An implementation of ELiPS-based Ciphertext-Policy Attribute-Based Encryption

Le Hoang Anh*‡, Yuta Kawada*, Samsul Huda†, Md. Arshad Ali§, Yuta Kodera*, and Yasuyuki Nogami*
*Graduate School of Environmental, Life, Natural Science and Technology, Okayama University, Japan
†§Green Innovation Center, Okayama University, Japan
{shuda, yuta_kodera, yasuyuki.nogami}@okayama-u.ac.jp
{lhanh, yuta_kawada}@s.okayama-u.ac.jp
§Faculty of CSE, Hajee Mohammad Danesh Science and Technology University, Bangladesh
arshad@hstu.ac.bd
‡An Giang University, Vietnam National University Ho Chi Minh City, Vietnam
lhanh@agu.edu.vn

***Abstract*—Pairing-based cryptography serves as a fundamental building block for a wide range of advanced cryptographic protocols, including Ciphertext-Policy Attribute-Based Encryption (CP-ABE), blockchain, zero-knowledge Succinct Non-interactive ARgument of Knowledge (zk-SNARKs), and Homomorphic Encryption (HE). A well-designed pairing library would catalyze progress in these specific applications and encourage innovation across various cryptographic protocols that rely on pairing. However, the CP-ABE framework relies on the PBC library, which has not been updated for a long time, needs higher security levels, and is vulnerable to multiple attacks, making it less practical. In contrast, the ELiPS library offers efficient operations related to pairing-based cryptography, delivering high performance while upholding a substantial security standard. To address the issues above, enhance performance, and strengthen security in CP-ABE, we adopt and implement the ELiPS as an efficient library for pairing systems into the CP-ABE framework, namely ELiPS-based CP-ABE. First, we generate a generator g. Then, we employ Shirase's technique to convert asymmetric to symmetric pairing as the precondition of ELiPS. After that, we make several modifications to the CP-ABE framework and choose the appropriate ELiPS functions for integration. Finally, we validate our proposal through experiments involving data access authorization scenarios. The results confirm the effectiveness of our proposal, showcasing reduced computational costs for almost all functions except for the decryption cost.**

***Index Terms*—Attribute-Based Encryption, CP-ABE, ELiPS, Pairing**

I. Introduction

Numerous advanced cryptographic protocols and technologies, such as CP-ABE, blockchain, zk-SNARKs, HE, and others, are built upon the foundation of pairing-based cryptography [1]–[3]. These cryptographic protocols and technologies hold significance across various application fields and technological developments, such as blockchain, as well as in collaborative environments where multiple entities or organizations need to share data securely and selectively [4], [5].

Blockchain technology has rapidly gained traction across industries and organizations, offering a secure and transparent framework for exchanging data [4]. Integrating CP-ABE and zk-SNARKs technologies into smart contracts can facilitate the establishment of fine-grained access control policies and enhance privacy-preserving distributed access control [4], [5]. It is distributed ledger architecture allows participants to verify transactions without relying on a central authority, enhancing trust and reducing risks [6].

Secure and selective data sharing in collaborative environments presents challenges, requiring privacy and confidentiality to be maintained. CP-ABE can address these issues by enabling access to encrypted data based on specific attributes [7], [8]. CP-ABE generates secret keys based on attributes while data is encrypted using a policy, ensuring that only users possessing secret keys with matching attributes can access the ciphertext [9]. The integration of CP-ABE with blockchain technology facilitates fine-grained access control and secure data sharing among participants, eliminating the need for a central authority [2].

HE is a cryptographic technique that allows multiple computations to be performed on encrypted data without decrypting it in advance [3]. It finds utility in ensuring privacy, safeguarding, and maintaining the confidentiality of sensitive data [10]. In essence, HE allows operations to be conducted on sensitive information while it remains encrypted, thus maintaining its privacy and security throughout the entire process [3], [10].

Blockchain, CP-ABE, zk-SNARKs, and HE offer numerous benefits. Therefore, the development of a more efficient and secure pairing library would not only accelerate progress in these specific applications but also stimulate innovation across various cryptographic protocols that rely on pairing.

However, the Pairing-Based Cryptography (PBC) library, which has not been updated for a significant period and lacks sufficient security strength, may pose a potential weakness in modern cryptographic applications [11]–[13]. The PBC library supports only an 80-bit security level, rendering it vulnerable to various attacks and limiting its practicality [1], [12], [13]. J. W. Bos et al. [12] recommend that transitioning to a security level greater than 80-bit is necessary. According to E. Barker [13], an 80-bit of security is no longer regarded as being sufficiently secure.

2832-1324/23/$31.00 ©2023 IEEE
DOI 10.1109/CANDARW60564.2023.00044

On the other hand, the ELiPS[1] library provides an efficient calculation cost while ensuring high security. It is a specialized cryptographic library that concentrates on efficient operations related to pairing-based cryptography [14]. ELiPS utilizes the BLS-12 curve and offers a 128-bit security level [14]. It provides several functions that support the implementation of algorithms and protocols that utilize pairing.

To deal with the challenges mentioned above, the authors propose an ELiPS-based CP-ABE scheme that incorporates ELiPS into the CP-ABE framework. However, there are several differences between the PBC and ELiPS libraries, including function parameters and data types [11], [14]. First, generating a generator g. The PBC library employs symmetric pairing, while ELiPS utilizes asymmetric pairing. Therefore, the authors also used Shirase's method in [15] for converting asymmetric to symmetric pairing. Furthermore, we also make essential modifications to ensure compatibility between the ELiPS and CP-ABE. These modifications span the setup, key generation, encryption, and decryption algorithms.

The authors conducted several experiments to validate the proposal, which demonstrated the improved performance and security of the ELiPS-based CP-ABE scheme. We implemented a data access authorization process at the university level using several attribute policy scenarios. The experiment results show the effectiveness of our proposal, revealing reduced computational requirements for the setup, key generation, and encryption functions, except for the decryption cost.

The following are some of this paper's main contributions:

- Generate a generator g.
- Use Shirase's method to transform asymmetric to symmetric pairing.
- Make several modifications to the CP-ABE framework and carefully select appropriate ELiPS functions to ensure compatibility between ELiPS and CP-ABE.

II. Preliminaries

We first introduce the background information on arithmetic operations over the elliptic curve, pairing, and the PBC library, which play vital roles in the CP-ABE algorithm. Next, the authors present an overview of CP-ABE and ELiPS.

A. Arithmetic operations over the elliptic curve

For a prime $p > 3$, an elliptic curve E of Weierstrass form defined over $\mathbb{F}_p$ is given as follows [16]:

$$E : y^2 = x^3 + ax + b, \text{ where } a, b \in \mathbb{F}_p. \tag{1}$$

Let $P = (x_P, y_P)$ and $Q = (x_Q, y_Q)$ be affine rational points on E, as can be seen in (1). The arithmetic operations over the elliptic curve are defined as follows.

- Elliptic Curve Addition (ECA): If $P \neq Q$, point addition formula for computing $R = P \oplus Q = (x_R, y_R)$ is given as:

$$\lambda = \frac{y_Q - y_P}{x_Q - x_P},$$
$$\begin{cases} x_R = \lambda^2 - x_P - x_Q, \\ y_R = \lambda(x_P - x_R) - y_P. \end{cases}$$

- Elliptic Curve Doubling (ECD): If $P = Q$, point doubling formula for computing $R = P \oplus Q = [2]P = (x_R, y_R)$ is given as follows:

$$\lambda = \frac{3x_P^2 + a}{2y_P},$$
$$\begin{cases} x_R = \lambda^2 - 2x_P, \\ y_R = \lambda(x_P - x_R) - y_P. \end{cases}$$

- Elliptic curve Scalar Multiplication (SCM): If $P \neq \mathcal{O}$, then let $s \in \mathbb{Z}$, point scalar multiplication formula for calculating $R = [s]P$ as:

$$R = [s]P = \underbrace{P \oplus P \oplus \cdots \oplus P}_{s\text{-1 times additions.}}.$$

B. Pairing

Let E be an elliptic curve over $\mathbb{F}_p$, the subgroups $\mathbb{G}_1$ and $\mathbb{G}_2$ are defined as follows [15]:

$$\begin{cases} \mathbb{G}_1 = E[r] \cap \text{Ker}(\pi_p - [1]), \\ \mathbb{G}_2 = E[r] \cap \text{Ker}(\pi_p - [p]), \end{cases}$$

where π_p is a frobenius map.

A pairing e is a map to $\mathbb{G}_T$, defined as:

$$e : \mathbb{G}_2 \times \mathbb{G}_1 \rightarrow \mathbb{G}_T,$$

which has the following properties:

- Bilinear map:
  If rational points $g_1 \in \mathbb{G}_1$, and $g_2, g_2' \in \mathbb{G}_2$, and integers $\alpha, \beta \in \mathbb{Z}_p$, then:

$$e(g_1, g_2 \oplus g_2') = e(g_1, g_2) \cdot e(g_1, g_2'),$$
$$e([\alpha]g_1, [\beta]g_2) = e([\beta]g_1, [\alpha]g_2) = e(g_1, g_2)^{\alpha \cdot \beta}.$$

- Non-degenerate:
  If $g_1 \neq \mathcal{O}$ and $g_2 \neq \mathcal{O}$, then:

$$e(g_1, g_2) \neq 1.$$

  Keep in mind that the pairing e is known as a symmetric pairing if $\mathbb{G}_1 = \mathbb{G}_2$. Otherwise, it is referred to as an asymmetric pairing.

C. Sextic twist

Let two elliptic curves:

$$\begin{cases} E : y^2 = x^3 + b & \text{over } \mathbb{F}_{p^{12}}, \\ E' : y^2 = x^3 + bz & \text{over } \mathbb{F}_{p^2}, \end{cases}$$

where z is quadratic non-residue and cubic non-residue over $\mathbb{F}_{p^2}$.

The twist $\phi : E' \rightarrow E$ is defined as follows:

$$\phi : E' \rightarrow E, \qquad (x, y) \mapsto (z^{-\frac{1}{3}}x, z^{-\frac{1}{2}}y).$$

[1] *ELiPS*. Information Security laboratory Okayama University. [Online]. Available: https://github.com/ISecOkayamaUniv/ELiPS

D. Pairing-Based Cryptography (PBC)

The GMP library served as the foundation for the PBC library, an open source library carrying out the essential mathematical operations in pairing-based cryptosystems [17]. Speed and portability are crucial considerations as the PBC library is intended to serve as the foundation for pairing-based cryptosystem implementations. It offers functions like pairing computation and elliptic curve arithmetic [1], [17].

In PBC, which utilizes symmetric pairing, let $\mathbb{G}_1$ be an additive group over an elliptic curve and $\mathbb{G}_T$ be a multiplicative cyclic group. Both groups $\mathbb{G}_1$ and $\mathbb{G}_T$ have order r [1]. The pairing operation is defined as:

$$e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T.$$

There are eight different parameter types available in PBC. In each case, the curve group has a group order of 160-bit. Type A is known to be the fastest pairing and is suitable for cryptosystems where the group size is not a critical factor [11]. However, this type only provides an 80-bit security level and is vulnerable to multiple attacks [11]–[13]. Type A utilizes a supersingular curve, which is defined as follows:

$$E : y^2 = x^3 + x.$$

E. Overview of CP-ABE

CP-ABE is an encryption scheme that provides fine-grained access control over encrypted data. In CP-ABE, data is encrypted based on a set of attributes, and access to the encrypted data is granted based on predefined access policies associated with those attributes [1], [18]. This approach allows for flexible and customizable access control, where data owners can define specific attributes required for decryption [1], [19].

CP-ABE offers several advantages in scenarios where access control needs to manage carefully. It enables data sharing among multiple users or organizations while ensuring that the data can only be accessed by those with the necessary credentials. The usage of CP-ABE is particularly relevant in cloud services, Internet of Things environments, and scenarios involving sensitive data storage and communication [20]–[22]. By leveraging attribute-based encryption, CP-ABE offers robust protection of data confidentiality and privacy [22]. It allows for secure data sharing, collaboration, and compliance with regulatory requirements.

The CP-ABE algorithm primarily relies on hash-to-curve and pairing procedures, comprising four main components [1]:

1) Setup: It mainly uses pairing and exponentiation operations. This phase begins by generating the $\mathbb{G}_1$ and $\mathbb{G}_T$ groups, where $\mathbb{G}_1$ has a generator g and both groups have an order r. A bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$. Random values α and $\beta \in \mathbb{Z}_r$. Next, the master key MK and public key PK are calculated as follows [1]:

$$\begin{aligned} MK &= (\beta, g^{\alpha}),\\ PK &= (\mathbb{G}_1, g, h, f, v), \end{aligned}$$

where $h = g^{\beta}$, $f = g^{1/\beta}$, $v = e(g, g)^{\alpha}$.

2) Key generation: This phase includes scalar multiplication and hash-to-curve operations. The program takes the master key MK as well as the attribute set $A = \{att_1, att_2, ...\}$ as input. It proceeds to calculate the secret key SK, which is associated with the attribute set A. The algorithm selects a random value $\gamma \in \mathbb{Z}_r$, and for each attribute $i \in A$, it selects a random value $\gamma_i \in \mathbb{Z}_r$. A hash function $\mathcal{H} = \{0, 1\}^* \to \mathbb{G}_1$ is utilized. Subsequently, the secret key SK is computed as [1]:

$$\begin{aligned} SK = (D = g^{(\alpha+\gamma)/\beta},&\\ &\{D_i = g^{\gamma} \cdot \mathcal{H}(i)^{\gamma_i}, D'_i = g^{\gamma_i}\}_{\forall i \in A}). \end{aligned}$$

3) Encryption: It primarily utilizes scalar multiplication and hash-to-curve operations. The data is encrypted using the tree structure policy $\mathcal{T}$. The function $ind(t)$ returns the value for node t, while the function $par(t)$ returns the parent node of t in the tree. For each node t, a polynomial q_t is chosen. The process chooses a random value $s \in \mathbb{Z}_r$, starting with the R node, setting $q_R(0) = s$. Then, for every $t \in \mathcal{T}, q_t(0) = q_{par(t)}(ind(t))$. The leaf nodes in $\mathcal{T}$ are denoted as $\mathcal{L}$, and the function $att(t)$ provides the attribute value of each node in $\mathcal{T}$. The message is encrypted using the access policy structure $\mathcal{T}$, as follows [1]:

$$\begin{aligned} CT = (\mathcal{T}, \tilde{C} = M \cdot e(g, g)^{\alpha \cdot s}, C = h^{s},&\\ &\{C_l = g^{q_l(0)}, C'_l = \mathcal{H}(att(l))^{q_l(0)}\}_{\forall l \in \mathcal{L}}). \end{aligned}$$

4) Decryption: This stage primarily employs pairing and multiplication operations. The decryption procedure takes the secret key SK as well as the ciphertext CT and calculates the plaintext M. The program computes $dec\_node(CT, SK, t)$, which receives CT, SK, and node t as input. If t is a leaf node, the attribute of node t is obtained as $i = att(t)$. Then, $dec\_node(CT, SK, t)$ is computed as [1]:

$$dec\_node(CT, SK, t) = \begin{cases} \frac{e(D_i, C_t)}{e(D'_i, C'_t)} & \text{if } i \in A,\\ null & \text{if } i \notin A. \end{cases}$$

The $dec\_node(CT, SK, t)$ function operates on leafless node t as follows: For each child node c of t, the algorithm calls $dec\_node(CT, SK, c)$ and stores the result in F_c. A_t is a list of nodes c, where $F_c \neq null$. If no such set exists, the function returns $null$. Otherwise, the following calculation is performed [1]:

$$\begin{aligned} &\text{Let: } k = ind(c), \qquad A'_t = \{ind(c), \forall c \in A_t\},\\ &\quad \Delta_{k, A'_t(0)} = \prod_{j \in A'_t, j \neq k} \frac{-j}{k - j}, \end{aligned}$$

$$
\begin{aligned}
F_t &= \prod_{c \in A_t} F_c^{\Delta_{k, A'_t(0)}} \\
&= \prod_{c \in A_t} \left(e(g,g)^{\gamma \cdot q_c(0)}\right)^{\Delta_{k, A'_t(0)}} \\
&= \prod_{c \in A_t} \left(e(g,g)^{\gamma \cdot q_{par(c)}(ind(c))}\right)^{\Delta_{k, A'_t(0)}} \\
&= \prod_{c \in A_t} e(g,g)^{\gamma \cdot q_t(k) \cdot \Delta_{k, A'_t(0)}} \\
&= e(g,g)^{\gamma \cdot q_t(0)}.
\end{aligned}
$$

To decrypt the data, the algorithm first calls the $dec\_node(CT, SK, R)$ function. If the attributes A match the tree access structure $\mathcal{T}$, we set [1]:

$$
\begin{aligned}
\tilde{A} &= dec\_node(CT, SK, R) \\
&= e(g,g)^{\gamma \cdot q_R(0)} \\
&= e(g,g)^{\gamma \cdot s}.
\end{aligned}
$$

The ciphertext is decrypted using the following formula [1]:

$$
\frac{\tilde{C}}{\frac{e(C,D)}{\tilde{A}}} = M.
$$

F. Efficient Library for Pairing Systems (ELiPS)

The ELiPS is a specialized cryptographic library that focuses on efficient operations related to pairing-based cryptography. Such cryptography involves mathematical pairings between points on elliptic curves. It has gained attention for its applications in advanced cryptographic schemes such as identity-based encryption, attribute-based encryption, and functional encryption. The ELiPS library offers a range of functionalities, including point arithmetic operations, scalar multiplications, and pairing computations [14].

ELiPS is specifically designed to support bilinear pairing using the BLS-12 curve, providing a 128-bit security level [14]. It employs the BLS curve E with an embedding degree of $k = 12$. ELiPS uses $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_T$ be cyclic groups of order r, with r being 308-bit. The work by Daichi Hattori et al. [14] demonstrated that ELiPS provides slightly faster execution time compared to previous libraries while maintaining a parameter set that ensures a high security level.

Here, we present a comparative analysis between four prominent libraries in this research area: PBC, MCL, RELIC, and ELiPS libraries. We compare them in terms of hash-to-curve domain, pairing domain, exponentiation domain, scalar multiplication domain, and security level. Table I suggests that the ELiPS library offers a significant advantage in these domains. Therefore, based on this evidence, we intend to enhance the performance of CP-ABE by adopting the ELiPS library.

TABLE I
COMPARISON AMONG PAIRING LIBRARIES

Parameters		**PBC**	**MCL**	**RELIC**	**ELiPS**
Security level		80-bit	128-bit	128-bit	128-bit
Hash-to-curve		3.2 [ms]	0.3 [ms]	0.6 [ms]	0.1 [ms]
Pairing		0.9 [ms]	1.1 [ms]	2.6 [ms]	2.2 [ms]
Exponent		0.1 [ms]	0.8 [ms]	1.3 [ms]	0.6 [ms]
Scalar multiplication	$\mathbb{G}_1$	1.2 [ms]	0.3 [ms]	0.3 [ms]	0.2 [ms]
	$\mathbb{G}_2$	1.2 [ms]	0.4 [ms]	0.7 [ms]	0.5 [ms]

TABLE II
COMPARISON BETWEEN PBC AND ELIPS

	PBC	**ELiPS**
Operations in $\mathbb{G}_T$	Multiplication Exponent	Multiplication Exponent
Operations in $\mathbb{G}_1, \mathbb{G}_2$	Elliptic curve addition Elliptic curve doubling Scalar multiplication	Elliptic curve addition Elliptic curve doubling Scalar multiplication
Type of pairing	Symmetric	Asymmetric
Security level	80-bit	128-bit

III. PROPOSED SCHEME

In this section, we present the main procedures required to implement CP-ABE using ELiPS. PBC and ELiPS use several different operations, as shown in Table II. Therefore, we have designed three procedures to make ELiPS appropriate for CP-ABE. These procedures include generating $\mathbb{G}, \mathbb{G}_1$, and $\mathbb{G}_2$, as well as transforming asymmetric to symmetric pairing and modifying CP-ABE framework functions.

A. Generator g generation

Fig. 1 illustrates the process of generating $\mathbb{G}$. Since $\mathbb{G}_1$ and $\mathbb{G}_2$ are subgroups of $E[r]$, we can add the elements of $\mathbb{G}_1$ and $\mathbb{G}_2$, and the result is the element of $E[r]$.

Let $g_1 \in \mathbb{G}_1$ and $g_2 \in \mathbb{G}_2$ be generators, respectively, and let the group $\mathbb{G}$ be the group generated by $g_1 \oplus g_2$.

$$
\mathbb{G} = \langle g_1 \oplus g_2 \rangle. \tag{2}
$$

Then $\mathbb{G}$ is a subgroup of order r of $E[r] (\subset E(\mathbb{F}_{p^{12}}))$. Thus, addition and scalar multiplication can be defined over $\mathbb{G}$ in the same way as those on $E(\mathbb{F}_{p^{12}})$ [15].

B. Asymmetric to symmetric transformation

The authors successfully implemented Shirase's method [15] for converting asymmetric pairing to symmetric pairing. Let $g_1 \in \mathbb{G}_1$ and $g_2 \in \mathbb{G}_2$ be generator rational points. Then

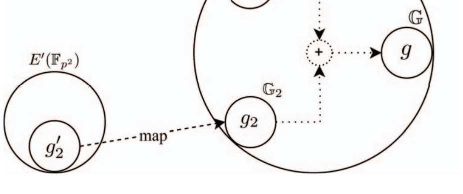

Fig. 1. The process for generating $\mathbb{G}, \mathbb{G}_1$, and $\mathbb{G}_2$.

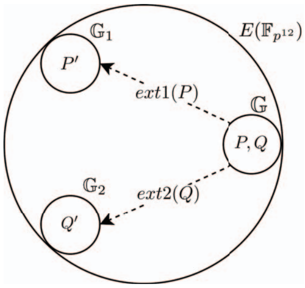

Fig. 2. Extraction of P' and Q' in transforming asymmetric pairing to symmetric pairing.

g is a generator point of $\mathbb{G}$, and this can be calculated as shown in (2). For two rational points $P, Q \in \mathbb{G}$ and we can use symmetric pairing $e_{sym}(Q, P)$ by defining a symmetric pairing as follows [15]:

$$e_{sym} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T.$$

Since ELiPS uses asymmetric pairing, the authors need to transform asymmetric pairing into symmetric pairing. This is done by extracting $\mathbb{G}_1$ and $\mathbb{G}_2$ from P and Q, respectively. Fig. 2 shows the concept of the extraction procedure. The transformation between asymmetric and symmetric pairing can be defined as [15]:

$$e_{sym}(Q, P) = e_{asy}(ext2(Q), ext1(P)).$$

Next, the authors provide a method for extracting $\mathbb{G}_1$ and $\mathbb{G}_2$ from $\mathbb{G}$.

Let $l = (p-1)^{-1} \pmod r$, where r is an order of subgroups $\mathbb{G}_1$ and $\mathbb{G}_2$. Then, the values of $ext1$ and $ext2$ can be calculated as follows [15]:

$$\begin{cases} ext1 = ([p] - \pi_p) \cdot [l], \\ ext2 = (\pi_p - [1]) \cdot [l]. \end{cases}$$

Let $g_1 \in \mathbb{G}_1$, and $g_2 \in \mathbb{G}_2$, and let $g = g_1 \oplus g_2$. Then,

$$\begin{cases} ext1(g) = g_1, \\ ext2(g) = g_2. \end{cases}$$

The symmetric pairing procedure in ELiPS is processed as follows:

- Algorithm first calls $e_{sym}(Q, P)$ function, where $Q = [m]g, P = [n]g$, and $m, n \in \mathbb{Z}_r$, and $P, Q \in \mathbb{G}$.
- Then, e_{sym} operates $P' = ext1(P)$ and $Q' = ext2(Q)$ functions, the algorithm is described as shown in Fig. 2. Afterward, the algorithm calls $e_{asy}(Q', P')$ to calculate asymmetric pairing.
- Asymmetric pairing uses Miller algorithm and final exponentiation algorithm to compute and return asymmetric pairing value.

C. CP-ABE algorithm modifications

We briefly present some modifications to enable ELiPS to work within the CP-ABE framework.

- **Setup:** The algorithm generates g_1' and g_2' over $E(\mathbb{F}_p)$ and $E'(\mathbb{F}_{p^2})$, respectively. Then, it maps g_1' and g_2' to g_1 and g_2 over $E(\mathbb{F}_{p^{12}})$. The generator point g is calculated using the formula $g = g_1 \oplus g_2$. Next, random $\alpha, \beta \in \mathbb{Z}_r$ are generated. The master key MK and public key PK are computed as follows:

$$MK = (\beta, [\alpha]g).$$
$$PK = (g, h = [\beta]g, f = [1/\beta]g, e(g,g)^{\alpha}).$$

- **Key generation:** The function takes the master key MK and attribute set A as input. It calculates the secret key SK as:

$$SK = (D = [(\alpha + \gamma)/\beta]g, \\ \{D_i = [\gamma]g \oplus [\gamma_i]\mathcal{H}(i), D_i' = [\gamma_i]g\}_{\forall i \in A}),$$

where random $\gamma, \gamma_i \in \mathbb{Z}_r$.
- **Encryption:** The algorithm takes the public key PK, message M, and tree structure policy $\mathcal{T}$ as input. The ciphertext is computed as follows:

$$CT = (\mathcal{T}, \tilde{C} = M \cdot e(g,g)^{\alpha \cdot s}, C = [s]h, \\ \{C_l = [q_l(0)]g, C_l' = [q_l(0)]\mathcal{H}(att(l))\}_{\forall l \in \mathcal{L}}),$$

where random $s \in \mathbb{Z}_r$, $\mathcal{L}$ is the leaf node set in $\mathcal{T}$.
- **Decryption:** The inputs for the procedure are ciphertext CT and secret key SK. It calls the $dec\_node(CT, SK, R)$ function to calculate $\tilde{A}$ as:

$$\tilde{A} = dec\_node(CT, SK, R) \\ = e(g,g)^{\gamma \cdot s}.$$

In this function, the algorithm calls the recursive $dec\_node(CT, SK, t)$ function to calculate the value $\tilde{A}$ and verify whether the secret key SK matches the access policy, where t is a leaf node, as follows:

$$dec\_node(CT, SK, t) = \begin{cases} \frac{e_{sym}(D_i, C_t)}{e_{sym}(D_i', C_t')} & \text{if } i \in A, \\ null & \text{if } i \notin A. \end{cases} \quad (3)$$

The original message is decrypted using the following formula:

$$\frac{\tilde{C} \cdot \tilde{A}}{e_{sym}(C, D)} = \frac{\tilde{C} \cdot \tilde{A}}{e_{asy}(ext2(C), ext1(D))} = M.$$

IV. Experimental evaluation

In this section, we evaluate the proposed method's performance and compare it with PBC-based CP-ABE.

TABLE III
EXPERIMENTAL ENVIRONMENTS

OS	Ubuntu 22.04.1 LTS (WSL2)
CPU	Intel(R) Core(TM) i7-6600U @ 2.60GHz
Memory	4 GB
Language	C
GMP version	6.2.1
GCC version	11.3.0
GCC optimization level	-O2

A. Experimental evaluation setup

Table III shows the devices and software used during the evaluation. In our experiments, we employed a data access authorization for administration procedures at the university level and attribute policy scenarios, as depicted in Fig. 3, which involve three entities:

- University administrator: Authority.
- President: Sender.
- Professors: Receiver.

Assuming the university president wishes to share private data exclusively with professors in the Faculty of Engineering, the president only encrypts the data once and shares the encrypted data with all intended recipients. Additionally, the president needs to define an access policy structure to determine who can decrypt the encrypted data. On the recipients' side, if their attributes satisfy the access policy, they can successfully decrypt the data; otherwise, they are unable to decrypt it.

B. Evaluation with two attributes

First, we employed two attributes to implement a data access authorization for the administration scenario. Next, we performed 10,000 executions to measure the computation time of setup, key generation, encryption, and decryption functions for both PBC-based CP-ABE and ELiPS-based CP-ABE, then we took the average values.

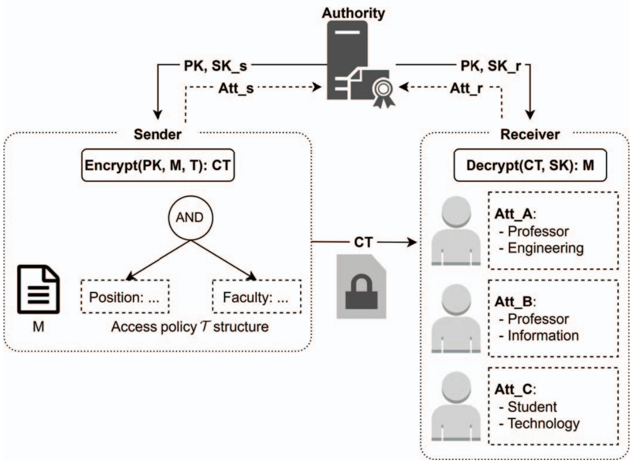

Fig. 3. An example of data access authorization for administrative procedures at the university level.

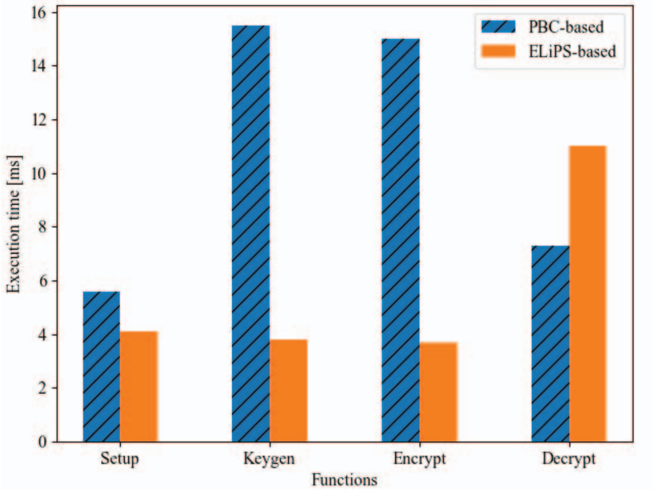

Fig. 4. A comparison between PBC-based CP-ABE and ELiPS-based CP-ABE in a two-attribute scenario.

Fig. 4 summarizes the comparison results. It shows that most of the CP-ABE functions in ELiPS-based CP-ABE perform faster than their counterparts in PBC-based CP-ABE. However, the decryption function is slower. It seems that the pairing operations are equivalent to the number of attributes of the attribute set A as in the sub section III-C. Further evaluation with increasing the attributes is needed.

C. Evaluation with increasing the number of attributes

After that, we conducted experiments with increasing the number of attributes, ranging from 2, 5, 10, 15, and 20 to implement the data access scenarios. Next, we ran 10,000 times to measure the computation time on each CP-ABE function, then took the average values. The experimental results are shown in Table IV.

The setup part of ELiPS-based CP-ABE exhibits a slight decrease in execution time, with a reduction of 28.17% compared to PBC-based CP-ABE. This reduction can be attributed to the efficient generation of the generator rational point g. The calculation cost of the setup part does not correlate with the number of attributes. Hence, the setup's computation time seems constant in several experimental scenarios.

In the case of key generation and encryption, the results show that the proposed ELiPS-based CP-ABE takes much less time than the PBC-based CP-ABE. The speed of the key generation and encryption functions in the ELiPS-based CP-ABE is more than 3.99 times faster compared to the PBC-based CP-ABE. This reduction is achieved through efficient calculations in $\mathbb{F}_p$ and $\mathbb{F}_{p^2}$, as well as the computation of the hash map attribute procedure.

Regarding the decryption phase, the results indicate that the execution time of the proposed ELiPS-based decrypt function is slightly higher than that of the PBC-based function, with an increase of 2.40 times. The pairing has a major effect on

TABLE IV

Computation time [ms] in four main functions for several evaluation scenarios of PBC-based CP-ABE and ELIPS-based CP-ABE

No. of attributes	Setup		Key generation		Encryption		Decryption	
	PBC-based	*ELiPS-based*	*PBC-based*	*ELiPS-based*	*PBC-based*	*ELiPS-based*	*PBC-based*	*ELiPS-based*
5	5.6	4.1	29.9	7.5	28.9	7.5	11.6	24.5
10	5.6	4.1	54.8	13.3	54.5	13.5	19.4	45.6
15	5.6	4.1	80.3	19.6	80.0	19.9	27.3	68.4
20	5.6	4.1	105.0	25.4	103.7	25.8	34.9	90.5

the decryption part. As seen in (3) and the decryption time presented in Table IV, the computation time for decryption increases linearly as the attributes increase. The cost of calculating ELiPS-based decryption is higher than that of PBC-based decryption due to the ELiPS pairing function costs more to calculate than the PBC pairing function. On the other hand, the security levels for both ELiPS and PBC are different. Comparing them will be more appropriate when both PBC-based CP-ABE and ELiPS-based CP-ABE use the same security level.

V. Conclusion

The authors proposed an ELiPS-based CP-ABE with several modifications to adapt ELiPS into CP-ABE. Our proposed scheme enhances the performance and security level of the encryption system. It is also noticeable that most cases of the proposed ELiPS-based CP-ABE exhibit faster performance compared to the PBC-based CP-ABE, except for the decryption calculation, despite their different security levels. Future development will concentrate on improving the decryption calculation of the ELiPS-based CP-ABE. Furthermore, we will try to integrate our proposal into the blockchain system.

References

[1] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-Policy Attribute-Based Encryption," in *2007 IEEE Symposium on Security and Privacy (SP '07),* Berkeley, CA, USA, pp. 321-334, 2007. doi: 10.1109/SP.2007.11.

[2] D. A. Luong and J. H. Park, "Privacy-Preserving Identity Management System on Blockchain Using Zk-SNARK," *IEEE Access,* vol. 11, pp. 1840-1853, 2023. doi: 10.1109/ACCESS.2022.3233828.

[3] A. Aloufi, P. Hu, Y. Song, and K. Lauter, "Computing Blindfolded on Data Homomorphically Encrypted under Multiple Keys: An Extended Survey," *CoRR*, pp. 1-55, 2020. doi: 10.48550/arXiv.2007.09270.

[4] A. I. Sanka, M. Irfan, I. Huang, and R. C. C. Cheung, "A survey of breakthrough in blockchain technology: Adoptions, applications, challenges and future research," *Computer Communications,* vol. 169, pp. 179-201, 2021. doi: 10.1016/j.comcom.2020.12.028.

[5] A. Ouaddah, "A blockchain based access control framework for the security and privacy of IoT with strong anonymity unlinkability and intractability guarantees," in *Role of Blockchain Technology in IoT Applications*, Eds. Elsevier, ch. 8, pp. 211-258, 2019. doi: 10.1016/bs.adcom.2018.11.001.

[6] S. N. Khan, F. Loukil, C. Ghedira-Guegan, E. Benkhelifa, and A. Bani-Hani, "Blockchain smart contracts: Applications, challenges, and future trends," *Peer-to-Peer Networking and Applications,* vol. 14, pp. 2901–2925, 2021. doi: 10.1007/s12083-021-01127-0.

[7] N. Helil and K. Rahman, "CP-ABE Access Control Scheme for Sensitive Data Set Constraint with Hidden Access Policy and Constraint Policy," *Security and Communication Networks,* vol. 2017, pp. 1-13, 2017. doi: 10.1155/2017/2713595.

[8] S. Huda, A. Sudarsono and T. Harsono, "Secure data exchange using authenticated ciphertext-policy attributed-based encryption," in *2015 International Electronics Symposium (IES),* Surabaya, Indonesia, pp. 134–139, 2015. doi: 10.1109/ELECSYM.2015.7380829.

[9] S. Huda, A. Sudarsono, and T. Harsono, "Secure Communication and Information Exchange using Authenticated Ciphertext Policy Attribute-Based Encryption in Mobile Ad-hoc Network," *EMITTER International Journal of Engineering Technology,* vol. 4, no. 1, pp. 115–140, 2016. doi: 10.24003/emitter.v4i1.116.

[10] C. Marcolla, V. Sucasas, M. Manzano, R. Bassoli, F. H. P. Fitzek and N. Aaraj, "Survey on Fully Homomorphic Encryption, Theory, and Applications," in *Proceedings of the IEEE,* vol. 110, no. 10, pp. 1572-1609, 2022. doi: 10.1109/JPROC.2022.3205665.

[11] B. Lynn. Stanford University. *PBC Library - Pairing-Based Cryptography*. (2006). Accessed: Jul. 15, 2023. [Online]. Available: https://crypto.stanford.edu/pbc

[12] J. W. Bos, M. E. Kaihara, T. Kleinjung, A. K. Lenstra, and P. L. Montgomery, "On the Security of 1024-bit RSA and 160-bit Elliptic Curve Cryptography," Cryptology ePrint Archive, Paper 2009/389, pp. 1-19, 2009. [Online]. Available: https://eprint.iacr.org/2009/389

[13] E. Barker, "Recommendation for Key Management," National Institute of Standards and Technology, 2020. doi: 10.6028/NIST.SP.800-57pt1r5.

[14] D. Hattori, Y. Takahashi, T. Tatara, Y. Nanjo, T. Kusaka, and Y. Nogami, "An Optimal Curve Parameters for BLS12 Elliptic Curve Pairing and Its Efficiency Evaluation," in *2021 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW),* Penghu, Taiwan, pp. 1–2, 2021. doi: 10.1109/ICCE-TW52618.2021.9602941.

[15] M. Shirase, "Symmetric Pairing on Ordinary Elliptic Curves," in *Information Processing Society of Japan Symposium Proceedings,* Japan, pp. 357–362, 2010.

[16] Y. Nanjo, M. Shirase, Y. Kodera, T. Kusaka, and Y. Nogami, "Efficient Final Exponentiation for Cyclotomic Families of Pairing-Friendly Elliptic Curves with Any Prime Embedding Degrees," *International Journal of Networking and Computing,* vol. 12, no. 2, pp. 317–338, 2022. [Online]. Available: http://www.ijnc.org/index.php/ijnc/article/view/285

[17] J. Bethencourt, A. Sahai, and B. Waters. The University of Texas. *Advanced Crypto Software Collection*. (2006). Accessed: Jul. 15, 2023. [Online]. Available: https://acsc.cs.utexas.edu/cpabe

[18] B. Lynn, "On the implementation of Pairing-Based Cryptosystems," Ph.D. dissertation, Stanford University, 2007. [Online]. Available: https://crypto.stanford.edu/pbc/thesis.pdf

[19] K. P. Praveen, K. P. Syam, and P. J. A. Alphonse, "Attribute based encryption in cloud computing: A survey, gap analysis, and future directions," *Journal of Network and Computer Applications,* vol. 108, pp. 37–52, 2018. doi: 10.1016/j.jnca.2018.02.009.

[20] C. I. Fan, Y. F. Tseng, and C. C. Feng, "CCA-Secure Attribute-Based Encryption Supporting Dynamic Membership in the Standard Model," in *2021 IEEE Conference on Dependable and Secure Computing (DSC),* Aizuwakamatsu, Fukushima, Japan, pp. 1–8, 2021. doi: 10.1109/DSC49826.2021.9346247.

[21] B. Chandrasekaran, R. Balakrishnan, and Y. Nogami, "TF-CPABE: An efficient and secure data communication with policy updating in wireless body area networks," *ETRI Journal,* vol. 41, no. 4, pp. 465–472, 2019. doi: 10.4218/etrij.2018-0320.

[22] J. Li, Y. Zhang, J. Ning, X. Huang, G. S. Poh, and D. Wang, "Attribute Based Encryption with Privacy Protection and Accountability for CloudIoT," *IEEE Transactions on Cloud Computing,* vol. 10, no. 2, pp. 762–773, 2022. doi: 10.1109/TCC.2020.2975184.