RESEARCH ARTICLE

New constructions of equality test scheme for cloud-assisted wireless sensor networks

Huijun Zhu[1,2,3]*, Dong Xie[4], Haseeb Ahmad[5], Hasan Naji Hasan Abdullah[1]

1 Nanyang Institute of Technology, Nanyang, Henan, China, **2** State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications (BUPT), Beijing, China, **3** Graphic Image and Intelligent Processing in Henan Province, International Joint Laboratory, Nanyang Institute of Technology, Nanyang, Henan, China, **4** Anhui Normal University, Wuhu, Anhui, China, **5** Department of Computer Science, National Textile University, Faisalabad, Pakistan

* zhuhj1201@163.com

Citation: Zhu H, Xie D, Ahmad H, Hasan Abdullah HN (2021) New constructions of equality test scheme for cloud-assisted wireless sensor networks. PLoS ONE 16(10): e0258746. https://doi.org/10.1371/journal.pone.0258746

Editor: Pandi Vijayakumar, University College of Engineering Tindivanam, INDIA

Received: June 10, 2021

Accepted: October 4, 2021

Published: October 25, 2021

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pone.0258746

Data Availability Statement: All relevant data are within the manuscript.

Funding: This work was supported by the Science and Technology Department of Henan Province (no.212102310297), the Open Foundation of State

Abstract

Public key encryption with equality test enables the user to determine whether two ciphertexts contain the same information without decryption. Therefore, it may serve as promising cryptographic technique for cloud-assisted wireless sensor networks (CWSNs) to maintain data privacy. In this paper, an efficient RSA with equality test algorithm is proposed. The presented scheme also handles the attackers based on their authorization ability. Precisely, the proposed scheme is proved to be one-way against chosen-ciphertext attack security and indistinguishable against chosen ciphertext attacks. Moreover, the experimental evaluations depict that the underlying scheme is efficient in terms of encryption, decryption, and equality testing. Thus, this scheme may be used as a practical solution in context of CWSNs, where the users may compare two ciphertexts without decryption.

1 Introduction

Recently, Internet of Things (IoT) as a new information network technology is booming. In order to achieve a Smart and Secure environment, Stergiou et al. proposed a scenario that try to combine the functions of the IoT with cloud computing and edge computing and big data [1]. With the development of IoT and the technology of cloud computing, cloud-assisted wireless sensor networks (CWSNs) are widely applied in many fields, such as agriculture, military, transportation, medical and other similar fields. Although, CWSNs have extensive applications, but there also exist challenges to be addressed such as reduction of energy consumption. Recently in 2020, Guermazi et al. proposed a method to reduce energy consumption as well as to extend the lifetime of wireless sensor network [2]. For the evaluation models, Cao et al. proposed five intelligent evaluation models and implemented their experiments on the Nearest Closer Protocol with the J-Sim simulation tool [3]. Al-Qerem et al. proposed the mechanism to reduce the communication delay significantly [4]. The proposed mechanism shall enable low-latency fog computing services of the IoT applications that are a delay sensitive. The security of data is another imperative issue. Practically, extensive amount of data is being transmitted and stored on distributed servers, where it may face several threats. Therefore, to protect

Key Laboratory of Networking and Switching Technology (Beijing University of Posts and Telecommunications) (SKLNST-2019-2-17) and the National Natural Science Foundation of China (no. 61801004).

Competing interests: The authors have declared that no competing interests exist.

the confidentiality of such data is particularly important [5]. At present, various cryptographic algorithms are introduced for CWSNs environment. However, private key is necessary to obtain information from the encrypted data, that reduces the availability of data. In order to enhance the availability and to realize the convenient access over the encrypted data, searchable encryption technology (SE) for ciphertext data retrieval has got the festivity. SE is divided into symmetric search encryption [6–8] and public key search encryption [9]. SE algorithms realize the search operation over encrypted data (without disclosing the user's private key). Subsequently, several searchable encryption algorithms have been proposed [10–14].

In 2016, Chen et al. proposed Dual-Server Public Key Encryption with Keyword Search (DS-PEKS), that utilized the smooth projective hash function to design the scheme [15]. In the same year, Chen et al. proposed a server-aided public key encryption encryption scheme with the keyword search that obtained the security against the offline keyword guessing attack [16]. Getting motivation from the Chen's scheme, Huang et al. developed a new searchable encryption scheme with dual server model, that offers the indistinguishability of keyword retrieval trapdoor and can resist the internal keyword guessing attack (KGA) attack [17]. To satisfy users requirements in a more secure way, a fuzzy keyword search scheme is also proposed, which improves the usability of the system by matching the keywords based on similar semantic or with the nearest possible file [18]. To overcome the key escrow problem in identity-based cryptography (IBC) and the cumbersome certificate problem in conventional public key encryption (PKE), Lu et al. presented a new certificateless encryption with keyword search (CLEKS) framework [19].

Since the proposed the searchable encryption schemes can only obtain the ciphertexts of related keywords, hence, it is impossible to know the information contained in the ciphertexts. The main limitation was due to the unavailability of the method to compare the ciphertexts. In 2010, Yang et al. proposed an public key encryption with equality test (PKEwET) to compare whether two ciphertexts contain the same information without decryption [20]. Since, Yang et al's scheme allows all users to perform an equality test of ciphertexts, researchers have further studied the authorization algorithm of equality test and the security of the scheme [21–25].

PKEwET is a promising cryptographic algorithm for CWSNs due to its practical applicability. The application scenario of PKEwET in context of CWSNs is depicted in Fig 1. Precisely, the users send data to sensor networks that further proceed the ciphertexts to cloud service through gateway for storage. For retrieval, the users send trapdoors and ciphertexts to the cloud service. After receiving trapdoors, the cloud service tests whether the received ciphertexts are consistent with the stored and returns the result to users.

To merge the functionality of ciphertext matching in RSA scheme [26–28], the construction of public key encryption with equality test based on RSA is proposed in this paper. For security enhancement, Fujisaki and Okamoto method is introduced into the proposed scheme. In general, the advantages of the proposed scheme can be summarized as follows:

- The idea of public key encryption with equality test is introduced into RSA scheme. The proposed scheme fills the gap of RSA algorithm in the context of equality test over ciphertext. The major target of this paper is to make the RSA algorithm enjoying the equality test of ciphertexts. To the best of our knowledge, this is the novel algorithm of RSA with equality test.
- A simple and efficient Fujisaki and Okamoto method is introduced to enhance the security of the proposed scheme. More precisely, a semantically secure public-key encryption scheme against passive adversaries is improved to a non-malleable public-key encryption scheme against adaptive chosen ciphertext attacks in the random oracle model.

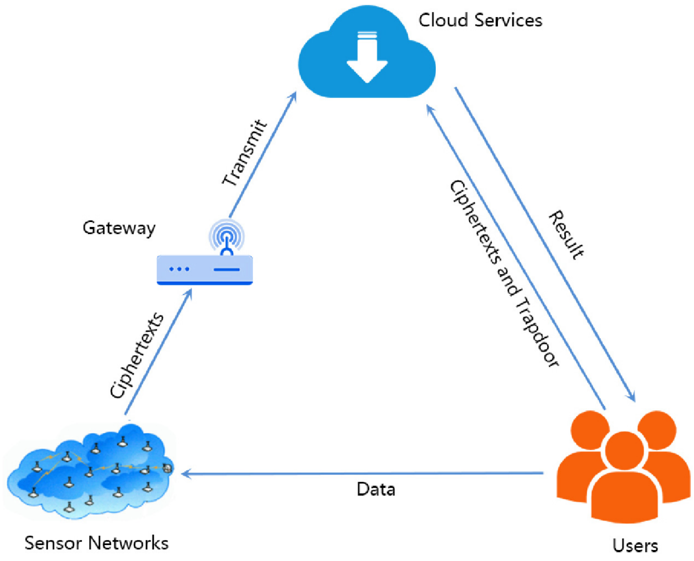

Fig 1. The framework of the mechanism proposed.

https://doi.org/10.1371/journal.pone.0258746.g001

- To prove the efficiency of the proposed scheme, the performance analysis is presented on 512, 1024 and 2048 bits. The time of algorithm is compared in context of KenGen, encryption, decryption and test algorithms. These comparisons validate the claim of scheme efficiency.
- Since, the equality test algorithm is based on a ray. Therefore, comparing the proposal with the schemes based on bilinear pairing, it is simpler and easier to implement.

2 Background knowledge

2.1 Public-key encryption with equality test

Public-key encryption with equality test (PKEwET) allows anyone to test whether underlying ciphertexts are equal without decryption. Numerous researchers introduced authorization to the PKEwET scheme. Ma et al. proposed a primitive called PKEwET supporting flexible authorization (PKEwET-FA), which provides 4-Types flexible authorization in four different scenarios [29]. Subsequently, Zhu et al. [30] and Lin et al. [31] improved the scheme of Ma's. In order to simplify the certificate management of PKEwET, Ma et al. combined the concepts of PKEwET and identity-based encryption to present identity-based encryption with equality test (IBEwET) [32]. In 2017, Wu et al. improved Ma et al.'s scheme by reducing time computational cost [33]. Duong et al. proposed the first lattice-based PEKwET scheme in the standard model [34]. In 2020, Chen et al. introduced the equality test algorithm into blockchain and proposed blockchain-based proxy re-encryption with equality test [35].

2.2 Cloud-assisted wireless sensor networks

With the increasing popularity of cloud-assisted wireless sensor networks (CWSNs), the maintenance of data confidentiality has become the new challenge [36]. Therefore, several cryptographic algorithms are introduced for CWSNs environment. Wang et al. introduced homomorphic encryption into CWSNs to present a data division scheme. In the proposed scheme, even if a forwarding node is compromised, the attacker may not be able to eavesdrop on the data [37]. In 2018, Wang et al. proposed a systematical evaluation framework for schemes to be assessed objectively [38], revisiting two foremost schemes proposed by Wu et al. [39] and Srinivas et al. [40]. The authors provided the missing evaluation for two-factor schemes in industrial CWSNs. In 2020, Li et al. proposed a multi-conditional proxy broadcast re-encryption scheme for sensor networks [41]. To offer a high level of confidentiality, Maria Azees et al. proposed an efficient affine cipher-based encryption technique [42]. Due to the decentralized nature of blockchain technology, Maria Azees et al. proposed an anonymous authentication scheme based on blockchain [43]. The proposed scheme diminishes the computational cost substantially.

In order to facilitate users to extract data from the database, Boneh et al. [44] and Bellare et al. [45] proposed searchable encryption (SE) and deterministic encryption (DE), respectively. In the application of CWSNs, SE is more useful than DE due to the searching facility is public-key encryption [46]. However, with the development of CWSNs, SE has observed some limitations, such as it is unable to perform the ciphertexts matching without decryption. To meet this challenge, we introduce PKEwET algorithm for CWSNs.

2.3 Notation and conventions

We denote k as the security parameter. Where, 1^k is the string consisting of consecutive "1" bits. $A \parallel B$ is referring to the connection of strings A and B. $\{0, 1\}^*$ is denoting the strings generated by 0 and 1. $y \leftarrow \mathcal{A}^{\mathcal{O}_1,\mathcal{O}_2,\mathcal{O}_3}(x_1, x_2, \ldots)$ denotes a probabilistic algorithm. It takes x_1, $x_2, \cdots$ as inputs, and outputs y. It allows to access the random oracles $\mathcal{O}_1, \mathcal{O}_2, \mathcal{O}_3$ before outputing the y. $x \in S$ indicates that x is randomly selected from the set S.

2.4 Public key encryption with equality test

A PKEwET scheme is comprised of four algorithms: KeyGen, Encrypt, Decrypt and Test. The KeyGen algorithm takes a security parameter k as input and outputs public key pk and secret key sk. The Encrypt algorithm takes pk and a message M as inputs, and returns a ciphertext CT. The Decrypt algorithm takes CT and sk as inputs, and outputs the M. The Test algorithm takes the trapdoor tr, two ciphertexts CT_i and CT_j as inputs and outputs 1 or 0. As described in Fig 2, Alice uses the public key of Bob to encrypt a message M_A and generates the ciphertext CT_A for Bob. Bob uses the public key of Alice to encrypt a message M_B and generates the ciphertext CT_B for Alice. Then, the third user can perform the test algorithm and checks that whether CT_A and CT_B contain the same message. If CT_A and CT_B contain the same message, the third user outputs 1. Otherwise, the third user outputs 0.

2.5 Security of PKE

Here, we present two definitions of security for PKE.

Definition 1 *One-way against chosen-ciphertext attack (OW-CCA) security: The attacker can decrypt queries at any time except for the target ciphertext CT^*, and the corresponding message M cannot be obtained from the public key and CT^*.*

Definition 2 *Indistinguishable against chosen ciphertext attacks (IND-CCA) security: The attacker can decrypt queries at any time except for the target ciphertext CT^*, and selects M_0 and M_1, then the challenger randomly selects $b \in \{0, 1\}$ and generates the target ciphertext CT^* by M_b. The attacker cannot guess the value of b by using ciphertext CT^*.*

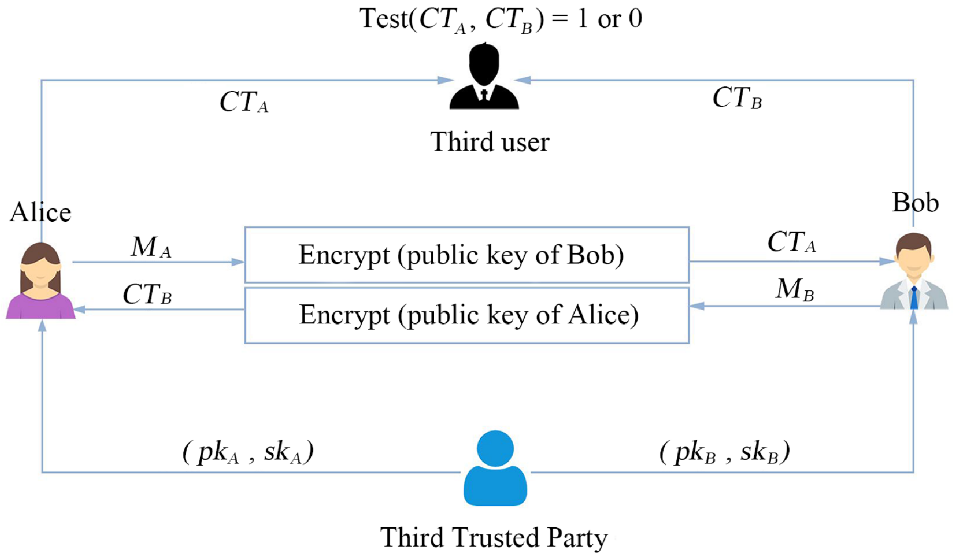

Fig 2. The PKEwET framework [18].

https://doi.org/10.1371/journal.pone.0258746.g002

2.6 Organization

The rest of this paper is organized as follows: In Section 3, the security model is discussed. In Section 4, the details of the new algorithm are presented. The security of the proposed scheme is discussed in Section 5. In Section 6, the efficiency of the algorithm is evaluated by experiments. Finally, we summarize the work in Section 7.

3 Security models

In this section, we describe two different types of adversaries based on the adversarial permissions.

- Type-1 adversary: We authorize this adversary a trapdoor. So this type of adversary cannot recover the plaintext with the challenge ciphertext.
- Type-2 adversary: To this adversary, we do not authorize the trapdoor. So this type of adversary cannot decide that the CT^* is encrypted through which message.

First, we define OW-CCA security to the Type-1 adversary in RSA with Equality Test scheme (RSA-FO-ET).

Game 1: Suppose that $\mathcal{A}_1$ is a Type-1 adversary and $\mathcal{S}$ is the simulator. The game between $\mathcal{A}_1$ and $\mathcal{S}$ is presented in Table 1:

Table 1. The security model of Type-1 adversary.

```
Experiment $Exp^{OW-CCA}_{\mathcal{S},\mathcal{A}_1}$
$(pk, sk) \leftarrow KeyGen(1^k, sp)$
$M \leftarrow \mathcal{A}_1^{\mathcal{O}_1,\mathcal{O}_2,\mathcal{O}_3}(i, pk)$
$CT_t^* \leftarrow Enc(pk, M)$
$M^* \leftarrow \mathcal{A}_1^{\mathcal{O}_4,\mathcal{O}_5,\mathcal{O}_6}(i, pk, CT_t^*)$
if $M = M^*$, then return 1;
else return 0
```

https://doi.org/10.1371/journal.pone.0258746.t001

Here, $\mathcal{O}_1(i) \triangleq KeyGen(i, sp)$, $\mathcal{O}_2(i, CT_i) \triangleq dec(sk_i, CT_i)$, $\mathcal{O}_3(i, \cdot) \triangleq ET - Auth(sk_i, \cdot)$, $\mathcal{O}_6 = \mathcal{O}_3$, but

$$\mathcal{O}_4(i) = \begin{cases} \mathcal{O}_1(i) & i \neq t \\ \perp & otherwise \end{cases}$$

$$\mathcal{O}_5(i, CT_i) = \begin{cases} \mathcal{O}_2(i, CT_i) & i \neq t, CT_i \neq CT_t^* \\ \perp & otherwise \end{cases}$$

The advantage of $\mathcal{A}_1$ in the aforementioned game is defined as follows:

$$Adv^{OW-CCA}_{RSA-FO-ET,\mathcal{A}_1}(k) = \Pr[M = M*]$$

As described in Table 1, the *KeyGen* algorithm takes 1^k and *sp* as inputs, then outputs the public key *pk* and private key *sk*. $\mathcal{A}_1$ asks for key queries, decryption queries and authorization queries. Then $\mathcal{S}$ initiate the challenge phase. $\mathcal{S}$ chooses a message *M* and outputs ciphertext CT_t^* by performing encryption algorithm (*Enc*(*pk*, *M*)). Then, $\mathcal{A}_1$ inquires more queries in phase 2, including key queries, decryption queries and authorization queries. But it must satisfy the requirements of $\mathcal{O}_4$ and $\mathcal{O}_5$. Afterwards, $\mathcal{A}_1$ outputs a message of *M*, a guessing of CT_t^*.

Definition 3 *The RSA-FO-ET scheme is OW-CCA secure if all polynomial time adversaries' advantage is negligible in the above game.*

Next, we define the IND-CCA security to the Type-2 adversary in RSA-FO-ET.

Game 2: Suppose that $\mathcal{A}_2$ is a Type-2 adversary and $\mathcal{S}$ is the simulator. The game between $\mathcal{A}_2$ and $\mathcal{S}$ is presented in Table 2:

Here, $\mathcal{O}_1(i) \triangleq KeyGen(i, sp)$, $\mathcal{O}_2(i, CT_i) \triangleq dec(sk_i, CT_i)$, but

$$\mathcal{O}_3(i) = \begin{cases} \mathcal{O}_1(i) & i \neq t \\ \perp & otherwise \end{cases}$$

$$\mathcal{O}_4(i, CT_i) = \begin{cases} \mathcal{O}_2(i, CT_i) & i \neq t, CT_i \neq CT^* \\ \perp & otherwise \end{cases}$$

The advantage of $\mathcal{A}_2$ in the aforementioned game is defined as follows:

$$Adv^{IND-CCA}_{RSA-FO-ET,\mathcal{A}_2}(k) = |\Pr[b = b*] - 1/2|$$

As mentioned in Table 2, the *KeyGen* algorithm takes 1^k and *sp* as inputs, then outputs the public key *pk* and private key *sk*. $\mathcal{A}_2$ asks for key queries and decryption queries. $\mathcal{A}_2$ chooses

Table 2. The security model of Type-2 adversary.

```
Experiment $Exp^{IND-CCA}_{\mathcal{S},\mathcal{A}_2}$
$(pk, sk) \leftarrow KeyGen(1^k, sp)$
$M_0, M_1 \leftarrow \mathcal{A}_2^{\mathcal{O}_1,\mathcal{O}_2(i,pk)}$
$b \xleftarrow{R} \{0,1\}$
$CT_t^* \leftarrow Enc(pk, M_b)(b \in 0,1)$
$b^* \leftarrow \mathcal{A}_2^{\mathcal{O}_3,\mathcal{O}_4}(i, pk, CT^*)$
if $b = b^*$, then return 1;
else return 0
```

https://doi.org/10.1371/journal.pone.0258746.t002

two message M_0 and M_1 for $\mathcal{S}$. Then, $\mathcal{S}$ performs challenge phase. $\mathcal{S}$ chooses a message M_b ($b \overset{R}{\leftarrow} \{0, 1\}$) and outputs ciphertext CT_t^* by performing encryption algorithm ($Enc(pk, M_b)$). Then, $\mathcal{A}_2$ inquires more queries in phase 2, such as key queries and decryption queries. But it must satisfy the requirements of $\mathcal{O}_4$ and $\mathcal{O}_5$. Then, $\mathcal{A}_2$ outputs the guess of b.

Definition 4 *The RSA-FO-ET scheme is IND-CCA secure if all polynomial time adversaries' advantage is negligible in the above game.*

4 Proposed construction

In this part, we present the details of proposed PKEwET-FA-RSA scheme as follows.

1) **Setup**(k): To generate the system public parameters sp, the algorithm takes a security parameter k as input and executes as follows:

- Choose the hash functions: H_1: $\{0, 1\}^k \rightarrow \{0, 1\}^k$, H_2: $\{0, 1\}^{2k} \rightarrow \{0, 1\}^k$, H_3: $\{0, 1\}^{5k} \rightarrow \{0, 1\}^{2l}$, and H_4, H_5: $\{0, 1\}^k \rightarrow Z_q$, here l expresses the length of elements in Z_q.

2) **KeyGen**(sp): This algorithm generates public keys and private keys, the details are discussed as follows:

- Select four large prime numbers p_1, q_1, p_2 and q_2 randomly and keep them confidential.;
- Compute $N_1 = p_1{}^*q_1$ and $\varphi(N_1) = (p_1 - 1)^*(q_1 - 1)$. Here $\varphi(N_1)$ is the Euler function value of N_1;
- Compute $N_2 = p_2{}^*q_2$ and $\varphi(N_2) = (p_2 - 1)^*(q_2 - 1)$. Here $\varphi(N_2)$ is the Euler function value of N_2;
- Select integers e_1, e_2 randomly. Here e_1, e_2 satisfy the following conditions:
  - $1 < e_1 < \varphi(N_1)$, $1 < e_2 < \varphi(N_2)$;
  - $gcd(e_1, \varphi(N_1)) = 1$ and $gcd(e_2, \varphi(N_2)) = 1$.
- Compute d_1 and d_2. Here d_1 and d_2 satisfy the conditions $d_1{}^*e_1 = 1 \bmod \varphi(N_1)$ and $d_2{}^*e_2 = 1 \bmod \varphi(N_2)$, respectively;
- Output the public key $pk = (e_1, e_2, N_1, N_2)$ and secret key $sk = (d_1, d_2)$.

3) **Encrypt**(M, pk): This algorithm outputs the ciphertext $CT = (C_1, C_2, C_3, C_4, C_5)$ as follows:

- Generate a ray.
  - Use H_4, H_5 to generate a point $p = (H_4(M), H_5(M))$;
  - Construct a ray $f(x)$ using the point p and the origin;
- To generate a point on the line randomly, choose $x_1 \in \{0, 1\}^l$ randomly and compute $f(x_1) = y_1$. If $x_1 = 0$, then, take $x_1 \in \{0, 1\}^l$ randomly again.
- Choose $r_1, r_2 \in Z_q^*$ randomly. Then, output the ciphertext $CT = (C_1, C_2, C_3, C_4, C_5)$ as follows:

$$C_1 = r_1^{e_1} \tag{1}$$

$$C_2 = r_2^{e_2} \tag{2}$$

$$C_3 = M||r_2 \oplus H_1(r_1) \tag{3}$$

$$C_4 = H_2(M, r_1) \tag{4}$$

$$C_5 = x_1||y_1 \oplus H_3(r_2, C_1, C_2, C_3, C_4) \tag{5}$$

4) **Decrypt**(CT, sk): On input sk and a ciphertext $CT = (C_1, C_2, C_3, C_4, C_5)$, the algorithm computes as follows:

$$r_1 = C_1^{d_1} \tag{6}$$

$$M||r_2 = C_3 \oplus H_1(r_1) \tag{7}$$

$$x_1||y_1 = C_5 \oplus H_3(r_2, C_1, C_2, C_3, C_4) \tag{8}$$

Use M to generate $f(x)$ as in algorithm **Encrypt**. Then, check whether $C_2 = r_2^{e_2}$, $C_4 = H_2(M, r_1)$, $C_5 = x_1||y_1 \oplus H_3(r_2, C_1, C_2, C_3, C_4)$ and $f(x_1) = y_1$ all hold. If yes, the algorithm outputs M; otherwise, it outputs $\perp$.

5) **The authorization and test algorithm**:

Suppose u_i and u_j are two users in the system and $CT_i = (C_{i,1}, C_{i,2}, C_{i,3}, C_{i,4}, C_{i,5})$ (resp. $CT_j = (C_{j,1}, C_{j,2}, C_{j,3}, C_{j,4}, C_{j,5})$) is a ciphertext of u_i (resp. u_j). $r_{2,i} \in Z_q^*$ (resp. $r_{2,j} \in Z_q^*$) denotes a randomness used in the generation of CT_i (resp. CT_j).

- The authorization algorithm (Auth) performs as follows:
  For user u_i, the authorized private key is $d_{i,2}$ and the trapdoor is $td_i = d_{i,2}$;
  For user u_j, the authorized private key is $d_{j,2}$ and the trapdoor is $td_j = d_{j,2}$;
- The test algorithm (Test) performs as follows:
  This algorithm takes inputs: td_i, td_j and CT_i, CT_j and computes as follows:

$$r_{i,2} = C_{i,2}^{d_{i,2}} \tag{9}$$

$$r_{j,2} = C_{j,2}^{d_{j,2}} \tag{10}$$

$$x_i||y_i = C_{i,5} \oplus H_3(r_{i,2}, C_{i,1}, C_{i,2}, C_{i,3}, C_{i,4}) \tag{11}$$

$$x_j||y_j = C_{j,5} \oplus H_3(r_{j,2}, C_{j,1}, C_{j,2}, C_{j,3}, C_{j,4}) \tag{12}$$

Use x_i, y_i and x_j, y_j to check

$$x_i^{-1} * y_i = x_j^{-1} * y_j \tag{13}$$

whether holds.

Output 1 if established, otherwise output 0.

5 Security analysis

This section analyzes the security of the proposed scheme and authorization.

Table 3. Phase1 of Game 1.

$\mathcal{S}$	**Public Channel**	$\mathcal{A}_1$	**Queries**
If the query α_i already exists in the H_1 list in the form (α_i, β_i)	$\xleftarrow{\alpha_i}$	Chooses α_i	H_1-query
	$\xrightarrow{\beta_i}$	Gets β_i	
Otherwise $\mathcal{S}$ chooses $\beta_i \in \{0, 1\}^k$ randomly and adds the tuple (α_i, β_i) to the H_1 list	$\xrightarrow{\beta_i}$	Gets β_i	
If the query (η_i, θ_i) already exists in the H_2 list in the form $(\eta_i, \theta_i, \vartheta_i)$	$\xleftarrow{\eta_i,\theta_i}$	Chooses (η_i, θ_i)	H_2-query
	$\xrightarrow{\vartheta_i}$	Gets ϑ_i	
Otherwise $\mathcal{S}$ chooses $\vartheta_i \in \{0, 1\}^k$ randomly and adds the tuple $(\eta_i, \theta_i, \vartheta_i)$ to the H_2 list	$\xrightarrow{\vartheta_i}$	Gets ϑ_i	
If the query $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i, \xi_i)$ already exists in the H_3 list in the form $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i, \xi_i)$	$\xleftarrow{\delta_i,\epsilon_i,\varepsilon_i,\zeta_i,\nu_i}$	Chooses $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i)$	H_3-query
	$\xrightarrow{\xi_i}$	Gets ξ_i	
Otherwise $\mathcal{S}$ chooses $\xi_i \in \{0, 1\}^{2l}$ randomly and adds the tuple $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i, \xi_i)$ to the H_2 list	$\xrightarrow{\xi_i}$	Gets ξ_i	
KeyGen(pp, i)	$\xleftarrow{i}$	Chooses $i(i \neq t)$	Decryption key queries
	$\xrightarrow{sk_i}$		
Looks at H_1, H_2 and H_3 lists $\mathcal{S}$ runs algorithm	$\xleftarrow{CT}$	Chooses CT	Decryption queries
Decrypt(CT, sk_i)	$\xrightarrow{M_i or \perp}$	Gets M_i or $\perp$	
KeyGen(pp, i)	$\xleftarrow{i}$	Chooses i	Authorization queries
	$\xrightarrow{sk_{i,2}}$	Gets $sk_{i,2}$	

https://doi.org/10.1371/journal.pone.0258746.t003

Theorem 1 *Under the random oracle model, the proposed RSA-FO-ET scheme is OW-CCA secure against Type-1 adversary.*

Let $\mathcal{A}_1$ be a Type-1 adversary breaking the RSA-FO-ET scheme in polynomial-time. We provide $CT^* = (C_1^*, C_2^*, C_3^*, C_4^*, C_5^*)$ to the simulator $\mathcal{S}$. The aim of $\mathcal{S}$ is to recover the plaintext of CT^* with a non-negligible advantage. Here, CT^* is the challenge ciphertext that is generated by the challenging algorithm.

The game between $\mathcal{A}_1$ and $\mathcal{S}$ is described as follows:

First $\mathcal{A}_1$ chooses t as his target at the beginning of the game. During the game, $\mathcal{S}$ maintains three watch lists of H_1, H_2 and H_3 and responds to $\mathcal{A}_1$ for all queries.

- 1) **Setup(1^k)**: This algorithm takes a security parameter 1^k as input and outputs the system public parameters $sp = (H_1, H_2, H_3)$. Then, $\mathcal{S}$ calls the *KeyGen* algorithm, generates public/private key pairs (pk, sk), and provides the public key to $\mathcal{A}_1$.
- 2) **Phase1**: $\mathcal{A}_1$ inquires H_1, H_2 and H_3 queries, decryption queries and authorization queries as he/she wants. The H_1, H_2 and H_3 lists are initially empty. Then, $\mathcal{S}$ outputs the results accurately. Detailed descriptions are as follows:
  $\mathcal{S}$ maintains a list of 2-tuples (α_i, β_i) in H_1, a list of 3-tuples $(\theta_i, \vartheta_i, \eta_i)$ in H_2 and a list of 6-tuples $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i, \xi_i)$ in H_3. Detailed maintenances are shown in Table 3.
- 3) **Challenge**: $\mathcal{S}$ chooses $M \subset \mathcal{M}$, $r_{t,1}^* \in Z_q^*$ and $r_{t,2}^* \in Z_q^*$. Then outputs CT^* as shown in Table 4.
- 4) **Phase2**: $\mathcal{A}_1$ inquires more queries as in Phase 1. But there is a condition:
  During decryption queries process, the ciphertext of t is not allowed to be queried.
- 5) **Guess**: $\mathcal{A}_1$ outputs a message $M^* \subset \mathcal{M}$. If $M^* = M$, it means that $\mathcal{S}$ wins the game. Otherwise, it fails.

Theorem 2 *Under the random oracle model, the proposed RSA-FO-ET scheme is IND-CCA secure against Type-2 adversary.*

Table 4. Challenge.

$\mathcal{S}$	**Public Channel**	$\mathcal{A}_1$
Chooses M and computes $C_1^* = r_{t,1}^{*e_1}$ $C_2^* = r_{t,2}^{*e_2}$ $C_3^* = M\|\|r_2 \oplus H_1(r_{t,1}^*)$ $C_4^* = H_2(M, r_{t,1}^*)$ $C_5^* = x_{t,1}\|\|y_{t,1}$ $\oplus H_3(r_{t,2}^*, C_1^*, C_2^*, C_3^*, C_4^*)$	$\xrightarrow{CT=(C_1^*,C_2^*,C_3^*,C_4^*,C_5^*)}$	Gets CT Inquires more queries as phase 2

https://doi.org/10.1371/journal.pone.0258746.t004

Let $\mathcal{A}_2$ be Type-2 adversary breaking the RSA-FO-ET scheme in polynomial-time. We provide $CT^* = (C_1^*, C_2^*, C_3^*, C_4^*, C_5^*)$ to the simulator $\mathcal{S}$. The aim of $\mathcal{S}$ is to decide the plaintext of CT^* is encrypted by M_0 or M_1 with a non-negligible advantage. Here, CT^* is the challenge ciphertext that is generated by the challenging algorithm.

The game between $\mathcal{A}_2$ and $\mathcal{S}$ is described as follows:

First $\mathcal{A}_2$ chooses t as a target at the beginning of the game. During the game, $\mathcal{S}$ maintains three watch lists of H_1, H_2 and H_3, then, responds $\mathcal{A}_2$ to all queries.

- 1) **Setup(1^k)**: This algorithm takes a security parameter 1^k as input and outputs the system public parameters $sp = (H_1, H_2, H_3)$. Then, $\mathcal{S}$ calls the *KeyGen* algorithm and to generate public/private key pairs (pk, sk), and provides the public key to $\mathcal{A}_2$.
- 2) **Phase1**: $\mathcal{A}_2$ inquires H_1, H_2 and H_3 queries, decryption queries and authorization queries as he/she wants. The H_1, H_2 and H_3 lists are initially empty. Then, $\mathcal{S}$ outputs the results accurately. Detailed descriptions are as follows:
  $\mathcal{S}$ maintains a list of 2-tuples (α_i, β_i) in H_1, a list of 3-tuples $(\theta_i, \vartheta_i, \eta_i)$ in H_2 and a list of 6-tuples $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i, \xi_i)$ in H_3. Detailed maintenances are shown in Table 5.
- 3) **Challenge**: $\mathcal{A}_2$ submits two equal-length messages $M_0, M_1 \subset \mathcal{M}$. $\mathcal{S}$ chooses $b \in \{0, 1\}$, $r_{t,1}^* \in Z_q^*$ and $r_{t,2}^* \in Z_q^*$. Then outputs CT^* as shown in Table 6.

Table 5. Phase 1 of Game 2.

$\mathcal{S}$	**Public Channel**	$\mathcal{A}_1$	**Queries**
If the query α_i already exists in the H_1 list in the form (α_i, β_i)	$\xleftarrow{\alpha_i}$	Chooses α_i	H_1-query
	$\xrightarrow{\beta_i}$	Gets β_i	
Otherwise $\mathcal{S}$ chooses $\beta_i \in \{0, 1\}^k$ randomly and adds the tuple (α_i, β_i) to the H_1 list	$\xrightarrow{\beta_i}$	Gets β_i	
If the query (η_i, θ_i) already exists in the H_2 list in the form $(\eta_i, \theta_i, \vartheta_i)$	$\xleftarrow{\eta_i,\theta_i}$	Chooses (η_i, θ_i)	H_2-query
	$\xrightarrow{\vartheta_i}$	Gets ϑ_i	
Otherwise $\mathcal{S}$ chooses $\vartheta_i \in \{0, 1\}^k$ randomly and adds the tuple $(\eta_i, \theta_i, \vartheta_i)$ to the H_2 list	$\xrightarrow{\vartheta_i}$	Gets ϑ_i	
If the query $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i, \xi_i)$ already exists in the H_3 list in the form $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i, \xi_i)$	$\xleftarrow{\delta_i,\epsilon_i,\varepsilon_i,\zeta_i,\nu_i}$	Chooses $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i)$	H_3-query
	$\xrightarrow{\xi_i}$	Gets ξ_i	
Otherwise $\mathcal{S}$ chooses $\xi_i \in \{0, 1\}^{2l}$ randomly and adds the tuple $(\delta_i, \epsilon_i, \varepsilon_i, \zeta_i, \nu_i, \xi_i)$ to the H_2 list	$\xrightarrow{\xi_i}$	Gets ξ_i	
KeyGen(pp, i)	$\xleftarrow{i}$	Chooses $i(i \neq t)$	Decryption key queries
	$\xrightarrow{sk_i}$		
Looks at H_1, H_2 and H_3 lists $\mathcal{S}$ runs algorithm	$\xleftarrow{CT}$	Chooses CT	Decryption queries
Decrypt(CT, sk_i)	$\xrightarrow{M_i or \perp}$	Gets M_i or $\perp$	
KeyGen(pp, i)	$\xleftarrow{i}$	Chooses i	Authorization queries
	$\xrightarrow{sk_{i,2}}$	Gets $sk_{i,2}$	

https://doi.org/10.1371/journal.pone.0258746.t005

Table 6. Challenge.

$\mathcal{S}$	Public Channel	$\mathcal{A}_2$
Chooses $b \in \{0, 1\}$ and computes $C_1^* = r_{t,1}^{*e_1}$ $C_2^* = r_{t,2}^{*e_2}$ $C_3^* = M_b \| \| r_2 \oplus H_1(r_{t,1}^*)$ $C_4^* = H_2(M_b, r_{t,1}^*)$ $C_5^* = x_{t,1} \| \| y_{t,1}$ $\oplus H_3(r_{t,2}^*, C_1^*, C_2^*, C_3^*, C_4^*)$	$\xleftarrow{M_0,M_1}$ $\xrightarrow{CT^*=(C_1^*,C_2^*,C_3^*,C_4^*,C_5^*)}$	Chooses M_0, M_1 Gets CT^* Inquires more queries as phase 2 as phase 2

https://doi.org/10.1371/journal.pone.0258746.t006

Table 7. The comparison with other schemes.

Scheme	Confidentiality	Encrypted Matching	Authorized	Ciphertext Test
[26]	√	√	-	-
[27]	√	√	-	-
[28]	√	√	-	-
Our scheme	√	√	√	√

https://doi.org/10.1371/journal.pone.0258746.t007

- 4) **Phase2**: $\mathcal{A}_2$ inquires more queries as in Phase 1. But there are two conditions: During decryption queries process, the ciphertext of t is not allowed to be queried. During Authorization queries process, the authorization of user t is not allowed to be queried.
- 5) **Guess**: $\mathcal{A}_2$ outputs a guess $b^* \in \{0, 1\}$. If $b^* = b$, it means that $\mathcal{S}$ wins the game. Otherwise, it fails.

6 Performance analysis

In this section, we discuss the performance of the proposed scheme. More precisely, the safety parameters of 512 bits, 1024 bits and 2048 bits are tested as depicted in Tables 8–10. The prototype system is developed using go 1.14.4. It is executed on intel(R) Core(TM) i7-6700 CPU 2.6GHz, 16.00GB of RAM. Fig 3 describes the performance of the proposed scheme in terms of the KenGen, Encrypt, Decrypt and Equality test algorithms. As shown in Fig 3, we

Table 8. Computational efficiency of 512bit.

Algorithm	runtime(ms)
KenGen	0.39
Encrypt	0.64
Decrypt	0.58
Test	0.12

https://doi.org/10.1371/journal.pone.0258746.t008

Table 9. Computational efficiency of 1024bit.

Algorithm	runtime(ms)
KenGen	3.5
Encrypt	4.9
Decrypt	4.9
Test	23

https://doi.org/10.1371/journal.pone.0258746.t009

Table 10. Computational efficiency of 2048bit.

Algorithm	runtime(s)
KenGen	1.6
Encrypt	3.7
Decrypt	4.6
Test	7

https://doi.org/10.1371/journal.pone.0258746.t010

performed the experiment on its runtime while increasing length of the security parameter. It may be observed that the proposed scheme depicts efficiency. In Table 7, we compare the proposed scheme with other approaches. All of the previous RSA schemes realize protection of the confidentiality of data, while only the proposed scheme supports ciphertext data matching as well. In Table 8, the computational efficiency of 512 bits is depicted. In Table 9, the computational efficiency of 1024 bits is shown. The computational efficiency of 2048 bits is shown in Table 10. In Table 11, we present the comparison with the earlier PKEwET schemes while

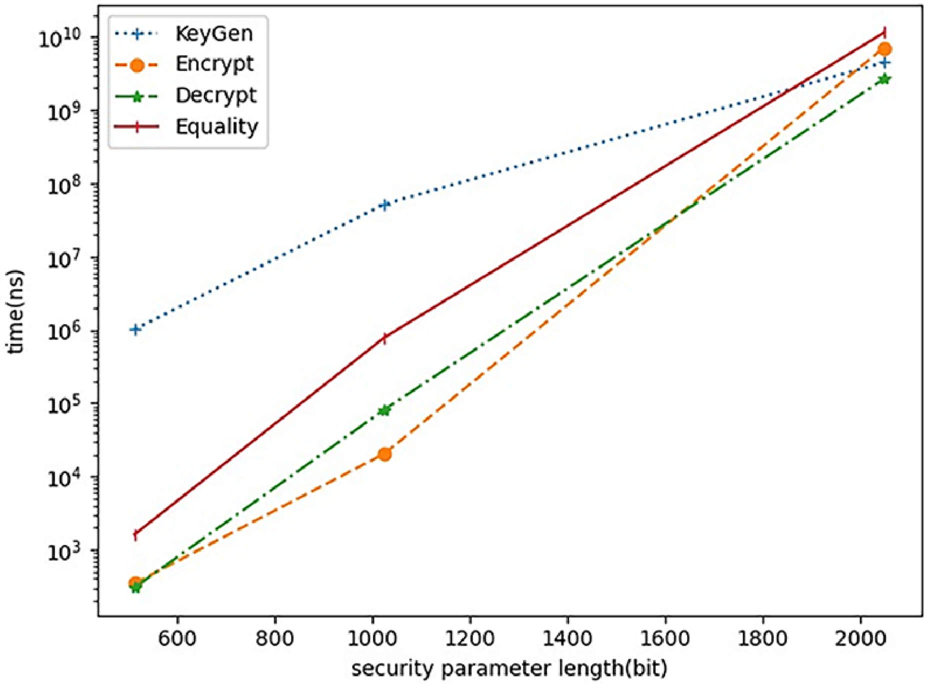

Fig 3. The performance of the proposed scheme.

https://doi.org/10.1371/journal.pone.0258746.g003

Table 11. The comparison of computational complexity with others.

	[29]	[32]	[30]	[31]	Our scheme
C_{Enc}	6E	6E+2P	3E+2I	4E+3I	2E+1I
C_{Dec}	5E	2E+2P+2I	2E+1I	3E+3I	2E+1I

C_{Enc} and C_{Dec}: the computation complexity of algorithms forencryption and decryption; E, P and I: the exponentiation operation, the pairing operation and the inversion operation in the group $\mathbb{G}$.

https://doi.org/10.1371/journal.pone.0258746.t011

considering the computation complexity of encryption and decryption algorithms. It shows that the presented scheme achieves lower computational complexity.

7 Conclusions

Different from the previous equality test schemes, a noval RSA with equality test scheme is proposed in this paper. To the best of our knowledge, it is the first attempt to integrate the equality test algorithm into RSA scheme. We introduced two types of attackers based on their privileges. The proposed scheme is proved to be one-way against chosen-ciphertext attack secure and indistinguishable against chosen ciphertext attacks secure. Moreover, we applied experiments on KeyGen, Encrypt, Decrypt and Equality algorithms to verify the rationality of proposed scheme in CWSNs scenario.

Author Contributions

Writing – original draft: Huijun Zhu.

Writing – review & editing: Dong Xie, Haseeb Ahmad, Hasan Naji Hasan Abdullah.

References

1. Stergiou C.L., Psannis K.E. and Gupta B.B. IoT-based big data secure management in the fog over a 6G wireless network. IEEE Internet of Things Journal, 2020, 8(7): 5164–5171. https://doi.org/10.1109/JIOT.2020.3033131
2. Guermazi A., Trabelsi H., Jerbi W. A novel energy consumption approach to extend the lifetime for wireless sensor network. International Journal of High Performance Computing and Networking, 2020, 16(2-3): 160–169. https://doi.org/10.1504/IJHPCN.2020.10034822
3. Cao N., Liu P., Li G., et al. Evaluation Models for the Nearest Closer Routing Protocol in Wireless Sensor Networks. IEEE Access, 2018, 6: 77043–77054. https://doi.org/10.1109/ACCESS.2018.2825441
4. Al-Qerem A., Alauthman M., Almomani A., et al. IoT transaction processing through cooperative concurrency control on fog-cloud computing environment. Soft Computing, 2020, 24(8): 5695–5711. https://doi.org/10.1007/s00500-019-04220-y
5. Jian X., Yang G., Chen Z. and Wang Q. A survey on the privacy-preserving data aggregation in wireless sensor networks. China Commun. 2015; 12: 162–180. https://doi.org/10.1109/CC.2015.7112038
6. A. Boldyreva, S. Fehr and A. O. Neill. On Notions of Security for Deterministic Encryption and Efficient Constructions without Random Oracles. Advances in Cryptology—CRYPTO 2008, 28th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17–21, 2008.
7. Huang M., Yang B., Zhao Y., et al. A generic construction of CCA-secure deterministic encryption. Information Processing Letters, 2020, 154(Feb.): 105865.1–105865.12.
8. X. Peng, L. Shuai, W. Wei, et al. Dynamic Searchable Symmetric Encryption with Physical Deletion and Small Leakage. Australasian Conference on Information Security and Privacy. Springer, Cham, 201: 207–226.
9. D. Boneh, D. G. Crescenzo, R. Ostrovsky, et al. Public key encryption with keyword search. International conference on the theory and applications of cryptographic techniques. Springer, Berlin, Heidelberg, 2004: 506–522.
10. Li H., Yang Y., Dai Y., Bai J., et al. Achieving secure and efficient dynamic searchable symmetric encryption over medical cloud data. IEEE Transactions on Cloud Computing. 2017, 8, 484–494 https://doi.org/10.1109/TCC.2017.2769645
11. Chatterjee S., Puria S. K. P., Shah A. Efficient backward private searchable encryption. Journal of Computer Security. 2020, 1, 1–39.
12. L. Ibraimi, S. Nikova, P. Hartel, and W. Jonker. Public-key encryption with delegated search. International Conference on Applied Cryptography and Network Security. 2011, 532–549.
13. Choi C., Wang T., Esposito C., et al. Sensored Semantic Annotation for Traffic Control Based on Knowledge Inference in Video. IEEE Sensors Journal, 2021, 21(10): 11758–11768. https://doi.org/10.1109/JSEN.2020.3048758
14. Li H., Tian H., Zhang F., He J. Blockchain-based searchable symmetric encryption scheme. Computers and Electrical Engineering. 2019, 73, 32–45. https://doi.org/10.1016/j.compeleceng.2018.10.015

15. Chen R., Mu Y., Yang G., et al. Dual-server public-key encryption with keyword search for secure cloud storage. IEEE Trans. Information Forensics and Security. 2016, 11, 789–798. https://doi.org/10.1109/TIFS.2016.2599293

16. Chen R., Mu Y., Yang G., et al. Server-aided public key encryption with keyword search. IEEE Trans. on Information Forensics and Security. 2016, 11, 2833–2842. https://doi.org/10.1109/TIFS.2016.2599293

17. K. Huang and R. Tso. Provable secure dual-server public key encryption with keyword search. Proc. IVSW. 2017, 39–44.

18. M. K. Yadav, D. Gugal, S. Matkar and S. Waghmare, Encrypted Keyword Search in Cloud Computing using Fuzzy Logic. 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT). 2019, 1–4.

19. Lu Y. and Li J. Constructing certificateless encryption with keyword search against outside and inside keyword guessing attacks. In China Communications. 2019, 16, 156–173. https://doi.org/10.23919/JCC.2019.07.013

20. G. Yang, C. H. Tan, Q. Huang, et al. Probabilistic public key encryption with equality test. Cryptographers Track at the RSA Conference. 2010, 119–131.

21. Wu L., Zhang Y., Choo K. et al. Efficient and secure identity-based encryption scheme with equality test in cloud computing. Future Generation Computer Systems. 2017, 73, 22–31. https://doi.org/10.1016/j.future.2017.03.007

23. Sarivougioukas J., Vagelatos A. Modeling Deep Learning Neural Networks With Denotational Mathematics in UbiHealth Environment. International Journal of Software Science and Computational Intelligence, 2020, 12(3):14–27. https://doi.org/10.4018/IJSSCI.2020070102

22. Zhu H., Wang L., Ahmad H., et al. Key-Policy Attribute-Based Encryption With Equality Test in Cloud Computing. IEEE Access, 2017, 5: 20428–20439. https://doi.org/10.1109/ACCESS.2017.2756070

24. Wang Y., Cui Y., Huang Q., Li H., Huang J. and Yang G. Attribute-Based Equality Test Over Encrypted Data Without Random Oracles. IEEE Access. 2020, 8, 32891–32903. https://doi.org/10.1109/ACCESS.2020.2973459

25. Huang K., Chen Y., Tso R. Somewhat semantic secure public key encryption with filtered-equality-test in the standard model and its extension to searchable encryption. Comput. Syst. Sci. 2017, 400–409. https://doi.org/10.1016/j.jcss.2017.06.001

26. R. Patidar, R. Bhartiya. Modified RSA cryptosystem based on offline storage and prime number. 2013 IEEE International Conference on Computational Intelligence and Computing Research. IEEE, 2013: 1–6.

27. Y. Wu and X. Wu, Implementation of efficient method of RSA key-pair generation algorithm. 2017 IEEE International Symposium on Consumer Electronics (ISCE). 2017, 72–73.

28. F. H. M. S. Al-Kadei, H. A. Mardan and N. A. Minas. Speed Up Image Encryption by Using RSA Algorithm. 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, 1302–1307.

29. Ma S., Huang Q., Zhang M. and Yang B. Efficient Public Key Encryption With Equality Test Supporting Flexible Authorization. IEEE Trans. on Information Forensics and Security. 2015, 10, 458–470. https://doi.org/10.1109/TIFS.2014.2378592

30. Zhu H., Wang L., Hase Eb A., et al. Pairing-Free Equality Test over Short Ciphertexts. International Journal of Distributed Sensor Networks, 2017, 13(6): 1550147717715605. https://doi.org/10.1177/1550147717715605

31. Lin X., Sun L., Qu H., et al. Public key encryption supporting equality test and flexible authorization without bilinear pairings. Computer Communications, 2021, 170: 190–199. https://doi.org/10.1016/j.comcom.2021.02.006

32. Ma S. Identity-based encryption with outsourced equality test in cloud computing. Information Sciences, 2016, 389–402. https://doi.org/10.1016/j.ins.2015.08.053

33. Wu L., Zhang Y., Choo K. R., et al. Efficient and secure identity-based encryption scheme with equality test in cloud computing. Future Generation Computer Systems. 2017, 73, 22–31. https://doi.org/10.1016/j.future.2017.03.007

34. D. H. Duong, K. Fukushima, S. Kiyomoto, P. S. Roy and W. Susilo. A lattice-based public key encryption with equality test in standard model. Australasian Conference on Information Security and Pri-vacy. 2019, 138–155.

35. B. Chen, D. He, N. Kumar, et al. A Blockchain-Based Proxy Re-Encryption with Equality Test for Vehicular Communication Systems. IEEE Transactions on Network ence and Engineering. 2020

36. Chen X., Li J., Weng J., Ma J., and Lou W. Verifiable computation over large database with incremental updates. IEEE Transactions on Computers. 2015, 65(10): 3184–3195. https://doi.org/10.1109/TC.2015.2512870

37. Wang X. and Zhang Z., Data division scheme based on homomorphic encryption in WSNs for health care. Journal of Medical Systems, 2015, 39(12): 1–7. https://doi.org/10.1007/s10916-015-0340-1

38. Wang D., Li W., Wang P. Measuring Two-Factor Authentication Schemes for Real-Time Data Access in Industrial Wireless Sensor Networks. IEEE Transactions on Industrial Informatics, 2018, 14(9): 4081–4092. https://doi.org/10.1109/TII.2018.2834351

39. Wu F., Xu L., Kumari S. and Li X. A new and secure authentication scheme for wireless sensor networks with formal proof. Peer Peer Netw. 2017, 10(1): 16–30. https://doi.org/10.1007/s12083-015-0404-5

40. Srinivas J., Mukhopadhyay S. and Mishra D. Secure and efficient user authentication scheme for multi-gateway wireless sensor networks. AdHoc Netw. 2017, 54: 147–169.

41. Li P., Zhu L., Gupta B., et al. A Multi-Conditional Proxy Broadcast Re-Encryption Scheme for Sensor Networks. Computers, Materials and Continua, 2020, 65(3):2079–2090. https://doi.org/10.32604/cmc.2020.013696

42. Azees M., et al. An efficient anonymous authentication and confidentiality preservation schemes for secure communications in wireless body area networks. Wireless Networks, 2021, 27 (3), 2119–2130. https://doi.org/10.1007/s11276-021-02560-y

43. Azees M., et al. BBAAS: Blockchain-Based Anonymous Authentication Scheme for Providing Secure Communication in VANETs. Security and Communication Networks, 2021. 1(2021):1–11.

44. D. Boneh, G. D. Crescenzo, R. Ostrovsky and G. Persiano. Public key encryption with keyword search. International Conference on the Theory and Applications of Cryptographic Techniques. Springer Berlin Heidelberg, 2004: 506–522.

45. M. Bellare, A. Boldyreva and A. ONeill. Deterministic and efficiently searchable encryption. Annual International Cryptology Conference. Springer Berlin Heidelberg, 2007: 535–552.

46. Xu P., He S., Wang W., et al. Lightweight Searchable Public-key Encryption for Cloud-assisted Wireless Sensor Networks. IEEE Transactions on Industrial Informatics, 2017, 14(8): 3712–3723. https://doi.org/10.1109/TII.2017.2784395