technology technion, Technion-Israel Institute of Technology, Faculty of computer science, 1996.

[6] M. Karchmer and A. Wigderson, "On span programs," in *Proceedings of the Eighth Annual Structure in Complexity Theory Conference*, pp. 102–111, IEEE, San Diego, CA, USA, 1993.

[7] T. Okamoto and K. Takashima, "Fully secure functional encryption with general relations from the decisional linear assumption," in *CRYPTO 2010*, T. Rabin, Ed., vol. 6223, pp. 191–208, Springer, Berlin, Heidelberg, 2010.

[8] A. Guillevic, "Comparing the pairing efficiency over composite-order and prime-order elliptic curves," in *ACNS 2013*, vol. 7954 of *Lecture Notes in Computer Science*, pp. 357–372, Springer, Berlin, Heidelberg, 2013.

[9] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *EUROCRYPT 2008*, N. P. Smart, Ed., vol. 4965, pp. 146–162, Springer, Berlin, Heidelberg, 2008.

[10] H. Wee, "Attribute-hiding predicate encryption in bilinear groups," in *TCC 2017*, Y. Kalai and L. Reyzin, Eds., vol. 10677 of *Lecture Notes in Computer Science*, pp. 206–233, Springer, Cham, 2017.

[11] J. Chen, R. Gay, and H. Wee, "Improved dual system ABE in prime-order groups via predicate encodings," in *EUROCRYPT 2015*, E. Oswald and M. Fischlin, Eds., vol. 9057 of *Lecture Notes in Computer Science*, pp. 595–624, Springer, Berlin, Heidelberg, 2015.

[12] H. Wee, "Dual system encryption via predicate encodings," in *TCC 2014*, Y. Lindell, Ed., vol. 8349 of *Lecture Notes in Computer Science*, pp. 616–637, Springer, Berlin, Heidelberg, 2014.

[13] A. Lewko and B. Waters, "New techniques for dual system encryption and fully secure HIBE with short ciphertexts," in *TCC 2010*, D. Micciancio, Ed., vol. 5978, pp. 455–479, Springer, Berlin, Heidelberg, 2010.

[14] S. Agrawal and M. Chase, "A study of pair encodings: predicate encryption in prime order groups," in *TCC 2016-A*, E. Kushilevitz and T. Malkin, Eds., vol. 9563 of *Lecture Notes in Computer Science*, pp. 259–288, Springer, Berlin, Heidelberg, 2016.

[15] N. Attrapadung, "Dual system encryption framework in prime-order groups via computational pair encodings," in *ASIACRYPT 2016*, J. H. Cheon and T. Takagi, Eds., vol. 10032 of *Lecture Notes in Computer Science*, pp. 591–623, Springer, Berlin, Heidelberg, 2016.

[16] J. Chen, J. Gong, L. Kowalczyk, and H. Wee, "Unbounded ABE via bilinear entropy expansion," in *EUROCRYPT 2018*, J. B. Nielsen and V. Rijmen, Eds., vol. 10820 of *Lecture Notes in Computer Science*, pp. 503–534, Springer, Cham, 2018.

[17] J. Chen, J. Gong, and H. Wee, "Improved inner-product encryption with adaptive security and full attribute-hiding," in *ASIACRYPT 2018*, T. Peyrin and S. D. Galbraith, Eds., vol. 11273, pp. 673–702, Springer, Cham, 2018.

[18] R. Gay, D. Hofheinz, E. Kiltz, and H. Wee, "Tightly CCA-secure encryption without pairings," in *EUROCRYPT 2016*, M. Fischlin and J. S. Coron, Eds., vol. 9665 of *Lecture Notes in Computer Science*, pp. 1–27, Springer, Berlin, Heidelberg, 2016.

[19] J. Gong, J. Chen, X. Dong, Z. Cao, and S. Tang, "Extended nested dual system groups," in *PKC 2016*, C. M. Cheng, K. M. Chung, G. Persiano, and B. Y. Yang, Eds., vol. 9614 of *Lecture Notes in Computer Science*, pp. 133–163, Springer, Berlin, Heidelberg, 2016.

[20] A. Lewko, "Tools for simulating features of composite order bilinear groups in the prime order setting," in *EUROCRYPT 2012*, D. Pointcheval and T. Johansson, Eds., vol. 7237, pp. 318–335, Springer, Berlin, Heidelberg, 2012.

[21] T. Okamoto and K. Takashima, "Fully secure unbounded inner-product and attribute-based encryption," in *ASIACRYPT 2012*, X. Wang and K. Sako, Eds., vol. 7658 of *Lecture Notes in Computer Science*, pp. 349–366, Springer, Berlin, Heidelberg, 2012.

[22] J. Chen and H. Wee, "Fully, (almost) tightly secure IBE and dual system groups," in *CRYPTO 2013*, R. Canetti and J. A. Garay, Eds., vol. 8043 of *Lecture Notes in Computer Science*, pp. 435–460, Springer, Berlin, Heidelberg, 2013.

[23] T. Okamoto and K. Takashima, "Homomorphic encryption and signatures from vector decomposition," in *Pairing 2008*, S. D. Galbraith and K. G. Paterson, Eds., vol. 5209 of *Lecture Notes in Computer Science*, pp. 57–74, Springer, Berlin, Heidelberg, 2008.

[24] T. Okamoto and K. Takashima, "Hierarchical predicate encryption for inner-products," in *ASIACRYPT 2009*, M. Matsui, Ed., vol. 5912 of *Lecture Notes in Computer Science*, pp. 214–231, Springer, Berlin, Heidelberg, 2009.

[25] J. Chen, H. W. Lim, S. Ling, H. Wang, and H. Wee, "Shorter IBE and signatures via asymmetric pairings," in *Pairing-Based Cryptography–Pairing 2012*, M. Abdalla and T. Lange, Eds., vol. 7708 of *Lecture Notes in Computer Science*, pp. 122–140, Springer, Berlin, Heidelberg, 2012.

[26] J. Chen, H. W. Lim, S. Ling, H. Wang, and H. Wee, "Shorter identity-based encryption via asymmetric pairings," *Designs, Codes and Cryptography*, vol. 73, pp. 911–947, 2014.

[27] P. Datta, T. Okamoto, and K. Takashima, "Efficient attribute-based signatures for unbounded arithmetic branching programs," in *PKC 2019*, D. Lin and K. Sako, Eds., vol. 11442 of *Lecture Notes in Computer Science*, pp. 127–158, Springer, Cham, 2019.

[28] H. K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-based signatures: achieving attribute-privacy and collusion-resistance," 2008, IACR Cryptol. ePrint Arch. p. 328 (2008), http://eprint.iacr.org/2008/328.

[29] T. Okamoto and K. Takashima, "Efficient attribute-based signatures for non-monotone predicates in the standard model," in *PKC 2011*, D. Catalano, N. Fazio, R. Gennaro, and A. Nicolosi, Eds., vol. 6571 of *Lecture Notes in Computer Science*, pp. 35–52, Springer, Berlin, Heidelberg, 2011.

[30] T. Okamoto and K. Takashima, "Decentralized attribute-based signatures," in *PKC 2013*, K. Kurosawa and G. Hanaoka, Eds., vol. 7778 of *Lecture Notes in Computer Science*, pp. 125–142, Springer, Berlin, Heidelberg, 2013.

[31] G. Ateniese, D. Francati, D. Nuñez, and D. Venturi, "Match me if you can: matchmaking encryption and its applications," *Journal of Cryptology*, vol. 34, no. 3, Article ID 16, 2021.

[32] D. Francati, D. Friolo, G. Malavolta, and D. Venturi, "Multi-key and multi-input predicate encryption from learning with errors," in *EUROCRYPT 2023*, C. Hazay and M. Stam, Eds., vol. 14006, pp. 573–604, Springer, Cham, 2023.

[33] S. Xu, J. Ning, Y. Li et al., "Match in my way: fine-grained bilateral access control for secure cloud-fog computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 2, pp. 1064–1077, 2022.

[34] S. Xu, J. Ning, J. Ma, X. Huang, H. Pang, and R. H. Deng, "Expressive bilateral access control for internet-of-things in cloud-fog computing," in *SACMAT '21: Proceedings of the 26th ACM Symposium on Access Control Models and Technologies*, pp. 143–154, ACM, New York, NY, USA, 2021.

[35] D. Boneh, E. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *TCC 2005*, J. Kilian, Ed., vol. 3378 of *Lecture Notes in Computer Science*, pp. 325–341, Springer, Berlin, Heidelberg, 2005.

[36] B. Waters, "Dual system encryption: realizing fully secure IBE and HIBE under simple assumptions," in *CRYPTO 2009*, S. Halevi, Ed., vol. 5677 of *Lecture Notes in Computer Science*, pp. 619–636, Springer, Berlin, Heidelberg, 2009.