Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4462134, 10 pages
https://doi.org/10.1155/2022/4462134

*Research Article*

# Public Key Encryption with Authorized Equality Test on Outsourced Ciphertexts for Cloud-Assisted IoT in Dual Server Model

**Meng Zhao,[1] Yong Ding,[1,2] Shijie Tang,[3] Hai Liang,[1] and Huiyong Wang[4]**

[1]*Guangxi Key Laboratory of Cryptography and Information Security, School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin, China*
[2]*Cyberspace Security Research Center, Pengcheng Laboratory, Shenzhen, China*
[3]*School of Electronic Engineering and Automation, Guilin University of Electronic Technology, Guilin, China*
[4]*School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin, China*

Correspondence should be addressed to Yong Ding; stone_dingy@126.com

Received 24 August 2021; Revised 6 October 2021; Accepted 15 October 2021; Published 20 January 2022

Academic Editor: Ximeng Liu



In cloud computing, the outsourced data face many privacy and security threats. To allow the cloud server to perform comparison, search, and classification on outsourced ciphertexts while simultaneously providing privacy guarantee, the encryption method that supports the ciphertext equality test is considered as a promising way. Users are able to authorize the cloud server to conduct the ciphertext equality test, so that two ciphertexts can be determined whether they encrypt the same message without being decrypted. In this process, users do not need to retrieve, decrypt, and then perform comparison on data; thus, the computing and communication efficiency can be greatly improved, and the privacy of user data can be guaranteed at the cloud server side. However, existing encryption schemes supporting authorized ciphertext equality test in the single server model cannot resist the keyword guessing attacks, and the solutions in the dual server model do not provide simultaneous authorization on two servers. To address these issues, this paper proposes a public key encryption scheme supporting authorized equality test on ciphertexts in the dual server model (PKE-AUT), where the primary server and secondary server must get the authorization from users before performing a sequential equality test on ciphertexts. Security and performance analysis demonstrate that the proposed PKE-AUT scheme not only guarantees the privacy of user data and authorization but also is practical in cloud-assisted IoT-related applications.

## 1. Introduction

In recent years, the cloud computing and Internet of Things (IoT) technologies have developed rapidly and become widely used. By leveraging the powerful computing capability and massive storage resources of cloud servers, the collected IoT data can be outsourced to cloud servers to save local storage and computing resources [1]. However, to guarantee the privacy of the user's sensitive information, the data should be encrypted before being outsourced, so that only the data in ciphertext format would be stored at the cloud server [2, 3]. Data encrypted with classic cryptographic schemes does not support equality test, keyword search, calculation, and other operations on ciphertexts, so that users need to download their outsourced data to the local and then complete the corresponding operations after decryption. Thus, this process would bring huge computing and communication burdens to users, while failing to reflect the advantages of cloud computing services [4, 5].

To enable equality test on outsourced ciphertexts, many public key encryption schemes [6–8] and identity-based encryption schemes [9–12] have been proposed in the single server model. After the cloud server received the authorization from the user, it is able to perform the equality test on outsourced ciphertexts or some related operations such as encrypted data classification [13, 14] based on the equality

test, without decryption. However, since these solutions were proposed in the single cloud server model, the authorized cloud server would be able to launch keyword guessing attacks on outsourced ciphertexts to infer user data [4, 15], which causes damage to the privacy of users. Specifically, the cloud server is able to generate ciphertexts on many messages using the public keys of some users. Note that the cloud server should hold the authentication from these users. In this way, the cloud server can compare the generated ciphertexts with the stored ones, which would leak the message information if some pairs of ciphertexts are matched.

To resist the above-mentioned keyword guessing attacks faced by outsourced ciphertexts under the single server model, Wu et al. [15] proposed an identity-based encryption scheme under the dual server model for data classification in the mobile health social network. With their scheme, the user can authorize the primary server to generate relevant intermediate parameters, and the secondary server can further determine whether the two ciphertexts encrypted the same plaintext according to these intermediate parameters. These two servers would not collude to launch the attacks on outsourced user data. During the execution of their solution, the secondary server without obtaining the legal authorization of the user can perform the equality test on ciphertexts from the intermediate results generated by the primary server.

*1.1. Our Contributions.* This paper proposes a public key encryption scheme supporting the authorized equality test on outsourced ciphertexts (PKE-AUT) in the dual server mode. Similar to [15], the primary server and secondary server would not collude for compromising the confidentiality of outsourced data. Without authorization from the data user, both servers are unable to perform any operation on outsourced ciphertexts. After obtaining the same authorization from the data user, the primary server and secondary server sequentially perform the equality test on outsourced ciphertexts; that is, the authorized primary server produces and sends the intermediate parameters to the secondary server, then the authorized secondary server can complete the equality test procedure.

In the proposed PKE-AUT scheme, the authorizations generated for two servers are the same. The authorization is encrypted by the data user, so that only the primary server and secondary server are able to decrypt the authorization with their privacy keys, respectively; in this way, the computing costs for producing authorization can be reduced and the privacy of authentication can be protected during transmission. Security analysis shows that the proposed PKE-AUT scheme can guarantee the privacy of outsourced ciphertexts in two phases before and after the primary and secondary servers are authorized. Efficiency analysis demonstrates that the proposed PKE-AUT scheme is suitable for IoT-related applications.

*1.2. Related Works.* Many studies have been conducted on the authorized equality test on ciphertexts in different application scenarios. Yang et al. [6] introduced the first probabilistic public key encryption scheme with equality test on ciphertexts (PKEET), where anyone without authorization was able to check whether the ciphertexts generated with different public keys encrypt the same data. Thus, when deployed in cloud computing, their scheme allows an unauthorized cloud server to compare the outsourced ciphertexts of different users.

Since Yang et al.'s work [6], many encryption schemes supporting the authorized equality test on ciphertexts in the single server model have been proposed [7, 16], such that the cloud server can only compare the ciphertexts after being authorized. In [17], Tang designed an all-or-nothing encryption scheme, where the cloud can test the ciphertexts only after being independently authorized by their owners. In [18], Lee et al. analyzed the security of Huang et al.'s construction [19] and presented a security-enhanced scheme. An identity-based encryption scheme with equality test on ciphertexts (IBEET) was constructed in [20], which combines the PKEET and identity-based encryption technologies. Lee et al. [21] studied the semigeneric constructions of PKEET and IBEET and proved their security under the Computational Diffie-Hellman (CDH) and Computational Bilinear Diffie-Hellman (CBDH) assumptions, respectively.

The mechanism of the equality test on ciphertexts has been used in equi-join in relational databases and secure deduplication of encrypted data. Pang and Ding [22] investigated equi-join across encrypted tables in the database in private key setting, where for an outsourced database, the user is able to control which data tables the cloud server can perform equi-join according to some data fields by issuing authorization. Then, controlled equi-join for encrypted databases in the public key setting was considered in [23]. Also, the technology of the equality test on ciphertexts was employed by Cui et al. [24] and Yan et al. [25] in achieving secure deduplication on outsourced data in clouds, without sacrificing data privacy.

Postquantum encryption schemes supporting the equality test on ciphertexts have also received attention from researchers. Le et al. [26] proposed the first lattice-based signcryption scheme with equality test on ciphertexts in the standard model, which was proven secure against insider attacks. Susilo et al. [27] designed an efficient postquantum IBEET scheme with smaller ciphertext and public key size, which enjoys CCA2 security. Nguyen et al. [10] presented a lattice-based IBEET scheme in the standard model, which supports flexible authorization for equality test so that the user is able to control the comparison of their ciphertexts with others.

*1.3. Paper Organization.* The remainder of this paper is organized as follows. Section 2 introduces the preliminaries for the proposed PKE-AUT scheme. Section 3 describes the system model and security requirements for the PKE-AUT system in the dual server model. A description of our PKE-AUT scheme is presented in Section 4, followed by the security and performance analysis in Section 5. Section 6 concludes the paper.

## 2. Preliminaries

This section reviews the bilinear groups, the Computational Diffie-Hellman (CDH) problem and the Computational Bilinear Diffie-Hellman (CBDH) problem.

*2.1. Bilinear Groups.* Let $G = \langle g \rangle$ and $G_T$ be two cyclic groups of prime order $q$. The map $\hat{e} : G \times G \longrightarrow G_T$ is a bilinear pairing if it satisfies the following conditions:

(i) *Bilinearity*: for any $g_1, g_2 \in_R G$ and $a, b \in_R Z_q^*$, we have

$$\hat{e}\left(g_1^a, g_2^b\right) = \hat{e}(g_1, g_2)^{ab}. \quad (1)$$

(ii) *Nondegeneracy*: there exists $g_1, g_2 \in G$ such that

$$\hat{e}(g_1, g_2) \neq 1. \quad (2)$$

(iii) *Computability*: for $g_1, g_2 \in_R G$, there is an efficient algorithm to compute $\hat{e}(g_1, g_2)$

*2.2. Complexity Assumptions.* The security of our construction relies on the following two assumptions.

*CDH assumption*. Let $G = \langle g \rangle$ be a cyclic group of prime order $q$. Given a tuple $(g, g^a, g^b)$ where $a, b \in_R Z_q^*$, there is no probabilistic ploynomial-time algorithm $\mathcal{A}$ to compute $g^{ab}$ with nonnegligible probability.

*CBDH assumption*. Let $G = \langle g \rangle$ and $G_T$ be two cyclic groups of prime order $q$ and satisfy bilinear pairing $\hat{e} : G \times G \longrightarrow G_T$. Given a tuple $(g, g^a, g^b, g^c)$ where $a, b, c \in_R Z_q^*$, there is no probabilistic ploynomial-time algorithm $\mathcal{A}$ to compute $\hat{e}(g, g)^{abc}$ with nonnegligible probability.

## 3. System Model and Security Requirements

*3.1. System Model.* As shown in Figure 1, the PKE-AUT system under the dual server model consists of four types of entities, namely, trusted authority, primary server, secondary server, and users. The trusted authority is responsible for initializing the system, picking the security parameter, and producing public system parameters. Both data sender and data receiver are system users. Before being uploaded to the primary server, the data is encrypted using the public keys of the data receiver and two servers, so that only the data in the ciphertext format is outsourced. The data receiver is able to retrieve the data from the primary server for decryption with his private key and issue the same authorization to the primary and secondary servers, so that the two servers can jointly perform equality test on ciphertexts.

In the PKE-AUT system, the primary server and secondary server are assumed not to collude. All outsourced data are stored at the primary server in ciphertext format to protect their privacy. After being authorized, the primary server can perform the partial equality test procedure on outsourced ciphertexts, where the intermediate results would be produced and sent to the secondary server for processing. The second server further determines whether the ciphertexts encrypt the same data according to the intermediate results and gives the final equality test result to the data user. This equality test procedure with two phases can be executed in multiuser setting; that is, the primary and secondary servers can perform the equality test on ciphertexts of multiple users according to their authorization.

*3.2. Security Requirements.* In the PKE-AUT system under the dual server model, the primary server and the secondary server are independent and would not collude to attack the outsourced data. A secure PKE-AUT system has to satisfy the following requirements.

(i) *Data privacy against the primary server*: user data are stored at the primary server. Although the primary server is authorized to perform the equality test on ciphertexts, it cannot obtain the plaintexts from ciphertexts.

(ii) *Data privacy against the secondary server*: after obtaining the authorization for conducting equality test from users, the secondary server cannot deduce the plaintext information of outsourced data from the received intermediate results.

(iii) *Privacy protection on authentication*: the authentication generated by the data user can only be decrypted by the primary server and secondary server.

*3.3. System Framework.* A PKE-AUT scheme is composed of nine procedures, namely, the system setup, user key generation, server key generation, data encryption, data decryption, authentication generation, authentication recovery, primary server equality test, and secondary server equality test.

*System setup*: on input of the security parameter $1^\lambda$, which is carried out by the trusted authority, outputs the system public parameters Para. We denote Para $\longleftarrow$ `Setup`$(1^\lambda)$.

*User key generation*: on input of the system public parameters Para, the user key generation procedure, which is carried out by each user $U_i$, generates a pair of public key $pk_i$ and secret key $sk_i$. We denote $(pk_i, sk_i)$ $\longleftarrow$ `UKeyGen`(Para).

*Server key generation*: on input of the system public parameters Para, the server key generation procedure, which is carried out by each server $S_j$ including the primary server $S_1$ and secondary server $S_2$, generates a pair of public key $spk_j$ and secret key $ssk_j$. We denote $(spk_j, ssk_j) \longleftarrow$ `SKey Gen`(Para).

*Data encryption*: on input of the public keys $pk_i, spk_1, s$ $pk_2$of data receiver $U_i$, primary server $S_1$ and secondary server $S_2$, and a message $m$, the data encryption procedure, which is run by the data sender, generates a ciphertext $C$ and outsources it to the primary server $S_1$. We denote $C \longleftarrow$ `Encrypt`$(pk_i, spk_1, spk_2, m)$.

*Data decryption*: on input of the secret key $sk_i$ of user $U_i$, the public keys $spk_1, spk_2$ of primary server $S_1$ and secondary server $S_2$, and a ciphertext $C$, the data decryption procedure, which is run by the data receiver, outputs a plaintext $m$ or $\perp$ that signifies an error in decryption. We denote $m/\perp \longleftarrow$ `Decrypt`$(sk_i, spk_1, spk_2, C)$.

*Authentication generation*: on input of the secret key $sk_i$ of user $U_i$ and the public keys $spk_1, spk_2$ of primary server $S_1$

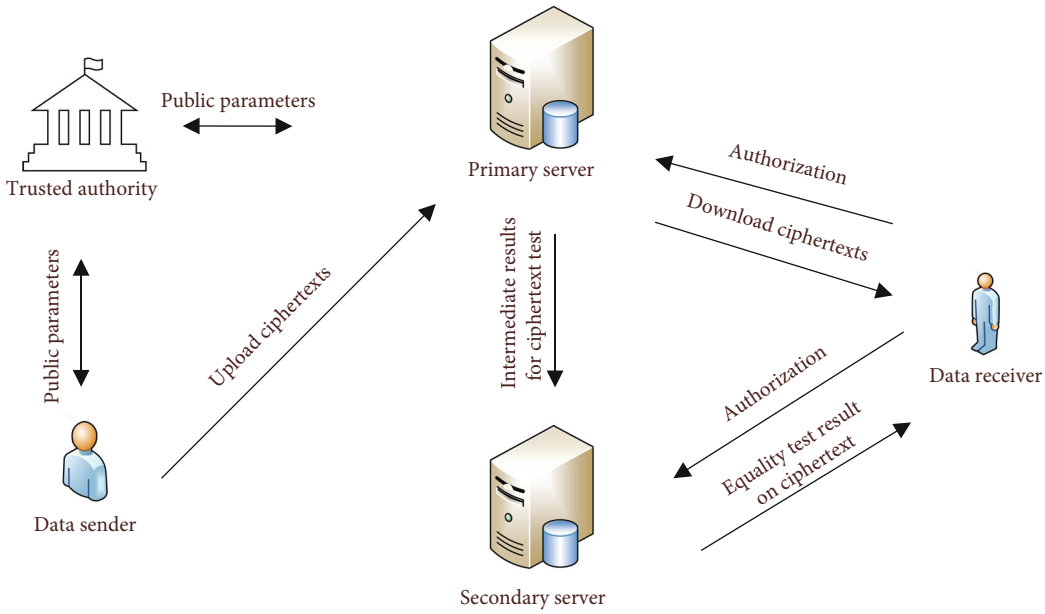


Figure 1: System model of PKE-AUT.

and secondary server $S_2$, the authentication generation procedure, which is run by user $U_i$, generates a ciphertext authentication $Z_i$ for two servers. Note that two servers have the same ciphertext authentication $Z_i$. We denote $Z_i \longleftarrow \mathtt{AuthGen}(sk_i, spk_1, spk_2)$.

*Authentication recovery*: on input of a ciphertext authentication $Z_i$, the secret key $ssk_1$ of primary server $S_1$ (resp., $ssk_2$ of secondary server $S_2$), and the public key $spk_2$ of secondary server $S_2$ (resp., $spk_1$ of primary server $S_1$), the authentication recovery procedure, which is run by the primary server $S_1$ (resp., secondary server $S_2$), outputs a plaintext authentication $r_i$ or $\perp$ that signifies an error in recovery. We denote $r_i/\perp \longleftarrow \mathtt{AuthRec}(Z_i, ssk_1, spk_2)$ or $r_i/\perp \longleftarrow \mathtt{AuthRec}(Z_i, ssk_2, spk_1)$.

*Primary server equality test*: on input of the authentications $r_i$ and $r_\ell$ of two users $U_i$ and $U_\ell$, respectively, their public keys $pk_i$ and $pk_\ell$, their ciphertexts $C$ and $C'$, and the secret key $ssk_1$ of the primary server $S_1$, the first equality test procedure, which is run by the primary server $S_1$, outputs an intermediate result $\Theta$ and gives it to the secondary server $S_2$. We denote $\Theta \longleftarrow \mathtt{TestS}_1(r_i, r_\ell, pk_i, pk_\ell, C, C', ssk_1)$.

*Secondary server equality test*: on input of the authentications $r_i$ and $r_\ell$ of two users $U_i$ and $U_\ell$, respectively, their public keys $pk_i$ and $pk_\ell$, an intermediate result $\Theta$, and the secret key $ssk_2$ of the secondary server $S_2$, the second equality test procedure, which is run by the secondary server $S_2$, outputs 1 if $C$ and $C'$ encrypt the same message or 0 otherwise. We denote $1/0 \longleftarrow \mathtt{TestS}_2(r_i, r_\ell, pk_i, pk_\ell, \Theta, ssk_2)$.

A PKE-AUT scheme must be *sound* in the sense that (1) each ciphertext produced by the data encryption procedure is decryptable by the data decryption procedure; (2) the ciphertext authentication produced by the authentication generation procedure can be recovered by the authentication recovery procedure; (3) for any two ciphertexts that encrypt the same message, which may be generated by different users, the two equality test procedures must finally output 1; and (4) for any two ciphertexts that encrypt different messages, which may be generated by different users, the two equality test procedures must finally output 0 with overwhelming probability.

*Definition 1* (soundness). A PKE-AUT scheme is sound if, for any security parameter $\lambda$, any public parameters Para $\longleftarrow \mathtt{Setup}(1^\lambda)$, any public/secret key pairs of two users $(pk_i, sk_i) \longleftarrow \mathtt{UKeyGen}(\text{Para})$ and $(pk_\ell, sk_\ell) \longleftarrow \mathtt{UKeyGen}(\text{Para})$, and any public/secret key pairs of two servers $(spk_1, ssk_1) \longleftarrow \mathtt{SKeyGen}(\text{Para})$ and $(spk_2, ssk_2) \longleftarrow \mathtt{SKeyGen}(\text{Para})$, the following conditions hold:

(i) For any message $m$, $\mathtt{Decrypt}(sk_i, spk_1, spk_2, \mathtt{Encrypt}(pk_i, spk_1, spk_2, m)) = m$.

(ii) $\mathtt{AuthRec}(\mathtt{AuthGen}(sk_i, spk_1, spk_2), ssk_1, spk_2) = r_i$ and $\mathtt{AuthRec}(\mathtt{AuthGen}(sk_i, spk_1, spk_2), ssk_2, spk_1) = r_i$.

(iii) For any two messages $m, m'$ such that $C \longleftarrow \mathtt{Encrypt}(pk_i, spk_1, spk_2, m)$ and $C' \longleftarrow \mathtt{Encrypt}(pk_\ell, spk_1, spk_2, m')$, if $m = m'$, then $\mathtt{TestS}_2(r_i, r_\ell, pk_i, pk_\ell, \Theta, ssk_2) = 1$; otherwise, $\Pr[\mathtt{TestS}_2(r_i, r_\ell, pk_i, pk_\ell, \Theta, ssk_2) = 1] \le \epsilon(\cdot)$, where $r_i = \mathtt{AuthRec}(\mathtt{AuthGen}(sk_i, spk_1, spk_2), ssk_1, spk_2) = \mathtt{AuthRec}(\mathtt{AuthGen}(sk_i, spk_1, spk_2), ssk_2, spk_1)$, $r_\ell = \mathtt{AuthRec}(\mathtt{AuthGen}(sk_\ell, spk_1, spk_2), ssk_1, spk_2) = \mathtt{AuthRec}(\mathtt{AuthGen}(sk_\ell, spk_1, spk_2), ssk_2, spk_1)$, and $\Theta \longleftarrow \mathtt{TestS}_1(r_i, r_\ell, pk_i, pk_\ell, C, C', ssk_1)$, and $\epsilon(\cdot)$ denotes a negligible function.

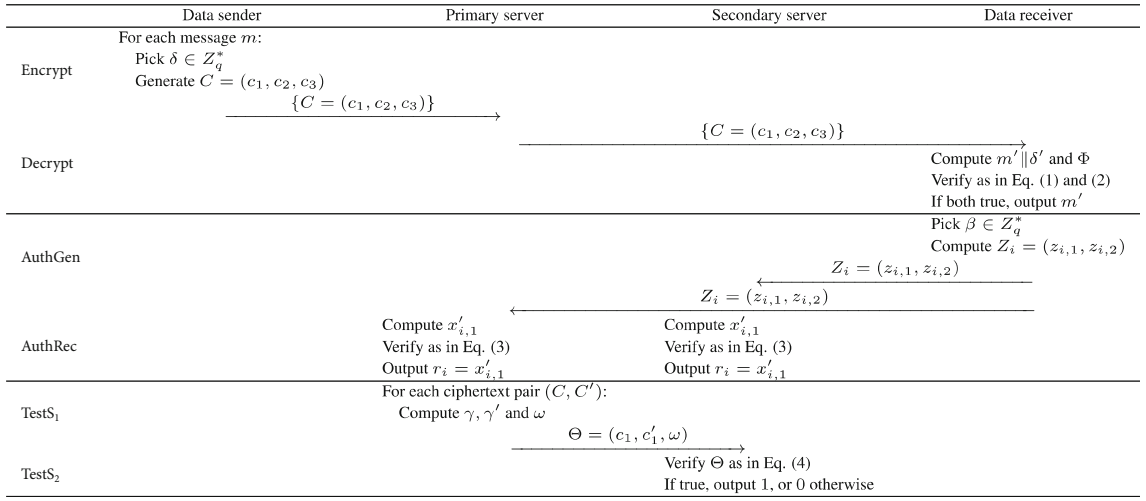


Figure 2: A procedure of the proposed PKE-AUT scheme.

## 4. PKE-AUT Construction

*4.1. Concrete Construction.* This section presents our PKE-AUT construction on bilinear groups in the dual server model, where a running procedure is shown in Figure 2. The frequently used symbols are summarized in Table 1.

*4.1.1. System Setup.* With security parameter $1^\lambda$, the trusted authority picks two cyclic groups $G = \langle g \rangle$ and $G_T$ of prime order $q$, which satisfy bilinear mapping $\hat{e} : G \times G \longrightarrow G_T$. It also chooses four cryptographic hash functions $H_1 : G \times G_T \longrightarrow G$, $H_2 : G \times G \longrightarrow \{0,1\}^{\tau_G + \log q}$, $H_3 : G_T \longrightarrow \{0,1\}^{\log q}$, and $H_4 : \{0,1\}^{\tau_m} \longrightarrow G$, where $\tau_G$ denotes the element size in group $G$ and $\tau_m$ represents the size of messages. The system public parameters are $\text{Para} = (\lambda, G, G_T, q, \hat{e}, g, H_1, H_2, H_3, H_4)$.

*4.1.2. User Key Generation.* Each user $\text{U}_i$ randomly picks three elements $x_{i,1}, x_{i,2}, x_{i,3} \in Z_q^*$ and computes

$$\chi_{i,1} = g^{x_{i,1}}, \chi_{i,2} = g^{x_{i,2}}, \chi_{i,3} = g^{x_{i,3}}. \tag{3}$$

Thus, the public key and secret key of user $\text{U}_i$ are $pk_i = (\chi_{i,1}, \chi_{i,2}, \chi_{i,3})$ and $sk_i = (x_{i,1}, x_{i,2}, x_{i,3})$, respectively.

*4.1.3. Server Key Generation.* The primary server $\text{S}_1$ randomly selects two elements $y_{1,1}, y_{1,2} \in Z_q^*$ and computes

$$\rho_{1,1} = g^{y_{1,1}}, \rho_{1,2} = g^{y_{1,2}}. \tag{4}$$

Thus, the public key and secret key of primary server $\text{S}_1$ are $spk_1 = (\rho_{1,1}, \rho_{1,2})$ and $ssk_1 = (y_{1,1}, y_{1,2})$, respectively. In a similar way, the secondary server $\text{S}_2$ is able to generate its public key $spk_2 = (\rho_{2,1}, \rho_{2,2})$ and secret key $ssk_2 = (y_{2,1}, y_{2,2})$.

Table 1: Notations.

| Symbol | Meaning |
|---|---|
| $\lambda$ | Security parameter |
| $G, G_T$ | Cyclic groups of prime order $q$ satisfying bilinear pairing $\hat{e} : G \times G \longrightarrow G_T$ |
| $H_1, H_2, H_3, H_4$ | Cryptographic hash functions |
| $g$ | A generator of $G$ |
| $sk_i = (x_{i,1}, x_{i,2}, x_{i,3})$ | Private key of user $\text{U}_i$ |
| $pk_i = (\chi_{i,1}, \chi_{i,2}, \chi_{i,3})$ | Public key of user $\text{U}_i$ |
| $spk_1 = (\rho_{1,1}, \rho_{1,2})$ | Public key of primary server $\text{S}_1$ |
| $ssk_1 = (y_{1,1}, y_{1,2})$ | Secret key of primary server $\text{S}_1$ |
| $spk_2 = (\rho_{2,1}, \rho_{2,2})$ | Public key of secondary server $\text{S}_2$ |
| $ssk_2 = (y_{2,1}, y_{2,2})$ | Secret key of secondary server $\text{S}_2$ |
| $\delta, \beta$ | Random elements in $Z_q^*$ |
| $C = (c_1, c_2, c_3)$ | Ciphertext of message $m$ |
| $Z_i = (z_{i,1}, z_{i,2})$ | Authentication of user $\text{U}_i$ in ciphertext format |
| $r_i, r_\ell$ | Authentications of users $\text{U}_i$ and $\text{U}_\ell$ |
| $\Theta = (c_1, c'_1, \omega)$ | Intermediate result of equality test |
| $\gamma, \gamma'$ | Temporary elements for computing $\omega$ |

*4.1.4. Data Encryption.* For a message $m \in \{0,1\}^{\tau_m}$, the data sender randomly picks $\delta \in Z_q^*$ and computes the ciphertext $C = (c_1, c_2, c_3)$ as follows:

$$\begin{aligned} c_1 &= g^\delta, \\ c_2 &= H_4(m) \cdot H_1\left(\chi_{i,1}^\delta \middle\| \hat{e}(\chi_{i,2}, \rho_{1,1})^\delta\right) \cdot H_1\left(\chi_{i,1}^\delta \middle\| \hat{e}(\chi_{i,2}, \rho_{2,1})^\delta\right), \\ c_3 &= (m \| \delta) \oplus H_2\left(\chi_{i,3}^\delta \| H_4(m)\right), \end{aligned} \tag{5}$$

where $\|$ denotes the concatenation of strings and $\oplus$ represents the XOR operation. Then, the ciphertext $C = (c_1, c_2, c_3)$ is sent to the primary server $S_1$.

*4.1.5. Data Decryption.* Given a ciphertext $C = (c_1, c_2, c_3)$, the data receiver computes

$$m' \| \delta' = c_3 \oplus H_2\left(c_1^{x_{i,3}} \| \Phi\right), \tag{6}$$

where

$$\Phi = \frac{c_2}{H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(c_1, \rho_{1,1}\right)^{x_{i,2}}\right) \cdot H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(c_1, \rho_{2,1}\right)^{x_{i,2}}\right)}, \tag{7}$$

then verifies

$$c_1 \stackrel{?}{=} g^{\delta'}, \tag{8}$$

$$\Phi \stackrel{?}{=} H_4\left(m'\right). \tag{9}$$

If both equalities hold, then the data receiver outputs $m'$, otherwise $\perp$.

*4.1.6. Authentication Generation.* Data user $U_i$ randomly picks an element $\beta \in Z_q^*$ and computes the ciphertext authentication $Z_i = (z_{i,1}, z_{i,2})$ as follows:

$$\begin{aligned} z_{i,1} &= g^{\beta}, \\ z_{i,2} &= x_{i,1} \oplus H_3\left(\widehat{e}\left(\rho_{1,2}, \rho_{2,2}\right)^{\beta}\right). \end{aligned} \tag{10}$$

Data user $U_i$ sends the ciphertext authentication $Z_i = (z_{i,1}, z_{i,2})$ to two servers $S_1$ and $S_2$.

*4.1.7. Authentication Recovery.* The primary server $S_1$ computes

$$x'_{i,1} = z_{i,2} \oplus H_3\left(\widehat{e}\left(z_{i,1}, \rho_{2,2}\right)^{y_{1,2}}\right), \tag{11}$$

and verifies

$$\chi_{i,1} \stackrel{?}{=} g^{x'_{i,1}}. \tag{12}$$

If the equality in (12) is satisfied, then the primary server $S_1$ outputs plaintext authentication $r_i = x'_{i,1}$, otherwise outputs symbol $\perp$. The secondary server can run the recovery procedure to obtain the same plaintext authentication $r_i = x'_{i,1}$ in the similar way.

*4.1.8. Primary Server Equality Test.* For ciphertext $C = (c_1, c_2, c_3)$ of user $U_i$ and ciphertext $C' = (c'_1, c'_2, c'_3)$ of user $U_\ell$, the primary server $S_1$ generates the intermediate result $\Theta = (c_1, c'_1, \omega)$ according to their authentications $r_i$ and $r_\ell$ as follows. The primary server $S_1$ computes

$$\begin{aligned} \gamma &= \frac{c_2}{H_1\left(c_1^{r_i} \| \widehat{e}\left(\chi_{i,2}, c_1\right)^{y_{1,1}}\right)}, \\ \gamma' &= \frac{c'_2}{H_1\left(c_1'^{r_\ell} \| \widehat{e}\left(\chi_{\ell,2}, c'_1\right)^{y_{1,1}}\right)}. \end{aligned} \tag{13}$$

It continues to compute

$$\omega = \frac{\gamma}{\gamma'}. \tag{14}$$

The intermediate result $\Theta = (c_1, c'_1, \omega)$ is sent to the secondary server $S_2$.

*4.1.9. Secondary Server Equality Test.* For the received intermediate result $\Theta = (c_1, c'_1, \omega)$, the secondary server $S_2$ verifies

$$\omega \stackrel{?}{=} \frac{H_1\left(c_1^{r_i} \| \widehat{e}\left(\chi_{i,2}, c_1\right)^{y_{2,1}}\right)}{H_1\left(c_1'^{r_\ell} \| \widehat{e}\left(\chi_{\ell,2}, c'_1\right)^{y_{2,1}}\right)}. \tag{15}$$

If the equality in (15) is satisfied, then the secondary server $S_2$ outputs 1; otherwise, it outputs 0.

*4.2. Soundness*

**Theorem 1.** *The proposed PKE-AUT scheme in the dual server model is sound.*

*Proof.*

(1) For data decryption, since

$$\begin{aligned} \Phi &= \frac{c_2}{H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(c_1, \rho_{1,1}\right)^{x_{i,2}}\right) \cdot H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(c_1, \rho_{2,1}\right)^{x_{i,2}}\right)} \\ &= \frac{H_4(m) \cdot H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{1,1}\right)^{\delta}\right) \cdot H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right)}{H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(g^{\delta}, \rho_{1,1}\right)^{x_{i,2}}\right) \cdot H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(g^{\delta}, \rho_{2,1}\right)^{x_{i,2}}\right)} \\ &= \frac{H_4(m) \cdot H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(\chi_{i,2}, \rho_{1,1}\right)^{\delta}\right) \cdot H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right)}{H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(\chi_{i,2}, \rho_{1,1}\right)^{\delta}\right) \cdot H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right)} \\ &= H_4(m), \end{aligned} \tag{16}$$

we have

$$\begin{aligned} m' \| \delta' &= c_3 \oplus H_2\left(c_1^{x_{i,3}} \| \Phi\right) \\ &= \left((m \| \delta) \oplus H_2\left(\chi_{i,3}^{\delta} \| H_4(m)\right)\right) \oplus H_2\left(g^{\delta x_{i,3}} \| H_4(m)\right) \\ &= (m \| \delta) \oplus H_2\left(\chi_{i,3}^{\delta} \| H_4(m)\right) \oplus H_2\left(\chi_{i,3}^{\delta} \| H_4(m)\right) \\ &= m \| \delta. \end{aligned} \tag{17}$$

Thus, the equalities in (8) and (9) hold.

(2) For authentication recovery, since

$$\begin{aligned} x'_{i,1} &= z_{i,2} \oplus H_3\left(\widehat{e}\left(z_{i,1}, \rho_{2,2}\right)^{y_{1,2}}\right) \\ &= \left(x_{i,1} \oplus H_3\left(\widehat{e}\left(\rho_{1,2}, \rho_{2,2}\right)^{\beta}\right)\right) \oplus H_3\left(\widehat{e}\left(g^{\beta}, \rho_{2,2}\right)^{y_{1,2}}\right) \\ &= x_{i,1} \oplus H_3\left(\widehat{e}\left(\rho_{1,2}, \rho_{2,2}\right)^{\beta}\right) \oplus H_3\left(\widehat{e}\left(\rho_{1,2}, \rho_{2,2}\right)^{\beta}\right) \\ &= x_{i,1}, \end{aligned} \tag{18}$$

the equality in (12) is satisfied.

(3) For equality test on ciphertexts, since

$$\begin{aligned} \gamma &= \frac{c_2}{H_1\left(c_1^{r_i} \| \widehat{e}\left(\chi_{i,2}, c_1\right)^{y_{1,1}}\right)} \\ &= \frac{H_4(m) \cdot H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{1,1}\right)^{\delta}\right) \cdot H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right)}{H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(\chi_{i,2}, g^{\delta}\right)^{y_{1,1}}\right)} \\ &= \frac{H_4(m) \cdot H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{1,1}\right)^{\delta}\right) \cdot H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right)}{H_1\left(c_1^{x_{i,1}} \| \widehat{e}\left(\chi_{i,2}, \rho_{1,1}\right)^{\delta}\right)} \\ &= H_4(m) \cdot H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right), \\ \gamma' &= \frac{c'_2}{H_1\left(c_1'^{r_\ell} \| \widehat{e}\left(\chi_{\ell,2}, c'_1\right)^{y_{1,1}}\right)} \\ &= \frac{H_4\left(m'\right) \cdot H_1\left(\chi_{\ell,1}^{\delta} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{1,1}\right)^{\delta'}\right) \cdot H_1\left(\chi_{\ell,1}^{\delta} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{2,1}\right)^{\delta'}\right)}{H_1\left(c_1'^{x'_{\ell,1}} \| \widehat{e}\left(\chi_{\ell,2}, g^{\delta'}\right)^{y_{1,1}}\right)} \\ &= \frac{H_4\left(m'\right) \cdot H_1\left(\chi_{\ell,1}^{\delta} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{1,1}\right)^{\delta'}\right) \cdot H_1\left(\chi_{\ell,1}^{\delta} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{2,1}\right)^{\delta'}\right)}{H_1\left(c_1'^{x_{\ell,1}} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{1,1}\right)^{\delta'}\right)} \\ &= H_4\left(m'\right) \cdot H_1\left(\chi_{\ell,1}^{\delta} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{2,1}\right)^{\delta'}\right), \end{aligned} \tag{19}$$

we have

$$\omega = \gamma / \gamma' = \frac{H_4(m) \cdot H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right)}{H_4\left(m'\right) \cdot H_1\left(\chi_{\ell,1}^{\delta'} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{2,1}\right)^{\delta'}\right)}. \tag{20}$$

Also, we know

$$\begin{aligned} \frac{H_1\left(c_1^{r_i} \| \widehat{e}\left(\chi_{i,2}, c_1\right)^{y_{2,1}}\right)}{H_1\left(c_1'^{r_\ell} \| \widehat{e}\left(\chi_{\ell,2}, c'_1\right)^{y_{2,1}}\right)} &= \frac{H_1\left(g^{\delta' x_{i,1}} \| \widehat{e}\left(\chi_{i,2}, g^{\delta}\right)^{y_{2,1}}\right)}{H_1\left(g^{\delta' x'_{\ell,1}} \| \widehat{e}\left(\chi_{\ell,2}, g^{\delta'}\right)^{y_{2,1}}\right)} \\ &= \frac{H_1\left(g^{\delta x_{i,1}} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right)}{H_1\left(g^{\delta' x_{\ell,1}} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{2,1}\right)^{\delta'}\right)} \\ &= \frac{H_1\left(\chi_{i,1}^{\delta} \| \widehat{e}\left(\chi_{i,2}, \rho_{2,1}\right)^{\delta}\right)}{H_1\left(\chi_{\ell,1}^{\delta'} \| \widehat{e}\left(\chi_{\ell,2}, \rho_{2,1}\right)^{\delta'}\right)}. \end{aligned} \tag{21}$$

It can be seen that if and only if $m = m'$, the equality in (15) is satisfied.

Therefore, the proposed PKE-AUT scheme in the dual server model is sound. □

## 5. Analysis and Comparison

### 5.1. Security Analysis

**Theorem 2.** *The proposed PKE-AUT scheme in the dual server model can protect the privacy of outsourced data against the primary server.*

*Proof.* The ciphertext in the proposed PKE-AUT scheme has the similar form in Lee et al.'s scheme [18]. The difference lies in that for generating the second element $c_2$ in ciphertext, all the public keys of the data receiver and two servers should be used in the proposed PKE-AUT scheme; in this way, these two servers after being authorized are allowed to jointly perform the equality test on ciphertexts with their private keys. The proof is similar to that of Theorem 4. 1 in [18], except for a small difference in the simulation on the decryption oracle; that is, the proposed PKE-AUT scheme offers the indistinguishability under adaptive chosen ciphertext attacks (IND-CCA) against the primary server assuming the CDH and CBDH assumptions hold. □

**Theorem 3.** *The proposed PKE-AUT scheme in the dual server model can protect the privacy of outsourced data against the secondary server.*

*Proof.* In the proposed PKE-AUT scheme, all outsourced ciphertexts are stored at the primary server. During the process of equality test on ciphertexts, only the intermediate result $\Theta = (c_1, \gamma, c'_1, \gamma')$ is delivered to the secondary server by the primary server. Note that the pairs $(c_1, \gamma)$ and $(c'_1, \gamma')$ have the similar form of Lee et al.'s scheme [18], where the difference lies in that their scheme also has another element for enabling decryption by the user. Thus, the proof is similar to that of Theorem 4.1 in [18]; that is, the proposed PKE-AUT scheme is IND-CCA secure against the secondary server under the CDH and CBDH assumptions. □

**Theorem 4.** *The proposed PKE-AUT scheme in the dual server model can protect the privacy of authentication.*

*Proof.* The ciphertext authentication generated by the proposed PKE-AUT scheme has the similar format as the ciphertexts in Boneh and Franklin's identity-based encryption scheme (Section 4 of [28]). The difference is that in the input to the hash function $H_3$, the public keys of two servers are both used in evaluating $\widehat{e}(\cdot, \cdot)$, whereas the user identity and public parameters are used in Boneh and Franklin's scheme [28]. Thus, the proof is similar to that of Theorem 4.1 in [28]; that is, the authentication in the proposed PKE-AUT scheme enjoys the indistinguishability under chosen plaintext attacks (IND-CPA) assuming the CBDH assumption holds. □

*5.2. Performance Analysis.* This section analyzes the performance of the proposed PKE-AUT scheme and compares with existing schemes, where only resource-intensive operations such as exponentiation, bilinear pairing, and map-to-point hash function are considered. The comparison with Wu et al.'s scheme [15] is shown in Table 2, where `Pair`, `Expo`, `Hash` denote the evaluation costs of a bilinear pairing $\widehat{e}(\cdot, \cdot)$, an exponentiation in group $G$, and a map-to-point hash function, respectively.

It can be seen from Table 2 that, for producing a pair of public and secret keys for each user, our `UKeyGen` procedure requires 3 exponentiations in group $G$. Although our `UKeyGen` procedure has one more exponentiation than Wu et al.'s scheme [15], it does not take any map-to-point hash evaluation. The `SKeyGen` procedure in our PKE-AUT scheme is executed by the primary server and secondary server, respectively, for generating their public and secret keys. Thus, their key pairs have the same form, where each takes 2 exponentiations in group $G$. While in Wu et al.'s scheme [15], the two servers run different key generation procedures, which implies their key pairs are in different form and take two and one exponentiation in group $G$, respectively.

In the data encryption phase, the exponentiations in group $G_T$ in our PKE-AUT scheme and Wu et al.'s scheme [15] can be transformed into exponentiations in group $G$; in this way, the corresponding parameters can be used in multiple steps and the efficiency can be improved. In this case, the `Encrypt` of our PKE-AUT scheme takes one less bilinear pairing operation than that in Wu et al.'s scheme [15] for encrypting a message. Note that our PKE-AUT scheme is able to concurrently authorize the primary server and secondary server to perform the equality test on ciphertexts, which makes the ciphertext contain more elements than that of Wu et al.'s scheme [15]. Thus, for data decryption, our PKE-AUT scheme should take more computations than Wu et al.'s scheme [15].

In our PKE-AUT scheme, the data user is able to generate the ciphertext authentication for two servers; that is, the same ciphertext authentication can be recovered by both the primary server and the secondary server with their respective secret keys. Thus, the computing costs for authentication generation can be reduced compared to issuing an authentication for each server separately. Since the exponentiation in group $G_T$ can be converted to the one in group $G$, both `AuthGen` and `AuthRec` procedures have the same computing costs, that is, two exponentiations in group $G$ and one map-to-point hash evaluation. In Wu et al.'s scheme [15], the privacy of authentication is not considered.

With authentication, the primary server and secondary server can cooperatively perform the equality test on ciphertexts. In our PKE-AUT scheme, both equality test procedures for two servers should take 4 more exponentiations in group $G$ than Wu et al.'s scheme [15], since the generation of the second element $c_2$ in the ciphertext of our PKE-AUT scheme requires more input parameters for achieving the equality test on the ciphertext by two servers. It can be seen that the two servers in both schemes do not have the same computing costs, since the secondary server needs to run two bilinear pairings in generating the result of the equality test on a pair of ciphertexts.

Table 2: Comparison of computing costs.

| Procedure | Our PKE-AUT scheme | Wu et al.'s scheme [15] |
|---|---|---|
| `UKeyGen` | 3Expo | 2Expo + 1Hash |
| `SKeyGen` | 2Expo | 2Expo/1Expo |
| `Encrypt` | 4Expo + 2Pair + 3Hash | 5Expo + 3Pair + 2Hash |
| `Decrypt` | 4Expo + 2Pair + 3Hash | 2Expo + 1Pair |
| `AuthGen` | 2Expo + 1Pair | — |
| `AuthRec` | 2Expo + 1Pair | — |
| $\texttt{TestS}_1$ | 4Expo + 2Pair + 2Hash | 2Pair + 2Hash |
| $\texttt{TestS}_2$ | 4Expo + 2Pair + 2Hash | 4Pair + 2Hash |

Table 3: Comparison of communication costs.

| | Our PKE-AUT scheme | Wu et al.'s scheme [15] |
|---|---|---|
| Ciphertext | $2\tau_G + \tau_m + \log q$ | $5\tau_G + \log q$ |
| Authentication | $\tau_G + \log q$ | $\tau_G$ |
| Intermediate result | $3\tau_G$ | $6\tau_G$ |

The communication costs of our PKE-AUT scheme and Wu et al.'s scheme [15] are compared in Table 3. In our scheme, each ciphertext has three elements, while the ciphertext in Wu et al.'s scheme [15] contains five elements. Note that the message space of Wu et al.'s scheme [15] is cyclic group $G$. Thus, when both schemes have the same message space $G$, the ciphertext size of their scheme would be $2\tau_G$ more than our PKE-AUT scheme. The authentication token was not encrypted for protecting privacy in Wu et al.'s scheme [15], which only contains one element in group $G$. For the equality test procedure by the primary server, the generated intermediate result $\Theta = (c_1, c_1', \omega)$ in our PKE-AUT scheme has three elements in group $G$, while Wu et al.'s scheme [15] requires six elements in $G$.

Moreover, we analyze the performance of our PKE-AUT scheme and compare with Wu et al.'s scheme [15] in the dual server model according to the experimental results of cryptographic operations in [29, 30]. In [29], the experiments were conducted on a platform with Windows 7 operating system, Intel I7-4700@3.40 GHz CPU and 4 GB memory. Moreover, the MIRACL Cryptographic SDK [31] was invoked with $\log p = 512$. The execution time of some cryptographic operations are summarized in Table 4.

The performance of all procedures of our PKE-AUT scheme and Wu et al.'s scheme [15] is depicted in Figures 3 and 4, respectively. The case where each procedure is executed once is considered for both schemes. It can be seen that the proposed PKE-AUT scheme is more efficient than Wu et al.'s scheme [15] in encrypting a message. Although the decryption and equality test procedures take more time than Wu et al.'s scheme [15], our PKE-AUT scheme supports strict and symmetric authorization for

Table 4: Execution time of cryptographic operations.

| Operation | Execution time (milliseconds) |
|---|---|
| Pair | 4.211 |
| Expo | 1.709 |
| Hash | 4.406 |

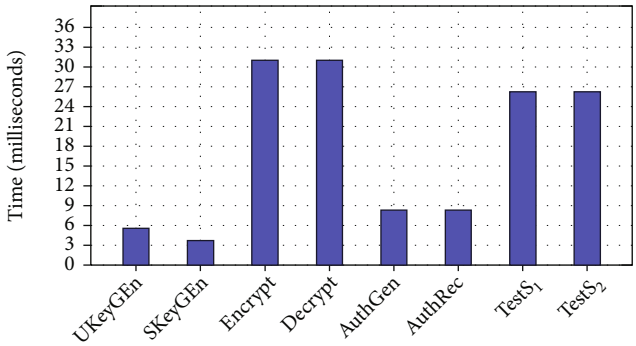


Figure 3: Performance of each procedure in our PKE-AUT scheme.

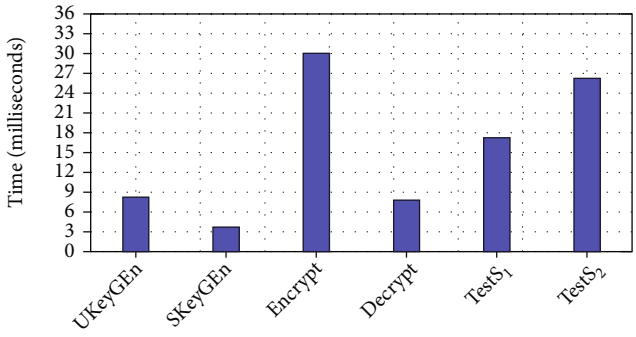


Figure 4: Performance of each procedure in Wu et al.'s scheme [15].

equality test on ciphertexts. Thus, to achieve this, the public keys of two servers have to be used in generating the ciphertext in our PKE-AUT scheme, which makes the efficiency of decryption and equality test reduced slightly.

## 6. Conclusion

To address the issues of privacy protection and resistance of keyword guessing attacks on outsourced ciphertexts in clouds, this paper presented a public key encryption scheme supporting the authorized equality test on ciphertexts in the dual server mode (PKE-AUT). User data can be only stored at the primary server to save local storage costs. With the same authentication, the primary server and secondary server can jointly carry out the equality test on ciphertexts of the corresponding users. The mechanism of the equality test on ciphertexts can be run in a multiuser setting, such that after being authorized, the two servers can compare the ciphertexts of these multiple users. Security analysis showed that the proposed PKE-AUT scheme guarantees the privacy of outsourced ciphertexts against two servers, as well as the privacy of authentication. Performance analysis and comparison demonstrated the practicality of the proposed PKE-AUT scheme.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This article is supported in part by the National Natural Science Foundation of China under projects 61862012 and 61962012; the Guangxi Natural Science Foundation under grants 2019GXNSFFA245015 and 2019GXNSFGA245004; and the PCNL Major Key Project under grants PCL2021A09-4 and PCL2021A02-3.

## References

[1] H. Deng, Z. Qin, L. Sha, and H. Yin, "A flexible privacy-preserving data sharing scheme in cloud-assisted IoT," *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11601–11611, 2020.

[2] W.-B. Kim, D. Seo, D. Kim, and I.-Y. Lee, "Group delegated ID-based proxy reencryption for the enterprise IoT-cloud storage environment," *Wireless Communications and Mobile Computing*, vol. 2021, 12 pages, 2021.

[3] X. Liu, R. H. Deng, K.-K. R. Choo, and Y. Yang, "Privacy-preserving outsourced clinical decision support system in the cloud," *IEEE Transactions on Services Computing*, vol. 14, no. 1, pp. 222–234, 2021.

[4] R. Chen, Y. Mu, G. Yang, F. Guo, and X. Wang, "Server-aided public key encryption with keyword search," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2833–2842, 2016.

[5] Y. Wang, H. H. Pang, N. H. Tran, and R. H. Deng, "CCA secure encryption supporting authorized equality test on ciphertexts in standard model and its applications," *Information Sciences*, vol. 414, pp. 289–305, 2017.

[6] G. Yang, C. H. Tan, Q. Huang, and D. S. Wong, "Probabilistic public key encryption with equality test," in *Proceedings of the 2010 international conference on topics in cryptology, CT-RSA'10*, pp. 119–131, Berlin, Heidelberg, 2010.

[7] S. Ma, M. Zhang, Q. Huang, and B. Yang, "Public key encryption with delegated equality test in a multi-user setting," *The Computer Journal*, vol. 58, no. 4, pp. 986–1002, 2014.

[8] Y. Wang, Q. Huang, H. Li, J. Huang, G. Yang, and W. Susilo, "Public key authenticated encryption with designated equality test and its applications in diagnostic related groups," *IEEE Access*, vol. 7, pp. 135999–136011, 2019.

[9] T. Wu, S. Ma, Y. Mu, and S. Zeng, "ID-based encryption with equality test against insider attack," in *Australasian conference on information security and privacy*, pp. 168–183, Cham, 2017.

[10] G. L. D. Nguyen, W. Susilo, D. H. Duong, H. Q. Le, and F. Guo, "Lattice-based IBE with equality test supporting flexible authorization in the standard model," in *Progress in Cryptology – INDOCRYPT 2020*, K. Bhargavan, E. Oswald, and M.

Prabhakaran, Eds., pp. 624–643, Springer International Publishing, Cham, 2020.

[11] Y. Ling, S. Ma, Q. Huang, X. Li, Y. Zhong, and Y. Ling, "Efficient group ID-based encryption with equality test against insider attack," *The Computer Journal*, vol. 64, no. 4, pp. 661–674, 2021.

[12] Y. Xu, M. Wang, H. Zhong, and S. Zhong, "IBEET-AOK: ID-based encryption with equality test against off-line KGAs for cloud medical services," *Frontiers of Computer Science*, vol. 15, no. 6, article 156814, 2021.

[13] Y. Wang, Y. Ding, Q. Wu, Y. Wei, B. Qin, and H. Wang, "Privacy-preserving cloud-based road condition monitoring with source authentication in VANETs," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 7, pp. 1779–1790, 2019.

[14] H. Xiong, Y. Hou, X. Huang, and Y. Zhao, "Secure message classification services through identity-based signcryption with equality test towards the internet of vehicles," *Vehicular Communications*, vol. 26, article 100264, 2020.

[15] L. Wu, Y. Zhang, and D. He, "Dual server identity-based encryption with equality test for cloud computing," *Journal of Computer Research and Development*, vol. 54, no. 10, pp. 2232–2243, 2017.

[16] Q. Tang, "Towards public key encryption scheme supporting equality test with fine-grained authorization," in *Proceedings of the 16th Australasian Conference on Information Security and Privacy, ACISP'11*, pp. 389–406, Berlin, Heidelberg, 2011.

[17] Q. Tang, "Public key encryption supporting plaintext equality test and user-specified authorization," *Security and Communication Networks*, vol. 5, no. 12, pp. 1351–1362, 2012.

[18] H. T. Lee, S. Ling, J. H. Seo, and H. Wang, "CCA2 attack and modification of Huang et al.'s public key encryption with authorized equality test," *The Computer Journal*, vol. 59, no. 11, pp. 1689–1694, 2016.

[19] K. Huang, R. Tso, Y.-C. Chen, S. M. M. Rahman, A. Almogren, and A. Alamri, "PKE-AET: public key encryption with authorized equality test," *The Computer Journal*, vol. 58, no. 10, pp. 2686–2697, 2015.

[20] S. Ma, "Identity-based encryption with outsourced equality test in cloud computing," *Information Sciences*, vol. 328, pp. 389–402, 2016.

[21] H. T. Lee, S. Ling, J. H. Seo, and H. Wang, "Semi-generic construction of public key encryption and identity-based encryption with equality test," *Information Sciences*, vol. 373, pp. 419–440, 2016.

[22] H. Pang and X. Ding, "Privacy-preserving ad-hoc equi-join on outsourced data," *ACM Transactions on Database Systems (TODS)*, vol. 39, no. 23, pp. 1–40, 2014.

[23] Y. Wang and H. H. Pang, "Probabilistic public key encryption for controlled equijoin in relational databases," *The Computer Journal*, vol. 60, no. 4, pp. 600–612, 2016.

[24] H. Cui, R. H. Deng, Y. Li, and G. Wu, "Attribute-based storage supporting secure deduplication of encrypted data in cloud," *IEEE Transactions on Big Data*, vol. 5, no. 3, pp. 330–342, 2019.

[25] Z. Yan, M. Wang, Y. Li, and A. V. Vasilakos, "Encrypted data management with deduplication in cloud computing," *IEEE Cloud Computing*, vol. 3, no. 2, pp. 28–35, 2016.

[26] H. Q. Le, D. H. Duong, P. S. Roy, W. Susilo, K. Fukushima, and S. Kiyomoto, "Lattice-based signcryption with equality test in standard model," *Computer Standards & Interfaces*, vol. 76, article 103515, 2021.

[27] W. Susilo, D. H. Duong, and H. Q. Le, "Efficient post-quantum identity-based encryption with equality test," in *2020 IEEE 26th International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 633–640, Hong Kong, 2020.

[28] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 586–615, 2003.

[29] D. He, S. Zeadally, B. Xu, and X. Huang, "An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2681–2691, 2015.

[30] W. Xiong, R. Wang, Y. Wang, F. Zhou, and X. Luo, "CPPA-D: efficient conditional privacy-preserving authentication scheme with double-insurance in VANETs," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3456–3468, 2021.

[31] *MIRACL Cryptographic SDK: multiprecision integer and rational arithmetic cryptographic library*, https://github.com/miracl/miracl.