sensors

MDPI

Article

Identity-Based Proxy Re-Encryption Scheme Using Fog Computing and Anonymous Key Generation

Han-Yu Lin, Tung-Tso Tsai *, Pei-Yih Ting and Yan-Rong Fan

Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung 202, Taiwan
* Correspondence: tttsai@mail.ntou.edu.tw

Abstract: In the fog computing architecture, a fog is a node closer to clients and responsible for responding to users' requests as well as forwarding messages to clouds. In some medical applications such as the remote healthcare, a sensor of patients will first send encrypted data of sensed information to a nearby fog such that the fog acting as a re-encryption proxy could generate a re-encrypted ciphertext designated for requested data users in the cloud. Specifically, a data user can request access to cloud ciphertexts by sending a query to the fog node that will forward this query to the corresponding data owner who preserves the right to grant or deny the permission to access his/her data. When the access request is granted, the fog node will obtain a unique re-encryption key for carrying out the re-encryption process. Although some previous concepts have been proposed to fulfill these application requirements, they either have known security flaws or incur higher computational complexity. In this work, we present an identity-based proxy re-encryption scheme on the basis of the fog computing architecture. Our identity-based mechanism uses public channels for key distribution and avoids the troublesome problem of key escrow. We also formally prove that the proposed protocol is secure in the IND-PrID-CPA notion. Furthermore, we show that our work exhibits better performance in terms of computational complexity.

Keywords: identity based; proxy re-encryption; fog computing; anonymous; cloud

Citation: Lin, H.-Y.; Tsai, T.-T.; Ting, P.-Y.; Fan, Y.-R. Identity-Based Proxy Re-Encryption Scheme Using Fog Computing and Anonymous Key Generation. *Sensors* **2023**, *23*, 2706. https://doi.org/10.3390/s23052706

Received: 6 February 2023
Revised: 24 February 2023
Accepted: 27 February 2023
Published: 1 March 2023

1. Introduction

Cloud computing was first introduced by Chellappa [1] who hoped to store data in remote servers via networks. In 2006, Amazon brought in the so-called Amazon Web Services (AWS) to seek more business opportunities. Since then, such services have been widely developed and spread out. Many companies like Microsoft and Google are gradually competing for the cloud computing market due to its advantages in reducing costs and improving productivity. Cloud services have also become an inevitable part of our daily life. When using cloud services, we have to pay attention to the safeguard of Internet security [2] by employing all kinds of cryptographic techniques [3–6].

Clouds are not only for storage; the deployment model influences managing and owning the cloud, and the location and users of the cloud. According to the location of stored data and how the technologies are deployed and consumed, we can classify cloud service models into three kinds, as follows:

(i) **Public Cloud:** It is usually constructed by third-party cloud service companies (such as Google (Mountain View, CA, USA), Azure (Redmond, WA, USA), etc.). Users can purchase storage space from service providers and the latter will be responsible for system maintenance, which helps with reducing unnecessary user costs. Yet, the security is low owing to uncontrollable cloud environments.
(ii) **Private Cloud:** It is constructed by the individual company and hence has high security and privacy. However, it requires self-maintenance and the cost is relatively high.

(iii) **Hybrid Cloud:** It combines the advantages of public and private clouds and can separately store data by its confidentiality. Nevertheless, it is also relatively difficult to manage and maintain.

According to the definition given in NIST SP 800-145, the cloud service model is also categorized into the following three kinds:

(i) **Software as a Service (SaaS):** It is the most common model in which users can utilize all kinds of interfaces (including web-based or program-based) to acquire resources and web services [7] such as stream media platforms running on cloud infrastructure. The advantage of this model is that users do not have to be responsible for controlling or maintaining the cloud infrastructure, such as communication networks [8], operating systems, storage, and applications.

(ii) **Platform as a Service (PaaS):** In this model, the cloud service provider is responsible for providing application development platforms such as storage capacity, computing resources, programming languages, libraries and related development tools, etc. Users can utilize these tools to deploy consumer-created application programs on the cloud infrastructure and they do not have to control or maintain the cloud infrastructure.

(iii) **Infrastructure as a Service (IaaS):** The cloud service provider supplies users with all kinds of storage, computing, and network resources, and users can utilize these infrastructures to deploy their own platforms and application programs. The advantage of this model is that users do not have to control or maintain the cloud infrastructure, but have the control over their deployed applications, storage, and operating systems.

Fog computing in IoT environments [9–11] is an extension of cloud computing and was first addressed by the research of Stolfo et al. [12], who attempt to protect the cloud data security [13] with the assistance of fog computing. Bonomi et al. [14] viewed the fog as a cloud closer to the ground and users. The architecture of fog computing can also benefit by sharing cloud data. A proxy re-encryption (PRE) scheme [15] is a commonly adopted ciphertext sharing protocol in which a ciphertext intended for Alice can be re-encrypted into another ciphertext designated for Bob by a proxy. When we combine the PRE scheme with fog computing, a fog would act as the proxy to carry out the cloud ciphertext transformation process, so as to reduce the network transmission latency. However, the privacy of cloud ciphertexts in the transformation must be further assured and the fog node (proxy) should learn nothing about the ciphertext.

2. Related Work

In 2010, Luo et al. [16] developed a ciphertext-policy attribute-based PRE scheme using AND-Gates policy. In particular, their policy supports multi-value attributes, negative attributes, and wildcards. They also showed that their mechanism fulfills the property of unidirectionality, non-interactivity, and multi-use. Moreover, in their scheme, the encryptor can decide if the ciphertext can be re-encrypted.

Considering the property of keyword search, in 2012, Fang et al. [15] proposed a chosen-ciphertext secure-anonymous-conditional PRE with keyword search. That is, they provided the PRE mechanism with the property of keyword search. Additionally, they gave the CCA security definition of conditional PRE schemes and showed that their protocol satisfies such a definition. Wang et al. [17] further introduced a constrained PRE scheme with a conjunctive keyword search. Specifically, their mechanism is both single-hop and unidirectional.

In 2013, Liang et al. [18] addressed a ciphertext-policy attribute-based PRE with chosen-ciphertext security assuming the hardness of the decisional q-parallel bilinear Diffie–Hellman exponent problem. In this mechanism, a ciphertext with respect to a given access policy is able to be re-encrypted into one in relation to another access policy. Their scheme is suitable for any monotonic access structure. They proved the security of their scheme in the random oracle model. Considering the queries from intra-domain and inter-domain in a cloud computing scenario, Han et al. [19] proposed an identity-based PRE scheme. Their scheme is secure against collusion attacks and the access permission could be made by the

data owner, rather than by the central authority. However, the computational complexity of their scheme is high.

In 2014, Liang et al. [20] presented an adaptively CCA-secure ciphertext-policy attribute-based PRE for cloud data sharing. Their work integrated the dual system encryption technology with selective proof technique to achieve the adaptive CCA security in the standard model. Additionally, their work supports any monotonic access structures.

To improve the security of sharing data using QR codes, Akhil et al. [21] combined the PRE mechanism with QR code applications. Using QR codes to share data among different users is a commonly utilized approach. However, it is easily altered during transmission, since the format of QR codes is only readable by machines.

For improving the security of cloud storage, in 2018, Zeng and Choo [22] introduced a new conditional PRE scheme that is also known as the sender-specified PRE, i.e., SS-PRE, since their scheme only allows the proxy to transform the ciphertexts of the designated sender to the delegatee. They also present the formal definition of their SS-PRE scheme and prove its security in the standard model.

In order to share data securely in the cloud, in 2020, Zhang et al. [23] proposed an identity-based data storage scheme combining the architecture of fog computing. In their scheme, the fog node sends the request of the data user to both the cloud and the data owner. If the requested data user is non-revoked and has access privilege, the data owner will delegate the fog node to perform the cloud ciphertext re-encryption process. The fog node then forwards the re-encrypted ciphertext to the data user for decryption.

Considering the security of message transmission in a group, in 2021, Xiong et al. [24] presented a so-called puncturable PRE scheme on the identity-based cryptosystem. In particular, there is a message server that carries out the ciphertext re-encryption for all users and thus the computational efforts of the user sides are released. Yet, the message server plays a crucial role in the system performance and might become an obvious attacking target.

In 2022, Lin et al. [25] improved Zhang et al.'s scheme [23] by eliminating some security flaws. They also showed that their enhanced scheme maintains the properties to revoke invalid users and generate private keys anonymously. Although their work has improved security, which is provably secure in the random oracle model, the computational complexity is still high. Motivated by this challenging problem, we propose a more efficient PRE scheme based on the study of Lin et al. [25]. Up to the present, there have been several PRE mechanisms [26–32] proposed for different applications. However, only a few works [19,23,25] take the issue of cloud computing or fog computing scenarios into consideration. We compare the proposed mechanism with these schemes and show the computational advantage of ours in a later section.

The main contribution of this study is to propose an identity-based PRE scheme for the fog computing scenario and using the technique of anonymous key generation. In the proposed system, we use public channels for key distribution and avoid the troublesome problem of key escrow. In addition, the decision of access privilege for cloud ciphertexts is controlled by the data owner, rather than by the central authority. Moreover, we demonstrate that the proposed protocol is not only IND-PrID-CPA secure, but also has lower computational costs.

3. Preliminaries

Let the symbols ($\boldsymbol{G}_1$, $\boldsymbol{G}_2$) be two multiplicative groups and p is a prime order of both groups. We express e: $\boldsymbol{G}_1 \times \boldsymbol{G}_1 \rightarrow \boldsymbol{G}_2$ as a symmetric bilinear pairing. The properties of e are listed as follows:

(i) Bilinearity:

Given a group element g in $\boldsymbol{G}_1$ and two integers a, b in Z_p, we have $e(g^a, g^b) = e(g, g)^{ab}$.

(ii) Non-degeneracy:

There are group elements A, B in $\boldsymbol{G}_1$ such that $e(A, B) \neq 1$.

(iii) Computability:

Given two group elements A, B in $\boldsymbol{G}_1$, the value $e(A, B)$ can be efficiently computed.

Decisional Bilinear Diffie–Hellman (DBDH) Problem and Assumption

Let the problem instance be $(g, g^x, g^y, g^z, e(g, g)^{xyz}, C)$ in which (g, g^x, g^y, g^z) are elements in $\boldsymbol{G}_1$ while $(e(g, g)^{xyz}, C)$ are elements in $\boldsymbol{G}_2$; the DBDH problem has to decide if the equality $e(g, g)^{xyz} = C$ holds or not. Its assumption asserts that the chance of any adversary running in polynomial-time to solve the DBDH problem is insignificant.

4. Proposed IB-PRE-FCAK Scheme

Before introducing the proposed identity-based proxy re-encryption scheme using fog computing and anonymous key generation (short for IB-PRE-FCAK), we first address the system model and composed algorithms.

4.1. System Model

We illustrate the system model for our proposed IB-PRE-FCAK scheme in Figure 1, and it is mainly composed of three levels. Among the hierarchy, the top level is a cloud server that can be viewed as a data repository center storing ciphertexts. The middle level is a collection of fog nodes that process requests from users and transmit ciphertexts to the cloud server. The third level consists of the data owner and the data user. The former generates ciphertexts to be uploaded to the cloud server while the latter can request access to cloud ciphertexts. Note that both the ciphertext uploading and downloading processes are assisted by the fog nodes. In particular, the data owner can authorize the fog node to perform the ciphertext re-encryption procedure for sharing cloud ciphertexts with other data users. Moreover, there is also a private key generation center (PKG) responsible for issuing private keys to all users.

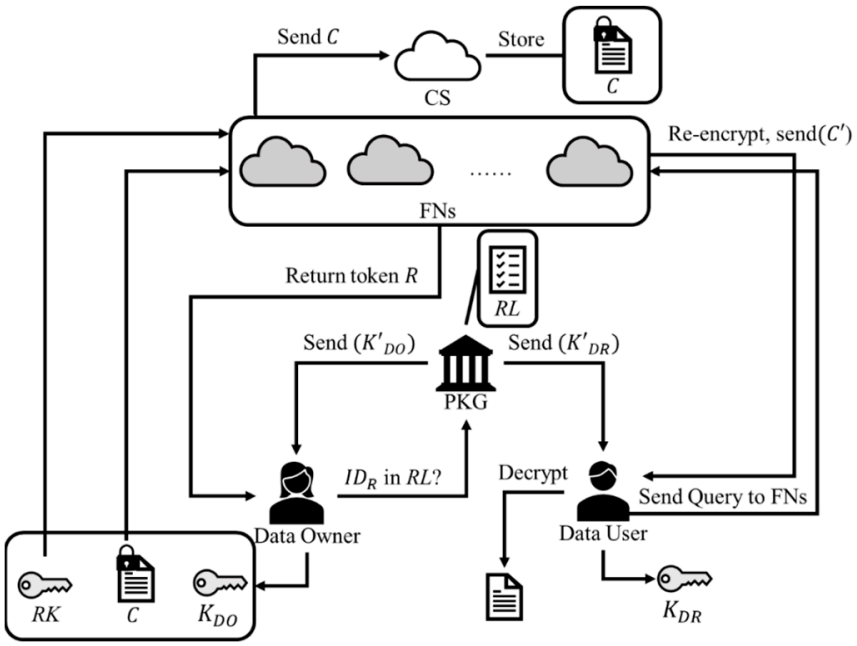

Figure 1. System model of the IB-PRE-FCAK scheme.

4.2. Algorithms

The IB-PRE-FCAK scheme can be divided into several subroutines, i.e., Setup, KeyExtract, Enc, Tkgen, RKgen, Re-Enc, and Dec. We define the parameters and corresponding outputs of these subroutines as follows:

- **Setup**(1^l)**:** This subroutine utilizes the value l as a security parameter and returns the system public information Φ along with the master secret key Msk.
- **KeyExtract**(Φ, Msk, ID)**:** This subroutine utilizes three input parameters (Φ, Msk, ID) where ID is a user identity, and performs an interactive process to return the private key K_{ID} associated with ID.
- **Enc**(Φ, ID, m, SK)**:** This subroutine utilizes four input parameters (Φ, ID, m, SK) where (m, SK) separately represents a plaintext and a symmetric encryption key. It returns a ciphertext C of the plaintext m under the key SK.
- **Tkgen**(Φ, ID_u, K_{IDu}, C_{ind})**:** This subroutine utilizes four input parameters (Φ, ID_u, K_{IDu}, C_{ind}) where C_{ind} is the name of data category, and then returns a token $T_{u,ind}$.
- **RKgen**(Φ, ID_u, K_{IDo}, $T_{u,ind}$)**:** This subroutine utilizes five input parameters (Φ, ID_u, K_{IDo}, $T_{u,ind}$) where ID_o is the data owner. It returns either the symbol of error $\perp$ or a corresponding key $RK_{o,u,ind}$ for re-encryption.
- **Re-Enc**(Φ, ID_u, C, $RK_{o,u,ind}$)**:** This subroutine utilizes four input parameters (Φ, ID_u, C, $RK_{o,u,ind}$) and then returns a re-encrypted ciphertext C'.
- **Dec**(Φ, K_{ID}, C^*)**:** This subroutine utilizes three input parameters (Φ, K_{ID}, C^*) where C^* could be C or C', and then returns a plaintext m.

4.3. Construction

We introduce a concrete construction based on the previously defined subroutines. First, some utilized symbols are defined as Table 1:

Table 1. Symbol notations.

Notation	Description
l	security value
$\boldsymbol{G}_1$, $\boldsymbol{G}_2$	groups of prime order p
g	a generator of $\boldsymbol{G}_1$
e	a bilinear map satisfying e: $\boldsymbol{G}_1 \times \boldsymbol{G}_1 \rightarrow \boldsymbol{G}_2$
s	master secret key
Q	master public key satisfying $Q = g^s$
RL	revocation list
h_1, h_2, h_3	collision-resistant hash functions
Φ	system public information
SK	symmetric key
$E(\cdot)/D(\cdot)$	symmetric encryption/decryption function
$\{r_1, r_2, r_3\}$	ciphertext
C_{ind}	data category name
(w_1, w_2, w_3)	re-encryption key
$(r_1', r_2, r_3, r_4', r_5')$	re-encrypted ciphertext

- **Setup:** Taking the value l as a security value, the PKG chooses $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ groups of prime order p and both are multiplicative. Let the symbol g denote a generator in group $\boldsymbol{G}_1$ and the notation e be a bilinear map written as e: $\boldsymbol{G}_1 \times \boldsymbol{G}_1 \rightarrow \boldsymbol{G}_2$. Msk determined by the PKG is a random value $s \in Z_p^*$, and its corresponding master public key (Mpk) is calculated as $Q = g^s$. There is also a user revocation list, i.e., RL, maintained by the PKG. Whenever ID_i has to be revoked, the PKG renews RL as $RL' = RL \cup \{ID_i\}$. Three collision-resistant hash functions are defined as follows:

  h_i: $\{0, 1\}^* \rightarrow \boldsymbol{G}_1$, for i = 1 and 2
  h_3: $\boldsymbol{G}_2 \rightarrow \boldsymbol{G}_1$

Except for *Msk*, all the other parameters could be viewed as the system public information Φ.

– **KeyExtract:** For obtaining his/her private key, a user ID_i randomly chooses integers d_i, $k_i \in Z_p^*$ and computes

$$D_i = g^{di}, \tag{1}$$

$$H'_i = D_i \cdot h_1(ID_i \parallel k_i) \tag{2}$$

The values (ID_i, H'_i) are delivered to the PKG. After receiving it, the PKG derives

$$K'_i = (H'_i \cdot h_2(ID_i \parallel ID_{PKG}))^s, \tag{3}$$

and sends K'_i back to ID_i. Consequently, ID_i is able to calculate his private key as

$$K_i = K'_i/(Q)^{di} = (h_1(ID_i \| k_i) \cdot h_2(ID_i \| ID_{PKG}))^s \tag{4}$$

With the following equality, the correctness of K_i can be easily verified.

$$e(K_i, g) = e(h_1(ID_i \parallel k_i) \cdot h_2(ID_i \parallel ID_{PKG}), Q) \tag{5}$$

– **Enc:** Let $m = (m_1, m_2, \ldots, m_n)$ be a plaintext to be encrypted and $SK \in \boldsymbol{G}_2$ a chosen symmetric key. A data owner ID_o then selects an integer $z \in_R Z_p^*$ to calculate

$$r_1 = SK \cdot e(Q, (h_1(ID_o \parallel k_o) \cdot h_2(ID_o \parallel ID_{PKG}))^z), \tag{6}$$

$$r_2 = g^z, \tag{7}$$

$$r_3 = (E(SK, m_1), E(SK, m_2), \ldots, E(SK, m_n)) \tag{8}$$

where the notation $E(\cdot)$ denotes the symmetric encryption function.

Here, the ciphertext C is composed of $\{r_1, r_2, r_3\}$. Next, the data owner sends (ID_o, C) along with the data category name C_{ind} to the adjacent fog. It stores $(ID_o, C_{ind}, r_1, r_2)$ in the local repository and further transmit (ID_o, C_{ind}, r_3) to the cloud server.

– **Tkgen:** For accessing the cloud data with respect to the data category name C_{ind}, a data user ID_u randomly selects an integer $r \in_R Z_p^*$ to compute

$$R = g^r, \tag{9}$$

and delivers his request (ID_u, C_{ind}, R) to the adjacent fog. It then searches for the record $(ID_o, C_{ind}, r_1, r_2)$ from the local repository and further forwards the token $T_{u,ind} = (ID_u, R)$ to the associated data owner ID_o.

– **RKgen:** Upon obtaining the token $T_{u,ind} = (ID_u, R)$, the data owner ID_o asks the PKG to check if the maintained user revocation list RL contains ID_u. If it does, an error symbol $\perp$ is sent to the requested data user ID_u via the assistance of the fog. Or else, ID_o randomly selects two random numbers $t, y \in Z_p^*$ and computes

$$w_1 = Q^t \tag{10}$$

$$w_2 = \frac{K_o w_1}{h_3(e(h_2(ID_u || ID_{PKG}) R^y, \ Q))} \tag{11}$$

$$w_3 = e(g^y, Q) \tag{12}$$

Next, ID_o sends the re-encryption key $RK_{o,u,ind} = (w_1, w_2, w_3)$ to the fog node.

– **Re-Enc:** Upon receiving $RK_{o,u,ind}$, the fog re-encrypts the original ciphertext C as C' by setting

$$r_1{}' = r_1 \cdot e(w_1, r_2), \tag{13}$$

$$r_4{}' = w_2, \tag{14}$$

$$r_5{}' = w_3 \tag{15}$$

At last, the re-encrypted ciphertext C' consisting of $(r_1{}', r_2, r_3, r_4{}', r_5{}')$ is sent back to ID_u. Note that the partial ciphertext r_3 can be retrieved from the cloud storage.

– **Dec:** In the case that the data owner ID_o wants to access his/her original ciphertext $C = (r_1, r_2, r_3)$, he/she can derive the symmetric key by computing

$$SK = \frac{r_1}{e(r_2,\ K_o)} \tag{16}$$

Then, the plaintext can be decrypted as

$$m = (m_1, m_2, \ldots, m_n) = (D(SK, r_{3,1}), D(SK, r_{3,2}), \ldots, D(SK, r_{3,n})). \tag{17}$$

where $D(\cdot)$ is a symmetric decryption function.

The correctness of Equation (16) can be verified as follows. From the right side of Equation (16), it can be derived that

$$\begin{aligned} \frac{r_1}{e(r_2, K_o)} &= \frac{SK \cdot e\big(Q,\ (h_1(ID_O||k_O)h_2(ID_O||ID_{PKG}))^z\big)}{e\big(g^z,\ (h_1(ID_O||k_O)h_2(ID_O||ID_{PKG}))^s\big)} \\ &= \frac{SK \cdot e(g^{sz},\ (h_1(ID_O||k_O)h_2(ID_O||ID_{PKG})))}{e\big(g^z,\ (h_1(ID_O||k_O)h_2(ID_O||ID_{PKG}))^s\big)} \\ &= SK \end{aligned}$$

Whenever a data user ID_u obtains a re-encrypted ciphertext C' that is composed of $(r_1{}', r_2, r_3, r_4{}', r_5{}')$, he/she first utilizes his/her own private key K_u to calculate

$$X = r_4{}' \cdot h_3\left(\frac{r'^r_5 \cdot e(K_u,\ g)}{e(h_1(ID_u||k_u),\ \ Q)}\right) \tag{18}$$

$$SK = \frac{r_1{}'}{e(X,\ \ r_2)}. \tag{19}$$

Consequently, the plaintext m can be decrypted with the symmetric key SK by Equation (17). We give the derivations of Equation (19) below. Our first step is to simplify Equation (18):

$$\begin{aligned} X &= r_4{}' \cdot h_3\left(\frac{r'^r_5 \cdot e(K_u,\ g)}{e(h_1(ID_u||k_u), Q)}\right) \\ &= \frac{K_o w_1}{h_3(e(h_2(ID_u||ID_{PKG})R^y, Q))} h_3\left(\frac{e(g^{yr},\ Q)e(K_u, g)}{e(h_1(ID_u||k_u), Q)}\right) \\ &= \frac{K_o w_1}{h_3(e(h_2(ID_u||ID_{PKG})R^y, Q))} h_3\left(\frac{e(g^{yr},\ Q)e(h_1(ID_u||k_u)h_2(ID_u||ID_{PKG}), g^s)}{e(h_1(ID_u||k_u), Q)}\right) \\ &= \frac{K_o w_1}{h_3(e(h_2(ID_u||ID_{PKG})R^y, Q))} h_3(e(R^y h_2(ID_u||ID_{PKG}),\ Q)) \\ &= K_o w_1 \end{aligned}$$

Next, we could rewrite Equation (19) as

$$\begin{aligned}\frac{r_1{}'}{e(X,\ r_2)} &= \frac{r_1 \cdot e(w_1,\ r_2)}{e(K_o w_1, r_2)} \\ &= \frac{SK \cdot e\big(Q,\ (h_1(ID_O||k_O)h_2(ID_O||ID_{PKG}))^z\big)e\big(Q^t, g^z\big)}{e\big((h_1(ID_O||k_O)h_2(ID_O||ID_{PKG}))^s Q^t, g^z\big)} \\ &= \frac{SK \cdot e\big((h_1(ID_O||k_O)h_2(ID_O||ID_{PKG}))^s Q^t, g^z\big)}{e\big((h_1(ID_O||k_O)h_2(ID_O||ID_{PKG}))^s Q^t, g^z\big)} \\ &= SK\end{aligned}$$

5. Formal Model and Security Proof

The fundamental security for any encryption scheme is confidentiality. There is also a well-defined security model for PRE schemes. We adopt this security model to demonstrate formally the security of the proposed IB-PRE-FCAK protocol. Namely, we initially review the security definition of IND-PrID-CPA, i.e., indistinguishability against adaptively chosen identity and chosen plaintext attacks. Then, we demonstrate that the proposed construction fulfills the secure notion of IND-PrID-CPA by utilizing the proof techniques of random oracle models.

The concept of this security proof is a technique of proof by contradiction. That is, we first assume that there is an adversary who is able to break the proposed scheme under the adaptively chosen identity and chosen plaintext attacks. Then, we can take the advantage of this adversary to break a well-known intractable cryptographic assumption with non-negligible advantage. Since there is no efficient polynomial-time algorithm that could solve any well-known cryptographic assumption, we conclude that our initial assumption is wrong, which also completes the whole security proof.

(IND-PrID-CPA) *In the following interactive game between a probabilistic adversary A and a challenger B, if the former does not have a non-negligible advantage to defeat the latter in polynomial-time, we say that an IB-PRE-FCAE scheme fulfills the security requirement of indistinguishability under the attacks of adaptively chosen identity and chosen-plaintext:*

Setup: At first, the challenger B invokes the Setup(1^l) subroutine to obtain system public information Φ along with the master secret key Msk. The adversary A can only learn the public information Φ.

Phase 1: A is allowed to adaptively invoke the following queries:

- *KeyExtract Queries:* A can query the private key for his chosen identity.
- *RKgen Queries:* A can query the re-encryption key for his chosen (ID_o, ID_u, C_{ind}) in which ID_u has to be a non-revoked data user and C_{ind} is the name of data category.

Challenge: A chooses the identity of ID^* as an object and prepares the plaintext $m^* = (m_1{}^*, m_2{}^*, \ldots, m_n{}^*)$. Let ($SK_0$, SK_1) be symmetric keys with an identical length. Then, B flips a bit bt and then creates a challenge ciphertext $C^* = (r_1{}^*, r_2{}^*, r_3{}^*)$ in relation to (ID^*, m^*, SK_{bt}) for A.

Phase 2: Given the ciphertext C^*, the adversary A goes on to invoke previous queries based on the following limits:

- The KeyExtract query with respect to ID^*, i.e., the target identity, is prohibited.
- Any RKgen query for the identities (ID^*, ID_u) or (ID_o, ID^*) is prohibited.
- A can invoke at most q_{ke} KeyExtract and q_{rk} RKgen queries.

Guess: After invoking enough queries, A returns a bit bt'. We say that A wins this game, provided that $bt' = bt$. Therefore, we can express A's advantage as $Adv(A) = \mid \Pr[bt' = bt'] - 1/2 \mid$.

Using the techniques of random oracle proof models, we prove that the proposed IB-PRE-FCAK scheme satisfies the security notion of IND-PrID-CPA as Theorem 1.

Theorem 1. *Provided thatthe DBDH assumption holds, the proposed IB-PRE-FCAK scheme satisfies the security requirement of indistinguishability under adaptively chosen identity and chosen plaintext attacks (IND-PrID-CPA). Specifically, an algorithm B breaking the DBDH problem with non-negligible advantage ε' can be created by utilizing a probabilistic adversary A that is able to break the IND-PrID-CPA security of the proposed IB-PRE-FCAK scheme with non-negligible advantage ε in polynomial-time. To be precise, the non-negligible advantage ε' can be expressed as*

$$\varepsilon' \geq \frac{\varepsilon}{e(q_{ke} + q_{rk} + 1)}$$

where q_{ke} and q_{rk} are the maximum numbers of KeyExtract and RKgen queries, respectively.

Proof. Given an instance $(g, g^a, g^b, g^c, e(g, g)^{abc}, F)$ of DBDH, we build an algorithm B to judge if $e(g, g)^{abc} = F$ holds or not by taking the adversary A as subroutine. In the following interactions, B is responsible for responding to various queries submitted by A.

Setup: At first, the challenger B invokes the Setup(1^l) subroutine to obtain system public information Φ. Let (h_1, h_2) be random oracles and h_3 a collision-resistant hash function. B further sets $Mpk = Q = g^a$, i.e., Msk is implicitly specified as the value a unknown to B.

Phase 1: A is allowed to adaptively invoke the following queries:

- $h_1(ID_i \parallel k_i)$ *query:* In this query, B first searches the maintained h_1-list for a matched record. Or else, he selects a bit η such that $\Pr[\eta = 1] = \tau$. The value τ would be derived subsequently. Whenever $\eta = 0$, B returns the value $J_1 = (g^b)^{v1}$ in which $v_1 \in Z_p{}^*$. Otherwise, J_1 is computed as g^{v1}. The maintained h_1-list is also renewed by adding the record $(ID_i, k_i, \eta, v_1, J_1)$.
- $h_2(ID_i \parallel ID_{PKG})$ *query:* In this query, B first searches the maintained h_2-list for a matched record. Or else, he returns the value $J_2 = g^{v2}$ in which $v_2 \in Z_p{}^*$. The maintained h_2-list is also renewed by adding the record $(ID_i, ID_{PKG}, v_2, J_2)$.
- *KeyExtract query:* In response to the KeyExtract(ID_i) query, B tries to determine the corresponding records $(ID_i, k_i, \eta, v_1, J_1)$ and $(ID_i, ID_{PKG}, v_2, J_2)$ in h_1-list and h_2-list, respectively. (If one datum exists, B could directly invoke the two queries to create records.) As long as $\eta = 1$, B aborts; or else, the return value is computed as $K_i = Q^{(v_1 + v_2)}$
- *RKgen query:* In response to the RKgen(ID_o, ID_u, C_{ind}) query in which ID_u is a non-revoked user, B obtains the private key K_{IDo} by invoking the KeyExtract(ID_o) query and checks the record $(ID_i, k_i, \eta, v_1, J_1)$ kept in the h_1-list. As long as $\eta = 0$, B aborts. Or else, B chooses random numbers $r, t, y \in Z_p{}^*$ and calculates $R = g^r$, $w_1 = Q^t$, $w_2 = \frac{K_o Q^t}{h_3(e(h_2(ID_u \| ID_{PKG}))R^y, Q))}$, $w_3 = e(g^y, Q)$. Thus, the returned re-encryption key $RK_{o,u,ind}$ is composed of (w_1, w_2, w_3).

Challenge: A chooses the identity of ID^* as an object and prepares the plaintext $m^* = (m_1{}^*, m_2{}^*, \ldots, m_n{}^*)$. Let (SK_0, SK_1) be symmetric keys with an identical length. Then, B flips a bit bt and then creates a challenge ciphertext $C^* = (r_1{}^*, r_2{}^*, r_3{}^*)$ in relation to (ID^*, m^*, SK_{bt}) for A as follows:

Step 1 Suppose that the $h_1(ID^* \parallel k^*)$ query has been made. As long as $\eta^* = 1$, B directly aborts.
Step 2 Define $h_2(ID^* \parallel ID_{PKG}) = g^{v2}$ in which $v_2 \in Z_p{}^*$.
Step 3 Set the partial ciphertext $r_2{}^* = g^c$.
Step 4 Determine the value v_1 of the record $(ID^*, k^*, \eta^*, v_1, J_1)$ in the h_1-list and calculate

$$r_1{}^* = SK_{bt} \cdot F^{v_1^{-1}} \cdot e\left(Q, \left(g^c\right)^{v_2}\right),$$

$$r_3{}^* = (E(SK_{bt}, m_1{}^*), E(SK_{bt}, m_2{}^*), \ldots, E(SK_{bt}, m_n{}^*)).$$

Consequently, the returned challenge ciphertext is $C^* = (r_1{}^*, r_2{}^*, r_3{}^*)$.

Phase 2: Given the ciphertext C^*, the adversary A goes on to invoke queries based on the previous limitations.

Guess: After invoking enough queries, A returns a bit bt'. In case that $bt' = bt$, B directly returns 1, meaning that $F = e(g, g)^{abc}$. Otherwise, the value 0 is outputted instead.

Analysis: In these simulation processes, it can be observed that when F is equivalent to $e(g, g)^{abc}$, the prepared challenge ciphertext C^* is a legal one. According to the initial assumption, A would have the non-negligible advantage to break the proposed IB-PRE-FCAK scheme provided that the simulated ciphertext C^* is valid. That is to say, we know that $Adv(A) = | \Pr[bt' = bt] - 1/2 | \geq \varepsilon$. Yet, when F is not equivalent to $e(g, g)^{abc}$, the advantage for A to output a correct bit bt' is not superior, which implies that $\Pr[bt' = bt] = 1/2$. Therefore, the chance for B to solve the problem of DBDH could be written as

$$\begin{aligned} &| \Pr[(g, g^a, g^b, g^c, e(g, g)^{abc}) = 1] - \Pr[(g, g^a, g^b, g^c, F) = 1] | \\ &\geq | (1/2 + \varepsilon) - 1/2 | \cdot \Pr[\text{Good}] \\ &= \varepsilon \cdot \Pr[\text{Good}] \end{aligned}$$

where Pr[Good] represents the probability event that B never aborts during the game interaction processes. To calculate Pr[Good], we further consider the following several cases:

Pr[¬KeyExtract]: the likelihood that B never aborts in any KeyExtract query;

Pr[¬RKgen]: the likelihood that B never aborts in any RKgen query;

Pr[¬Challenge]: the likelihood that B never aborts in the challenge phase.

In the first case of a KeyExtract query, B aborts as long as the bit η in the corresponding entry of the h_1-list equals 0. Thus, we can learn that $\Pr[\neg\text{KeyExtract}] \leq \tau^{q_{ke}}$. Likewise, as for the second case of an RKgen query, we also know that B aborts on the condition that the bit $\eta = 0$, which indicates that $\Pr[\neg\text{RKgen}] \leq \tau^{q_{rk}}$. In the third case that B might abort only when the bit η^* for the chosen identity ID^* is equivalent to 1. Therefore, we could derive that $\Pr[\neg\text{Challenge}] \leq (1 - \tau)$. Putting all the three independent probability events together, we further obtain

$$\begin{aligned} \Pr[\text{Good}] &= \Pr[\neg\text{KeyExtract}] \cdot \Pr[\neg\text{RKgen}] \cdot \Pr[\neg\text{Challenge}] \\ &\leq (\tau)^{q_{ke}} (\tau)^{q_{rk}} (1 - \tau) \\ &= (\tau)^{q_{ke} + q_{rk}} (1 - \tau). \\ &= \frac{1}{e\left(q_{pk} + q_{pr} + 1\right)} \end{aligned}$$

To maximize the value of Pr[Good], we set τ to be $1 - \frac{1}{q_{ke} + q_{rk} + 1}$ such that $\Pr[\text{Good}] = \frac{1}{e\left(q_{pk} + q_{pr} + 1\right)}$ becomes the greatest value, where e denotes the base of natural logarithm. As a result, we claim that the constructed algorithm B has a non-negligible advantage $\varepsilon' \geq \frac{\varepsilon}{e\left(q_{pk} + q_{pr} + 1\right)}$ to break the DBDH problem. □

6. Efficiency and Comparison

We made some efficiency comparisons with related protocols [19,23,25] in terms of several time-consuming computations. For convenience, the simulation environments are listed in Table 2 and the compared computation is also converted into approximate running time in Table 3. The detailed evaluation results are summarized in Table 4 and Figure 2.

Table 2. Simulation environments.

Item	Environment
Processor	Intel Core 2 Duo @ 2.1 Ghz
Memory size	2 GB
Operating system	Linux Ubuntu version 9.1
Software	PBC library [33]

Table 3. Computation and approximate running time.

Item / Computation	Notation	Running Time
Bilinear pairing	C_0	5.883 ms
Exponentiation over G_1	C_1	0.736 ms
Exponentiation over G_2	C_2	0.142 ms

Table 4. Evaluation of computational cost.

Scheme / Phase	Han et al. [19]	Zhang et al. [11]	Lin et al. [13]	Proposed
Setup cost	$6C_1$	$2C_1$	$2C_1$	C_1
KeyExtract cost	$5C_0 + 5C_1$	$5C_1$	$5C_1$	$3C_1$
Enc cost	$3C_0 + 3C_1 + C_2$	$C_0 + 2C_1 + C_2$	$C_0 + 2C_1 + C_2$	$C_0 + 2C_1$
Tkgen cost	$2C_1$	C_1	C_1	C_1
RKgen cost	$5C_0 + 4C_1 + C_2$	$2C_1$	$2C_0 + 3C_1 + C_2$	$2C_0 + 3C_1$
Re-Enc cost	0	C_0	C_0	C_0
Dec cost by ID_o	$2C_0$	$2C_0$	$2C_0$	C_0
Dec cost by ID_u	$2C_0 + 2C_1$	$2C_0 + C_1$	$4C_0 + C_2$	$3C_0 + C_2$

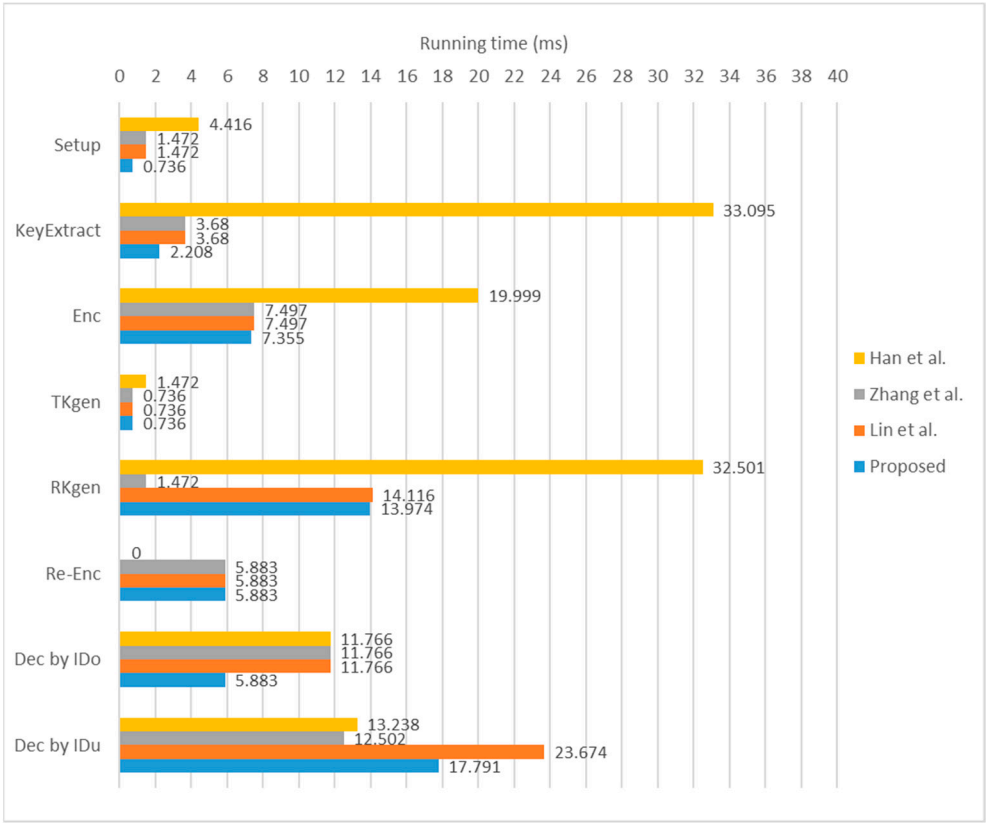

Figure 2. Comparison of approximate running time [11,13,19].

Although Han et al.'s scheme is cost-free in the Re-Enc phase, their scheme has the highest computation costs in the Setup, KeyExtract, Enc, TKgen, and RKgen phases. Zhang et al.'s scheme has the lowest computation costs in both the RKgen and the Dec by ID_u phases. Lin et al.'s scheme incurs higher computation cost in the Dec by ID_u phase. As a whole, the proposed protocol exhibits optimal computational costs in the Setup, KeyExtract, Enc, and Dec by ID_o phases.

7. Conclusions

Fog-based applications have received much attention in recent years due to their advantages in fast response time and more bandwidth savings. A fog-enabled proxy re-encryption scheme allows a fog node to perform the ciphertext re-encryption process, so as to share cloud ciphertexts to desired data users. In this paper, we propose an identity-based proxy re-encryption scheme taking the advantage of fog computing. Specifically, the proposed scheme removes the necessity for a fully trusted system authority, as the private key of each user is not generated by the system authority solely. Therefore, it is unnecessary to establish a secure channel for distributing private keys in the proposed scheme. The access privilege of cloud ciphertexts can be determined independently by the data owner. As for security, we adopt the security notion of IND-PrID-CPA to formally prove that the proposed mechanism is able to withstand the adaptive adversary in random oracle models. In the performance analyses, we also demonstrate that our work is efficient in the processes of Setup, KeyExtract, Enc, and Dec, when compared with related protocols.

Author Contributions: Writing—original draft, H.-Y.L.; writing—review and editing, T.-T.T. and P.-Y.T.; Resources, Y.-R.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Ministry of Science and Technology of the Republic of China under the contract number MOST 110-2221-E-019-041-MY3.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chellappa, R. Intermediaries in cloud-computing: A new computing paradigm. In Proceedings of the 1997 INFORMS Annual Meeting, San Diego, CA, USA, 4–5 May 1997; pp. 26–29.
2. Chen, Z. Research on Internet security situation awareness prediction technology based on improved RBF neural network algorithm. *J. Comput. Cogn. Eng.* **2022**, *1*, 103–108.
3. Gutub, A.; Gong, M. Boosting image watermarking authenticity spreading secrecy from counting-based secret-sharing. *CAAI Trans. Intell. Technol.* **2022**, 1–13. [CrossRef]
4. Pavithran, P.; Mathew, S.; Namasudra, S.; Srivastava, G. A novel cryptosystem based on DNA cryptography, hyperchaotic systems and a randomly generated Moore machine for cyber physical systems. *Comput. Commun.* **2022**, *188*, 1–12. [CrossRef]
5. Mahmood, Z.H.; Ibrahem, M.K. New fully homomorphic encryption scheme based on multistage partial homomorphic encryption applied in cloud computing. In Proceedings of the 2018 1st Annual International Conference on Information and Sciences (AiCIS), Fallujah, Iraq, 20–21 November 2018; pp. 182–186.
6. Dostalek, L.; Safarik, J. Strong password authentication with AKA authentication mechanism. In Proceedings of the 2017 International Conference on Applied Electronics (AE), Pilsen, Czech Republic, 5–6 September 2017; pp. 1–6.
7. Sarkar, M.; Saha, K.; Namasudra, S.; Roy, P. An efficient and time saving web service based android application. *SSRG Int. J. Comput. Sci. Eng.* **2015**, *2*, 18–21.
8. Kumari, S.; Kumar, R.; Kadry, S.; Namasudra, S.; Taniar, D. Maintainable stochastic communication network reliability within tolerable packet error rate. *Comput. Commun.* **2021**, *178*, 161–168. [CrossRef]
9. Wani, A.; Revathi, S.; Khaliq, R. SDN-based intrusion detection system for IoT using deep learning classifier (IDSIoT-SDL). *CAAI Trans. Intell. Technol.* **2021**, *6*, 281–290. [CrossRef]
10. Bajaj, K.; Sharma, B.; Singh, R. Comparative analysis of simulators for IoT applications in fog/cloud computing. In Proceedings of the 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 7–9 April 2022; pp. 983–988.
11. Tseng, C.L.; Lin, F.J. Extending scalability of IoT/M2M platforms with fog computing. In Proceedings of the 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), Singapore, 5–8 February 2018; pp. 825–830.
12. Stolfo, S.J.; Salem, M.B.; Keromytis, A.D. Fog computing: Mitigating insider data theft attacks in the cloud. In Proceedings of the 2012 IEEE Symposium on Security and Privacy Workshops, San Francisco, CA, USA, 24–25 May 2012; pp. 125–128.
13. Verma, R.; Kumari, A.; Anand, A.; Yadavalli, V.S.S. Revisiting shift cipher technique for amplified data security. *J. Comput. Cogn. Eng.* **2022**, 1–7. [CrossRef]
14. Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S. Fog computing and its role in the internet of things. In Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, Helsinki, Finland, 17 August 2012; pp. 13–16.

15. Fang, L.; Susilo, W.; Ge, C.; Wang, J. Chosen-ciphertext secure anonymous conditional proxy re-encryption with keyword search. *Theor. Comput. Sci.* **2012**, *462*, 39–58. [CrossRef]
16. Luo, S.; Hu, J.; Chen, Z. Ciphertext policy attribute-based proxy re-encryption. In Proceedings of the International Conference on Information and Communications Security, Barcelona, Spain, 15–17 December 2010; pp. 401–415.
17. Wang, X.A.; Huang, X.; Yang, X.; Liu, L.; Wu, X. Further observation on proxy re-encryption with keyword search. *J. Syst. Softw.* **2012**, *85*, 643–654. [CrossRef]
18. Liang, K.; Fang, L.; Susilo, W.; Wong, D.S. A ciphertext-policy attribute-based proxy re-encryption with chosen-ciphertext security. In Proceedings of the IEEE 2013 5th International Conference on Intelligent Networking and Collaborative Systems (INCoS), Xi'an, China, 9–11 September 2013; pp. 552–559.
19. Han, J.; Susilo, W.; Mu, Y. Identity-based data storage in cloud computing. *Future Gener. Comput. Syst.* **2013**, *29*, 673–681. [CrossRef]
20. Liang, K.; Au, M.H.; Susilo, W.; Wong, D.S.; Yang, G.; Yu, Y. An adaptively CCA-secure ciphertext-policy attribute-based proxy re-encryption for cloud data sharing. In Proceedings of the International Conference on Information Security Practice and Experience, Fuzhou, China, 5–8 May 2014; pp. 448–461.
21. Akhil, N.V.; Vijay, A.; Kumar, D.S. QR code security using proxy re-encryption. In Proceedings of the 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Nagercoil, India, 18–19 March 2016; pp. 1–5.
22. Zeng, P.; Choo, K.R. A new kind of conditional proxy re-encryption for secure cloud storage. *IEEE Access* **2018**, *6*, 70017–70024. [CrossRef]
23. Zhang, J.; Bai, W.; Wang, X. Identity-based data storage scheme with anonymous key generation in fog computing. *Soft Comput.* **2020**, *24*, 5561–5571. [CrossRef]
24. Xiong, H.; Wang, L.; Zhou, Z.; Zhao, Z.; Huang, X.; Kumari, S. Burn after reading: Adaptively secure puncturable identity-based proxy re-encryption scheme for securing group message. *IEEE Internet Things J.* **2021**, *9*, 11248–11260. [CrossRef]
25. Lin, H.Y.; Tsai, T.T.; Ting, P.Y.; Chen, C.C. An improved ID-based data storage scheme for fog-enabled IoT environments. *Sensors* **2022**, *22*, 4223. [CrossRef] [PubMed]
26. Chandini, A.G.; Basarkod, P.I. A robust blockchain architecture for electronic health data using efficient lightweight encryption model with re-encryption scheme. In Proceedings of the 2022 IEEE International Conference on Data Science and Information System (ICDSIS), Hassan, India, 29–30 July 2022; pp. 1–6.
27. Hu, H.; Cao, Z.; Dong, X. Autonomous path identity-based broadcast proxy re-encryption for data sharing in clouds. *IEEE Access* **2022**, *10*, 87322–87332. [CrossRef]
28. Yang, H.; Li, L.; Yang, C. A fine-grained certificateless conditional proxy broadcast re-encryption scheme without pairing. In Proceedings of the 2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 17–19 June 2022; pp. 1414–1423.
29. Devaki, K.; Leena, J.L. Re-encryption model for multi-block data updates in network security. In Proceedings of the 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 9–11 May 2022; pp. 1331–1336.
30. Yang, C.C.; Tso, R.; Liu, Z.Y.; Hsu, J.C.; Tseng, Y.F. Improved proxy re-encryption scheme with equality test. In Proceedings of the 2021 16th Asia Joint Conference on Information Security (AsiaJCIS), Seoul, Republic of Korea, 19–20 August 2021; pp. 37–44.
31. Khashan, O.A. Parallel proxy re-encryption workload distribution for efficient big data sharing in cloud computing. In Proceedings of the 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), Virtual, 27–30 January 2021; pp. 554–559.
32. Yao, S.; Dayot, R.V.J.; Kim, H.J.; Ra, I.H. A novel revocable and identity-based conditional proxy re-encryption scheme with ciphertext evolution for secure cloud data sharing. *IEEE Access* **2021**, *9*, 42801–42816. [CrossRef]
33. PBC Library, the Pairing-Based Cryptography Library. Available online: http://crypto.stanford.edu/pbc/ (accessed on 28 December 2022).