

CLOUD DATA SHARING SYSTEMS AND METHODS FOR SHARING DATA USING THE SYSTEMS

FIELD OF THE INVENTION

5

[001] The present invention relates to data security, and in particular to a cloud data sharing system supporting encrypted data search.

BACKGROUND OF THE INVENTION

10

[002] As an attractive computation paradigm, cloud computing is envisioned to provide elastic and adequate computing and storage resources to users in an economical way. With the constant advances in cloud computing technologies, more and more individuals and organizations would outsource their various data to third-party public clouds for ease of sharing or for cost saving.

15

[003] To promote the long-term development of cloud computing, it is important to address the security issues in the cloud-based data sharing system. For data owners who upload their data to cloud storage servers for future use and sharing, it is necessary to ensure data security so that even cloud servers cannot obtain the owners' data. To fulfil this goal, the technique of identity-based encryption¹ can be adopted, which enables each data consumer to decrypt only the encrypted data associated with his/her identity.

20

[004] Although data owners can encrypt their data by the identity-based encryption technique and outsource the encrypted data on the cloud, such one-to-one encryption mode of identity-based encryption cannot realize scalable access control on cloud data. Practically, data owners usually wish that all authorized data consumers have access to their data. The technique of attribute-based encryption can be adopted to realize scalable access control on cloud data. Attribute-based encryption can be classified into two categories: key-policy attribute-based encryption² and ciphertext-policy attribute-based encryption³. Compared with key-policy attribute-based encryption, ciphertext-policy attribute-based encryption is more suitable for the cloud computing environment because it puts the access policy decision right in the data owners' hands.

25

30

[005] However, if the ciphertext-policy attribute-based encryption technique is directly used in cloud data sharing, the cloud server cannot perform keyword searches on the encrypted data, which significantly limits its practical applications. To realize keyword search on the encrypted data, a public-key encryption scheme with keyword search⁴ was proposed, however, the scheme only allows the user with the corresponding secret key to search on encrypted data. Although various researches have been conducted on the techniques of searchable encryption to realise keyword search over encrypted data, no solution has been proposed to efficiently realise the function that any registered data consumer in the system may perform search on the encrypted data from any data owner.

35

40

[006] On the other hand, with ciphertext-policy attribute-based encryption technique, although data owners can encrypt their data and outsource the encrypted data on the cloud, the decryption is inefficient as multiple bilinear pairing operations are involved. In real life cases, data consumers may try to access cloud data through resource-limited devices, which requires the decryption to be performed efficiently.

[007] Further, user revocation is also an important mechanism in cloud data sharing system. The user revocation mechanism enables the administrator of the system to revoke some data consumers such that they cannot access or search on the encrypted data anymore. Existing approaches suitable for cloud data sharing incorporate a time interval in every attribute, and issue a new key to every non-revoked data consumers such that revoked data consumers cannot decrypt ciphertext in the new time interval; however, the storage system needs to update all past ciphertext with the new time interval, which is a time-consuming process especially when there is a large amount of ciphertext. Due to the requirements of issuing new keys and updating ciphertext, this kind of user revocation techniques are inefficient and not real-time. Efficient user revocation, preferably an immediate user revocation, would be needed.

[008] Therefore, it is one purpose of the present invention to provide a system that enables any data owner to share data with authorised data consumers through a cloud infrastructure, and in the meantime preventing the cloud service provider from accessing the data.

[009] It is another purpose of the present invention to provide a system that enables the data consumers to perform keyword search over encrypted data to obtain target ciphertext data.

[010] It is another purpose of the present invention to provide a system that supports efficient revocation of data consumers, preferably an immediate revocation of data consumers.

[011] It is another purpose of the present invention to provide a system whereby the cloud service provider module is able to partially decrypt the encrypted data to alleviate the computation burden of data consumers' devices.

[012] It is another purpose of the present invention to provide a system which addresses the above issues including scalable access control, keyword search on encrypted data, and efficient user revocation, simultaneously.

SUMMARY OF THE INVENTION

[013] According to one aspect of the present invention, it provides a cloud data sharing system, including at least one key generation center module, at least one data owner module, at least one cloud service provider module, and at least one data consumer module, wherein

the at least one key generation center module is configured to-

- generate a system public parameter and a master secret key, and distribute the system public parameter to the at least one data owner module, the at least one cloud service provider module, and the at least one data consumer module;
- 5 - generate and distribute an owner query key to each of the at least one data owner module, and an owner search key associated to the owner query key to the cloud service provider module, based on the system public parameter and the master secret key;
- generate and distribute a user query key to each of the at least one data consumer module, and a user search key associated to the user query key to the cloud service provider module,
- 10 based on the system public parameter and the master secret key;
- generate and distribute a decryption key to each of the at least one data consumer module, and a transformation key associated to the decryption key to the cloud service provider module, based on the system public parameter, the master secret key and attribute set of the corresponding data consumer; and
- 15 - generate and distribute revocation information to the at least one cloud service provider module;

the at least one data owner module is configured to-

- generate user-side encrypted data for a data, based on at least one access policy, at least one keyword of the data, the owner query key and the system public parameter received from the
- 20 at least one key generation center module; and
- upload user-side encrypted data to the at least one cloud service provider module;

the at least one cloud service provider module is configured to-

- generate cloud-side encrypted data, based on the user-side encrypted data received from a data owner module, the system public parameter and the corresponding owner search key received from the at least one key generation center module;
- 25 - store the cloud-side encrypted data;
- perform search on the cloud-side encrypted data, based on at least one search request received from a data consumer module and the corresponding user search key received from the at least one key generation module, and retrieve target cloud-side encrypted data found in
- 30 the search;
- transform the target cloud-side encrypted data to transformed ciphertext data, based on the corresponding transformation key received from the at least one key generation center module;
- provide the transformed ciphertext data to the corresponding data consumer module; and
- revoke one or more data consumers based on the revocation information received from the
- 35 key generation center module; and

the at least one data consumer module is configured to-

- generate a search token for a specified keyword, based on the system public parameter and the user query key received from the at least one key generation center module;
- provide at least one search request including at least one search token to the at least one
- 40 cloud service provider module; and

- decrypt the transformed ciphertext data received from the cloud service provider module, based on the system public parameter and the decryption key received from the key generation center module.

5 [014] According to another aspect of the present invention, it provides a method for sharing data using a cloud data sharing system, including at least five processes: initialization process, registration process, data outsourcing process, data access process and revocation process, wherein

the initialization process includes-

- generating and providing a system public parameter, by at least one key generation center module, to at least one data owner module, at least one cloud service provider module and at least one data consumer module; and
- generating a master secret key by the at least one key generation center module, and keeping the master secret key confidential in the at least one key generation center module;

the registration process includes-

- generating and providing at least one owner query key, by the at least one key generation center module, to each of the at least one data owner module, based on the system public parameter and the master secret key;
- generating and providing one owner search key associated to each owner query key, by the at least one key generation center module, to the at least one cloud service provider module, based on the system public parameter and the master secret key;
- generating and providing at least one user query key, by the at least one key generation center module, to each of the at least one data consumer module, based on the system public parameter and the master secret key;
- generating and providing one user search key associated to each user query key, by the at least one key generation center module, to the at least one cloud service provider module, based on the system public parameter and the master secret key;
- generating and providing at least one decryption key, by the key generation center module, to each of the at least one data consumer module, based on the system public parameter, the master secret key and corresponding data consumer's attribute set; and
- generating and providing one transformation key associated to each decryption key, by the key generation center module, to the at least one cloud service provider module, based on the system public parameter, the master secret key and corresponding data consumer's attribute set;

the data outsourcing process includes-

- generating and providing user-side encrypted data, by the at least one data owner module, to the at least one cloud service provider module, based on at least one access policy, at least one keyword of the data to be encrypted, the system public parameter and the at least one owner query key received from the at least one key generation center module; and
- generating and storing cloud-side encrypted data, by the at least one cloud service provider module, based on the user-side encrypted data received from the at least one data owner

module, the system public parameter, and the owner search key associated to each of the at least one owner query key received from the at least one key generation center module;

the data access process includes-

- 5 - generating and providing at least one search request including at least one search token, by the at least one data consumer module, to the at least one cloud service provider module, based on the system public parameter and the at least one user query key received from the at least one key generation center module;
 - 10 - performing search on the cloud-side encrypted data, by the at least one cloud service provider module, based on at least one search request received from a data consumer module and the corresponding user search key received from the at least one key generation module, and retrieving target cloud-side encrypted data found in the search;
 - 15 - transforming the target cloud-side encrypted data to transformed ciphertext data, by the at least one cloud service provider module, based on the transformation key and providing the transformed ciphertext data to the corresponding data consumer module; and
 - 20 - decrypting the transformed ciphertext data, by the corresponding data consumer module, based on the decryption key received from the at least one key generation center module; and
- the revocation process includes-
- 25 - providing revocation information, by the at least one key generation center module, to the at least one cloud service provider module; and
 - 30 - revoking at least one relevant data consumer, by the cloud service provider module, based on the revocation information received from the at least one key generation center module.

[015] According to another aspect of the present invention, the at least one cloud service provider module revokes one or more data consumers by deleting the corresponding user search key/s and/or transformation key/s.

[016] According to another aspect of the present invention, the user-side encrypted data contains ciphertext data and index generation requests.

[017] According to another aspect of the present invention, the at least one cloud service provider module is further configured to generate at least one ciphertext index, based on the index generation requests received from the at least one data owner module, the system public parameter and the corresponding owner search key received from the at least one key generation center module; and wherein the cloud-side encrypted data contains ciphertext index and ciphertext data.

[018] According to another aspect of the present invention, the at least one cloud service provider module transforms the ciphertext data of the target cloud-side encrypted data to transformed ciphertext data only if attribute set associated with the transformation key matches access policy of the ciphertext data of the target cloud-side encrypted data.

[019] According to another aspect of the present invention, the at least one key generation center module includes at least one initialization component, at least one registration component, and at least one revocation component.

5 [020] According to another aspect of the present invention, the at least one data owner module includes at least one local storage component and at least one data outsourcing component.

[021] According to another aspect of the present invention, the at least one data outsourcing component includes at least one encryption component and at least one index generation component.

10

[022] According to another aspect of the present invention, the at least one cloud service provider module includes at least one cloud storage component, at least one search component, and at least one transformation component.

15 [023] According to another aspect of the present invention, the at least one data consumer module includes at least one local storage component and at least one data access component.

[024] According to another aspect of the present invention, the at least one data access component includes at least one search token component and at least one decryption component.

20

BRIEF DESCRIPTION OF THE DRAWINGS

[025] FIG. 1 depicts the structure of a cloud data sharing system according to one embodiment of the present invention.

25

[026] FIG. 2 depicts the structure of a key generation center module in a cloud data sharing system according to one embodiment of the present invention.

[027] FIG. 3 depicts the structure of a data owner module in a cloud data sharing system according to one embodiment of the present invention.

30

[028] FIG. 4 depicts the structure of a cloud service provider module in a cloud data sharing system according to one embodiment of the present invention.

35 [029] FIG. 5 depicts the structure of a data consumer module in a cloud data sharing system according to one embodiment of the present invention.

[030] FIG. 6 depicts an initialization process of a method for sharing data using a cloud data sharing system according to one embodiment of the present invention.

40

[031] FIG. 7 depicts a registration process of a method for sharing data using a cloud data sharing system according to one embodiment of the present invention.

5 [032] FIG. 8 depicts a data outsourcing process of a method for sharing data using a cloud data sharing system according to one embodiment of the present invention.

[033] FIG. 9 depicts a data access process of a method for sharing data using a cloud data sharing system according to one embodiment of the present invention.

10 [034] FIG. 10 depicts a revocation process of a method for sharing data using a secure cloud data sharing system according to one embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

15 [035] While the concepts of the present disclosure are susceptible to various modifications and alternative forms, specific embodiments will herein be described in detail. It should be understood, however, that there is no intent to limit the concepts of the present disclosure to the particular forms disclosed or terms used, but on the contrary, the intention is to cover all modifications, equivalents and alternatives consistent with the present disclosure and the appended claims.

20

[036] In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, that the present invention may be practised without these specific details. Well-known methods, procedures, materials and conditions have not been described in detail so as not to unnecessarily obscure aspects of the implementations.

25

[037] It will be appreciated that the inclusion of a schematic element in a drawing is not meant to imply that such element is required in all embodiments or that the features represented by such element may not be included in or combined with other elements in some embodiments.

30

[038] Several features are described hereafter, each of which may be used independently of one another or with any combination of other features. However, any individual feature may not address any of the problems discussed above or might only address one of the problems discussed above. Some of the problems discussed above might not be fully addressed by any of the features described herein. Although headings are provided, information related to a particular heading, but not found in the section having that heading, may also be found elsewhere in the specification.

35

40

[039] In this specification:

- (1) the terms “contain” and “include” and derivatives of these terms are not to be construed as being restricted or confined to the ensuing matters enumerated but as potentially encompassing other matters not so enumerated;
- (2) words in the singular shall be construed to include the plural and *vice versa*, unless the context requires or indicates otherwise;
- (3) unless defined otherwise, all technical and scientific terms used herein have the same meanings as commonly understood by those skilled in the art to which the present invention belongs.

[040] The cloud data sharing system includes at least one key generation center module, at least one data owner module, at least one cloud service provider module and at least one data consumer module.

[041] A key generation center module is configured to be able to generate information for different purposes and different modules in the system. For example, the key generation center module is able to generate system public parameters, master secret keys, owner query keys, user query keys, owner search keys, user search keys, decryption keys, transformation keys and revocation information for use by different parties or modules.

[042] A data owner is a party who intends to store its data in cloud and share the data with one or more authorised parties through cloud. A data owner has a data owner module configured to be able to specify the access policy in terms of user attributes of data consumers, encrypt the data into user-side encrypted data, and upload the user-side encrypted data to cloud. The data owner module is also configured to be able to generate other information, such as one or more index generation requests.

[043] The data to be encrypted, stored and shared can be any kind of commercial or non-commercial information or document, such as personal medical data, emails, account information, software, etc.

[044] A cloud service provider module is configured to be able to receive user-side encrypted data from the at least one data owner module, encrypt the received user-side encrypted data into cloud-side encrypted data, store the cloud-side encrypted data, perform searches on the cloud-side encrypted data upon search request from any data consumer module, transform the cloud-side encrypted data into transformed ciphertext data and provide the transformed ciphertext data to the authorised data consumer module.

[045] A data consumer is a party or user who wishes to access the data which is encrypted and stored in cloud, under commercial or non-commercial terms subject to the data owner. A party or user may have its information registered in the system to become a registered data consumer in the system.

Each data consumer has a set of user attributes. Each data consumer has a data consumer module configured be able to provide search request to the cloud service provider module to perform searches on the cloud-side encrypted data in the cloud service provider module. A registered data consumer may be authorised to access certain data in the system. An authorised data consumer will be provided transformed ciphertext data and the transformed ciphertext data will be decrypted in the data consumer module.

[046] Through the cloud data sharing system, any data owner is able to share its data to any authorised data consumer without compromising confidentiality of the data, such as leaking information to the cloud service provider or any other unauthorised consumer. Through the cloud data sharing system, any registered data consumer in the system may request for searches on the encrypted data stored in cloud to look for target encrypted data, and only the authorised consumer will be able to access the data. The system will be further discussed in detail below.

[047] FIG.1 depicts a cloud data sharing system according to one embodiment of the present invention. The cloud data sharing system includes a key generation center module **10**, one or more data owner modules **20**, a cloud service provider module **30** and one or more data consumer modules **40**.

[048] FIG. 2 depicts a key generation center module **10** according to one embodiment of the present invention, which includes at least an initialization component **101**, at least a registration component **102**, and at least a revocation component **103**. The key generation center module **10** is configured to generate a system public parameter and a master secret key. Based on the system public parameter and the master secret key, the key generation center module **10** is configured to generate and distribute at least one owner query key to each of the data owner modules **20** and at least one user query key to each of the data consumer modules **40**, and to generate and distribute an owner search key associated to each of the owner query key and a user search key associated to each of the user query key to the cloud service provider module **30**. Preferably, each data owner module has one unique owner query key and each data consumer module has one unique user query key. Based on the system public parameter, the master secret key and the attribute set of each of the data consumers, the key generation center module **10** generates a unique decryption key for each of the data consumers and provides the decryption key to the corresponding data consumer module **40**, generates a transformation key associated to each of the decryption key and provides the transformation key to the cloud service provider module **30**. In the condition where there is a need to revoke one or more data consumers, the key generation center module **10** provides revocation information which specifies identities of the one or more data consumers to be revoked to the cloud service provider module **30**. The cloud service provider module **30** revokes the one or more data consumers by deleting the involved user search key/s and/or transformation key/s based on the received revocation information, which enables immediate user revocation.

[049] FIG. 3 depicts a data owner module **20** according to one embodiment of the present invention, which includes at least a local storage component **201** and at least a data outsourcing component **202**. The data outsourcing component includes at least an encryption component **2021** and an index generation component **2022**. A data owner module **20** is configured to specify the access policy in terms of user attributes of data consumers, and encrypt the data into corresponding ciphertext data. Based on the system public parameter, the data owner module generates index generation requests based on keywords of the data, its owner query key and the system public parameter, and uploads user-side encrypted data containing ciphertext data and index generation requests to the cloud service provider module **30**.

[050] FIG. 4 depicts a cloud service provider module **30** according to one embodiment of the present invention, which includes at least a cloud storage component **301**, at least a search component **302**, and at least a transformation component **303**. After receiving index generation requests from a data owner module **20**, based on system public parameter and corresponding owner search key received from the key generation center module **10**, the cloud service provider module **30** generates ciphertext index. The cloud service provider module **30** further generates and stores cloud-side encrypted data which contains the ciphertext data received from the data owner module **20** and the ciphertext index. The cloud service provider module **30** is configured to perform search on all the cloud-side encrypted data based on search request containing search token received from a data consumer module **40** and the corresponding user search key received from the key generation center module **10**. If the search performed by the cloud service provider module **30** has a positive result and hits the target cloud-side encrypted data, the cloud service provider module **30** transforms the target ciphertext data in the target cloud-side encrypted data based on the transformation key received from the key generation center **10** and provides the transformed ciphertext data to the data consumer module **40**. The target ciphertext data can be successfully transformed if the attribute set associated with the transformation key matches the access policy of the target ciphertext data. Further, the cloud service provider module **30** is configured to revoke involved data consumers based on the revocation information received from the key generation center module **10**.

[051] FIG. 5 depicts a data consumer module **40** according to one embodiment of the present invention, which includes at least a local storage component **401** and at least a data access component **402**. A data access component **402** includes at least a search token component **4021** and at least a decryption component **4022**. Each data consumer module **40** is configured to generate a corresponding search token for a keyword specified by the data consumer, based on the system public parameter and the data consumer's user query key received from the key generation center module **10**. After the search token is generated, the data consumer module **40** provides the search request containing one or more search tokens to the cloud service provider module **30** to perform search on the cloud-side encrypted data. Further, the data consumer module **40** is configured to decrypt the transformed ciphertext data received from the cloud service provider module **30** to obtain the corresponding data for

use, based on the system public parameter and the decryption key received from the key generation center module **10**.

[052] According to one aspect of the present invention, it is provided a data sharing method using the cloud data sharing system, which includes at least five processes: initialization process, registration process, data outsourcing process, data access process, and revocation process. For demonstrating purposes, reference is made to the cloud data sharing system depicted in FIG. 1, in the initialization process, the key generation module **10** generates and provides initiation information to each of the data owner modules **20**, the cloud service provider module **30** and each of the data consumer module **40** to initiate each of the modules. In the registration process, the key generation center module **10** generates and provides information to the each of the data owner modules **20**, the cloud service provider module **30** and each of the data consumer module **40**, to enable each of the modules to take necessary actions respectively for the share of data. In the outsourcing process, each of the data owner modules **20** encrypts its data and upload the user-side encrypted data for the cloud service provider module **30** to further encrypt to cloud-side encrypted data, based on the initiation information and registration information. In the data access process, each of the data consumer module **40** generates one or more search requests containing one or more search tokens for the cloud service provider module **30** to perform search on the cloud-side encrypted data, and the cloud service provider module **30** transforms the target cloud-side encrypted data found in the search to transformed ciphertext data for the authorised data consumer module **40** to decrypt and use. In the revocation process, the cloud service provider module **30** receives revocation information containing identities of one or more data consumers, and delete the involved user search key/s and/or transformation key/s. The method will be further discussed in detail below.

[053] FIG. 6 depicts the initialization process being performed by initialization component **101** of the key generation center module **10** according to one embodiment of the present invention. The initialization component **101** performs the initialization process to specify a security parameter and generate a system public parameter, and provides the system public parameter to the local storage component **201** of the data owner module **20**, the search component **302** and the transformation component **303** of the cloud service provider **30** and the local storage component **401** of the data consumer module **40**. Based on the security parameter, the initialization component **101** of the key generation center module **10** also generates a master secret key which is kept confidential in the key generation center module **10**.

[054] FIG. 7 depicts the registration process being performed by registration component **102** of the key generation center module **10** according to one embodiment of the present invention. The registration component **102** performs the registration process to generate and distribute an owner query key to the index generation component **2022** of the data outsourcing component **202** of the data owner module **20** and the associated owner search key to the search component **302** of the cloud service provider **30** through secure channels, based on the system public parameter and the master secret key

from the initialization component **101**. A secure channel is a way of transferring data that is resistant to overhearing and tampering.

[055] Separately, the registration component **102** performs the registration process to generate and distribute a user query key to the search token component **4021** of the data access component **402** of the data consumer module **40** and the associated user search key to the search component **302** of the cloud service provider module **30** through secure channels based on the system public parameter and the master secret key from the initialization component **101**.

[056] Separately, the registration component **102** of the key generation center module **10** performs the registration process to generate a decryption key for the decryption component **4022** of the data access component **402** of the data consumer module **40** and the associated transformation key for the transformation component **303** of the cloud service provider module **30** through secure channels based on the system public parameter and the master secret key from the initialization component **101** and the attribute set from the data consumer module **40**.

[057] FIG. 8 depicts the data outsourcing process which includes data uploading process and data storage process respectively performed by the data owner module **20** and the cloud service provider module **30**, according to one embodiment of the present invention.

[058] In the data uploading process, the data owner module **20** generates and uploads user-side encrypted data to the cloud service provider module **30**, which contains ciphertext data and corresponding index generation requests. Specifically, the encryption component **2021** of the data outsourcing component **202** of the data owner module **20** retrieves the data and the system public parameter from the local storage component **201** of the data owner module **20**, specifies an access policy in terms of user attributes of data consumers, generates the corresponding ciphertext data based on the system public parameter, and provide the ciphertext data to the search component **302** of the cloud service provider module **30**. Further, the index generation component **2022** of the data outsourcing component **202** of the data owner module **20** retrieves the data and the system public parameter from the local storage component **201** of the data owner module **20**, specifies the corresponding keywords, generates index generation requests for the keywords and provide the index generation requests to the search component **302** of the cloud service provider module **30**.

[059] In the data storage process, the search component **302** of the cloud service provider module **30** receives the ciphertext data from the encryption component **2021** of the data outsourcing component **202** of the data owner module **20** and the corresponding index generation requests from the index generation component **2022** of the data outsourcing component **202** of the data owner module **20**, computes ciphertext indexes based on the system public parameter and the owner search key associated to the owner query key of the data owner module **20**, and stores cloud-side encrypted data

on the cloud storage component **301** which contains the received ciphertext data and the ciphertext indexes.

[060] FIG. 9 depicts the data access process which includes query process, search process, transformation process, and decryption process, performed by the data consumer module **40** and the cloud service provider module **30**, according to one embodiment of the present invention.

[061] In the query process, the search token component **4021** of the data access component **402** of the data consumer module **40** specifies one or more keywords and generates corresponding search tokens based on the system public parameter and the consumer's user query key, sends one or more search requests containing one or more search tokens to the search component **302** of the cloud service provider module **30**.

[062] In the search process, the search component **302** of the cloud service provider module **30** receives one or more search requests from the search token component **4021** of the data access component **402** of the data consumer module **40**, searches all the cloud-side encrypted data on the cloud storage component **301** to obtain the target cloud-side encrypted data based on one or more search tokens in the one or more search requests and the corresponding user search key, provides the target cloud-side encrypted data to the transformation component **303** of the cloud service provider module **30**.

[063] In the transformation process, the transformation component **303** of the cloud service provider module **30** transforms the ciphertext data in the target cloud-side encrypted data, based on the transformation key and the system public parameter, to obtain the corresponding transformed ciphertext data, provides the transformed ciphertext data to the decryption component **4022** of the data access component **402** of the authorised data consumer module **40**.

[064] In the decryption process, the decryption component **4022** of the data access component **402** of the data consumer module **40** decrypts the received transformed ciphertext data, based on the system public parameter and the consumer's decryption key, to get the corresponding data, which is then stored on the corresponding local storage component **401**.

[065] FIG. 10 depicts the revocation process which is performed by the key generation center module **10** and the cloud service provider module **30**, according to one embodiment of the present invention. In the revocation process, the revocation component **103** of the key generation center module **10** provides revocation information which specifies identity information of one or more data consumers to be revoked, to the search component **302** and the transformation component **303** of the cloud service provider module **30**. The search component **302** of the cloud service provider module **30** deletes the involved user search key/s based on the received revocation information. The transformation component **303** of the cloud service provider module **30** deletes the involved transformation keys based on the received

revocation information. The person skilled in the art would appreciate that a data consumer will no longer be able to access the data as long as one of the user search key and transformation key is deleted.

- 5 [066] For illustrating purpose, three examples of algorithms are further discussed below.

Example I

Initialization process

10

[067] The initialization component of the key generation center module chooses a bilinear mapping $e: G_1 \times G_2 \rightarrow G_T$, and a secure symmetric encryption scheme $SE = (K, SE.Enc, SE.Dec)$ where K , $SE.Enc$ and $SE.Dec$ respectively represent the secret key space, the symmetric encryption algorithm and the symmetric decryption algorithm. Then, it randomly chooses $g_1 \in G_1, g_2 \in G_2, \alpha, \beta, x \in Z_p$, where

15 g_1 and g_2 are respectively a generator of groups G_1 and G_2 of order p , and cryptographic hash functions

$$\mathcal{H}_0: \{0,1\}^* \rightarrow Z_p, \mathcal{H}_1: \{0,1\}^* \rightarrow G_1, \mathcal{H}_2: \{0,1\}^* \rightarrow G_2, \mathcal{H}_3: \{0,1\}^* \rightarrow G_T, \text{ and } \mathcal{H}_k: \{0,1\}^* \rightarrow K.$$

[068] Next, the initialization component of the key generation center sets

$$PP = \{(G_1, G_2, G_T, e), \mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3, \mathcal{H}_k, SE, g_1, g_2, h_1, h_2\}$$

20 as the public parameter and $MSK = \{\beta, x, h_3\}$ as the master secret key, where $h_1 = g_1^\beta, h_2 = e(g_1, g_2)^\alpha, h_3 = g_2^\alpha$.

[069] At last, the initialization component of the key generation center module secretly keeps MSK and provides PP to the local storage components of data owners, the local storage components of data consumer modules and the search component and the transformation component of the cloud service provider.

25

Registration process

30 [070] For the registration of a data owner, the registration component of the key generation center module takes as inputs the system public parameter PP and the master secret key MSK , randomly chooses $x_{DO} \in Z_p$, sets the data owner's owner query key as $qk_{DO} = x_{DO}$ and the corresponding owner search key as $shk_{DO} = g_2^{\frac{x}{x_{DO}}}$. Then, the registration component of the key generation center chooses a data owner identity $u_{DO} \in \{0,1\}^*$ and provides $\{u_{DO}, qk_{DO}\}$ and $\{u_{DO}, shk_{DO}\}$ to the index generation component of the data outsourcing component of the data owner module and the search component of the cloud service provider module through secure channels, respectively.

35

[071] For the registration of a data consumer, the registration component of the key generation center module takes as inputs the system public parameter PP and the master secret key MSK , randomly

chooses $x_{DC} \in Z_P$, sets the data consumer's user query key as $qk_{DC} = x_{DC}$ and the corresponding user search key as $shk_{DC} = g_2^{\frac{x}{x_{DC}}}$. Then, the registration component of the key generation center module chooses a data consumer identity $u_{DC} \in \{0,1\}^*$ and provides $\{u_{DC}, qk_{DC}\}$ and $\{u_{DC}, shk_{DC}\}$ to the search token component of the data access component of the data consumer module and the search component of the cloud service provider module through secure channels, respectively. In addition, the registration component of the key generation center module takes as inputs the system public parameter PP and the master secret key MSK , and the data consumer's attribute set S , which is from the data consumer module, randomly chooses $\gamma, z, \{\gamma_j\}_{j \in S} \in Z_P$, and sets the decryption key dk_{DC} of the data consumer and the corresponding transformation key tk_{DC} as $dk_{DC} = z, tk_{DC} = \{S, D, \{D_j, D'_j\}_{j \in S}\}$, where

$$D = g_2^{\frac{\alpha+\gamma}{\beta z}}, D_j = g_2^{\frac{\gamma}{z}} \cdot \mathcal{H}_2(j)^{\frac{\gamma_j}{z}}, D'_j = g_1^{\frac{\gamma_j}{z}}.$$

[072] Finally, the registration component of the key generation center module provides dk_{DC} and $\{u_{DC}, tk_{DC}\}$ to the decryption component of the data access component of the data consumer module and the transformation component of the cloud service provider module, respectively.

Data outsourcing process

[073] The data outsourcing process includes data uploading process and data storage process, which are separately discussed below.

[074] **Data uploading:** When a data owner with identity u_{DO} intends to outsource his/her data $Data$ with a set of keywords KWS , the encryption component of the data outsourcing component of the data owner module does the following:

25

[075] The encryption component of the data outsourcing component of the data owner module specifies an access policy T which is a tree. Each node of the tree has a unique index i and the index of the root node is denoted as i_{root} . The parent node of node i is denoted as $parent(i)$. Each leaf node has a threshold value 1 and it is associated with an attribute. Each non-leaf node i is a threshold node with a corresponding natural number pair $(k_{nl,i}, n_{nl,i})$, where $n_{nl,i}$ represents the number of child nodes of the non-leaf node i and $k_{nl,i} \leq n_{nl,i}$ is the threshold value. Especially, the non-leaf node becomes an "AND" node if $k_{nl,i} = n_{nl,i}$ and an "OR" node if $k_{nl,i} = 1$. In the subsequent procedures, a polynomial of degree $k_{nl,i} - 1$ will be chosen for each node.

[076] The encryption component of the data outsourcing component of the data owner module takes as inputs the system public parameter PP , the access policy T , randomly chooses $r \in Z_P, m \in G_T$, and computes $s = \mathcal{H}_0(r, m)$. Then it chooses a polynomial $q_i(x)$ of degree d_i for every node i of T , where

35

$$q_i(x) = \begin{cases} \sum_{l \in [1, d_{i_{root}}]} a_{i_{root}, l} x^l + s, & \text{if } i = i_{root}, \\ \sum_{l \in [1, d_i]} a_{i, l} x^l + q_{parent(i)}(i), & \text{if } i \neq i_{root} \text{ and is a non-leaf node}, \\ q_{parent(i)}(i), & \text{if } i \text{ is a leaf node} \end{cases}$$

with the coefficients $\{a_{i, l} \in Z_p\}_{l \in [1, d_i]}$ being randomly chosen. Note that $d_i = k_{nl, i} - 1$ if node i is a non-leaf node and $d_i = 0$ if node i is a leaf node. Suppose Y is the set of leaf node indexes of T and the function $attr(y)$ is to get the attribute of the node $y \in Y$. Then, it sets $CT_{ABE} =$

$$5 \quad \left\{ T, C_0, C_1, C_2, \{C_{3, y}, C'_{3, y}\}_{y \in Y} \right\} \text{ and } k_{SE} = \mathcal{H}_k(m), \text{ where } C_0 = m \oplus \mathcal{H}_3(r), C_1 = r \cdot h_2^s = r \cdot e(g_1, g_2)^{as}, \\ C_2 = h_1^s = g_1^{\beta s}, C_{3, y} = g_1^{q_y(0)}, C'_{3, y} = \mathcal{H}_2(attr(y))^{q_y(0)}.$$

[077] The encryption component of the data outsourcing component of the data owner module takes as inputs the system public parameter PP , the symmetric encryption key k_{SE} and the data $Data$, and generates a symmetric ciphertext $CT_{SE} = SE.Enc(k_{SE}, Data)$.

[078] For each keyword $kw_i \in KWS$, the index generation component of the data outsourcing component of the data owner module takes as inputs the system public parameter PP , the owner query key $qk_{DO} = x_{DO}$, and kw_i , computes $CT_{kw_i} = \mathcal{H}_1(kw_i)^{x_{DO}}$. It sets $CT_{KWS} = \{CT_{kw_i}\}_{kw_i \in KWS}$ which is used as the index generation request.

[079] The data owner module sets the user-side encrypted data as $CT_{DO} = \{u_{DO}, CT_{Data}, CT_{KWS}\}$, where $CT_{Data} = \{CT_{ABE}, CT_{SE}\}$, and provides CT_{DO} to the cloud service provider module. Specifically, the encryption component of the data outsourcing component of the data owner module provides CT_{Data} to the search component of the cloud service provider module, and the index generation component of the data outsourcing component of the data owner module provides $\{u_{DO}, CT_{KWS}\}$ to the search component of the cloud service provider module.

[080] **Data storage:** Upon receiving the user-side encrypted data $CT_{DO} = \{u_{DO}, CT_{Data}, CT_{KWS}\}$, the search component of the cloud service provider module retrieves the owner search key shk_{DO} associated with u_{DO} , takes as inputs the system public parameter PP , the owner search key shk_{DO} , and CT_{KWS} , and does the following:

For each CT_{kw_i} in CT_{KWS} , it randomly chooses $y_i \in Z_p$, and computes

$$30 \quad k_{SE, kw_i} = \mathcal{H}_k(e(CT_{kw_i}, shk_{DO})) = \mathcal{H}_k\left(e\left(\mathcal{H}_1(kw_i)^{x_{DO}}, g_2^{\frac{x}{x_{DO}}}\right)\right) \\ = \mathcal{H}_k(e(\mathcal{H}_1(kw_i), g_2)^x), \\ I_{kw_i} = \{y_i, C_{y_i} = SE.Enc(k_{SE, kw_i}, y_i)\},$$

where $SE.Enc$ also can be replaced by a hash function to get an alternative embodiment.

Then, it sets $I_{KWS} = \{I_{kw_i}\}_{CT_{kw_i} \in CT_{KWS}}$. The search component of the cloud service provider module provides $\{CT_{Data}, I_{KWS}\}$ to the cloud storage component of the cloud service provider.

[081] The cloud storage component of the cloud service provider module sets the cloud-side encrypted data as $CT_{Cloud} = \emptyset$ at first. Upon receiving $\{CT_{Data}, I_{KWS}\}$, it sets $CT_{Cloud} = CT_{Cloud} \cup \{CT_{Data}, I_{KWS}\}$.

Data access process

[082] The data access process includes query process, search process, transformation process, and decryption process, which are separately discussed below.

[083] **Query:** When a data consumer with identity u_{DC} intends to get the data with a keyword kw , the search token component of the data access component of the data consumer module takes as inputs the system public parameter PP , the user query key $qk_{DC} = x_{DC}$, and kw , computes the corresponding search token $ToK_{kw} = \mathcal{H}_1(kw)^{x_{DC}}$, and provides $\{u_{DC}, ToK_{kw}\}$ to the search component of the cloud service provider module.

[084] **Search:** Upon receiving $\{u_{DC}, ToK_{kw}\}$, the search component of the cloud service provider module takes as inputs the system public parameter PP , and the user search key shk_{DC} associated with u_{DC} , and sets the target ciphertext data $TarCT_{Data} = \emptyset$ at first. Then, it does the following:

The search component of the cloud service provider computes

$$k'_{SE, kw} = \mathcal{H}_k(e(ToK_{kw}, shk_{DC})) = \mathcal{H}_k\left(e\left(\mathcal{H}_1(kw)^{x_{DC}}, g_2^{\frac{x}{x_{DC}}}\right)\right) = \mathcal{H}_k(e(\mathcal{H}_1(kw), g_2)^x).$$

For each item $\{CT_{Data}, I_{KWS}\} \in CT_{Cloud}$ from the cloud storage component of the cloud service provider module, and for each $I_{kw_i} \in I_{KWS}$, it parses $I_{kw_i} = \{y_i, C_{y_i}\}$, computes $C'_{y_i} = SE.Enc(k'_{SE, kw}, y_i)$, and sets $TarCT_{Data} = TarCT_{Data} \cup CT_{Data}$ if and only if $C'_{y_i} = C_{y_i}$.

[085] Finally, the search component of the cloud service provider module provides $\{u_{DC}, TarCT_{Data}\}$ to the transformation component of the cloud service provider module.

[086] **Transformation:** The transformation component of the cloud service provider module sets the transformed ciphertext data as $TranCT_{Data} = \emptyset$. If and only if $TarCT_{Data} \neq \emptyset$, for each $CT_{Data} \in TarCT_{Data}$, it takes as inputs the system public parameter PP , $CT_{Data} = \{CT_{ABE}, CT_{SE}\}$, the transformation key $tk_{DC} = \{S, D, \{D_j, D'_j\}_{j \in S}\}$ associated with the data consumer u_{DC} , and parses $CT_{ABE} = \{T, C_0, C_1, C_2, \{C_{3,y}, C'_{3,y}\}_{y \in Y}\}$. If and only if S satisfies T , for each node i in the access policy T , it does the following:

1) If node i is a leaf node, then let $t = attr(i)$, and does the following:

- a) If $t \in S$, where S is the data consumer's attribute set and it is given in tk_{DC} , computes

$$F_i = \frac{e(C_{3,i}, D_t)}{e(D'_t, C'_{3,i})} = \frac{e\left(g_1^{q_i(0)}, g_2^{\frac{\gamma}{z} \cdot \mathcal{H}_2(t) \frac{\gamma t}{z}}\right)}{e\left(g_1^{\frac{\gamma t}{z}}, \mathcal{H}_2(t) q_i(0)\right)} = e(g_1, g_2)^{\frac{\gamma q_i(0)}{z}}.$$

- b) If $t \notin S$, it sets $F_i = \perp$.

- 2) If i is a non-leaf node, then does the following:

- 5 a) For each child node j of i , it sets $t = attr(j)$, if $t \notin S$, it sets $F_j = \perp$. Otherwise, it computes

$$F_j = \frac{e(C_{3,j}, D_t)}{e(D'_t, C'_{3,j})} = \frac{e\left(g_1^{q_j(0)}, g_2^{\frac{\gamma}{z} \cdot \mathcal{H}_2(t) \frac{\gamma t}{z}}\right)}{e\left(g_1^{\frac{\gamma t}{z}}, \mathcal{H}_2(t) q_j(0)\right)} = e(g_1, g_2)^{\frac{\gamma q_j(0)}{z}}.$$

- b) It sets $S_i = \{j \mid F_j \neq \perp\}$ where j is child node of i .

- c) If the size of S_i is less than $d_i + 1$, then it returns $F_i = \perp$. Otherwise, suppose S_i has $d_i + 1$ elements and compute

$$\begin{aligned} 10 \quad F_i &= \prod_{j \in S_i} F_j^{\Delta_{j,S_i}(0)} \\ &= \prod_{j \in S_i} \left(e(g_1, g_2)^{\frac{\gamma q_j(0)}{z}} \right)^{\Delta_{j,S_i}(0)} \\ &= \prod_{j \in S_i} e(g_1, g_2)^{\frac{\gamma q_j(0) \Delta_{j,S_i}(0)}{z}} \\ &= e(g_1, g_2)^{\frac{\gamma}{z} \sum_{j \in S_i} q_i(j) \Delta_{j,S_i}(0)} \\ &= e(g_1, g_2)^{\frac{\gamma q_i(0)}{z}}, \end{aligned}$$

- 15 where $\Delta_{j,S_i}(0) = \prod_{a \in S_i, a \neq j} \frac{0-a}{j-a}$. For the root node i_{root} , $F_{i_{root}} = e(g_1, g_2)^{\frac{\gamma s}{z}}$ or $\perp$ since $q_{i_{root}}(0) = s$. Let $B = F_{i_{root}}$ and if and only if $B \neq \perp$, it computes

$$C'_2 = \frac{e(C_2, D)}{B} = \frac{e(g_1^{\beta s}, g_2^{\frac{\alpha + \gamma}{\beta z}})}{e(g_1, g_2)^{\frac{\gamma s}{z}}} = e(g_1, g_2)^{\frac{\alpha s}{z}},$$

and sets $TranCT_{Data} = TranCT_{Data} \cup \{C_0, C_1, C'_2, CT_{SE}\}$.

- 20 [087] Finally, the transformation component of the cloud service provider module provides $TranCT_{Data}$ to the decryption component of the data access component of the data consumer module.

[088] **Decryption:** Upon receiving $TranCT_{Data}$, the decryption component of the data access component of the data consumer module does the following:

- 25 If $TranCT_{Data} = \emptyset$, it knows no suitable data exist in the cloud service provider. Otherwise, $TranCT_{Data} \neq \emptyset$, it takes as inputs the system public parameter PP , its decryption key dk_{DC} and $TranCT_{Data}$, and does the following:

For each tuple $\{C_0, C_1, C'_2, CT_{SE}\} \in TranCT_{Data}$, it computes

$$r_d = \frac{C_1}{C'_2{}^{dk_{DC}}} = \frac{r \cdot e(g_1, g_2)^{\alpha s}}{\left(e(g_1, g_2)^{\frac{\alpha s}{z}} \right)}, m_d = C_0 \oplus \mathcal{H}_3(r_d), s_d = \mathcal{H}_0(r_d, m_d).$$

[089] Then, it checks if $C_1 = r_d \cdot h_2^{s_d}$ and $C'_2 = h_2^{\frac{s_d}{z}}$. If no, it fails to decrypt. Otherwise, it computes the symmetric key $k_{SE} = \mathcal{H}_k(m_d)$ and recovers the data $Data = SE.Dec(k_{SE}, CT_{SE})$. Finally, the decryption component of the data access component of the data consumer module puts all the recovered data into the local storage component of the data consumer module.

Revocation process

[090] To revoke a data consumer u_{DC} from the system, the revocation component of the key generation center module provides u_{DC} as the revocation request to the cloud service provider module. Then, the search component of the cloud service provider deletes the data consumer's user search key shk_{DC} , and/or the transformation component of the cloud service provider module deletes the consumer's transformation key tk_{DC} .

Performance analysis

[091] **Computation cost of the key generation center module:** In the process of initialization, the key generation center module needs to perform a bilinear pairing operation $h_2 = e(g_1, g_2)^\alpha$ and two exponentiation operations $h_1 = g_1^\beta, h_3 = g_2^\alpha$. In the process of registration, the key generation center module needs to perform one exponentiation operation to generate a search key. For example, $shk_{DC} = g_2^{\frac{x}{x_{DC}}}$ is computed for a user search key associated with the data consumer u_{DC} . In addition, to generate a transformation key $tk_{DC} = \{S, D, \{D_j, D'_j\}_{j \in S}\}$, where

$$D = g_2^{\frac{\alpha+y}{\beta z}}, D_j = g_2^{\frac{y}{z}} \cdot \mathcal{H}_2(j)^{\frac{r_j}{z}}, D'_j = g_1^{\frac{r_j}{z}},$$

$2n + 2$ exponentiation operations are required where n is the size of the data consumer's attribute set.

25

[092] **Computation cost of the data owner module:** In the process of data uploading, the data owner module computes the user-side encrypted data as $CT_{DO} = \{u_{DO}, CT_{Data}, CT_{KWS}\}$, where $CT_{Data} = \{CT_{ABE}, CT_{SE}\}$, $CT_{ABE} = \{T, C_0, C_1, C_2, \{C_{3,y}, C'_{3,y}\}_{y \in Y}\}$, $CT_{SE} = SE.Enc(k_{SE}, Data)$, $CT_{KWS} = \{CT_{kw_i}\}_{kw_i \in KWS}$, and $C_0 = m \oplus \mathcal{H}_3(r), C_1 = r \cdot h_2^s = r \cdot e(g_1, g_2)^{\alpha s}$, $C_2 = h_1^s = g_1^{\beta s}, C_{3,y} = g_1^{q_y(0)}, C'_{3,y} = \mathcal{H}_2(attr(y))^{q_y(0)}, k_{SE} = \mathcal{H}_k(m), CT_{kw_i} = \mathcal{H}_1(kw_i)^{x_{DO}}$.

30

[093] Accordingly, $2n_{leaf} + n_{kw} + 2$ exponentiation operations are required where n_{leaf} is the number of leaf nodes in the access policy tree T , and n_{kw} is the number of keywords in the corresponding data.

35

[094] **Computation cost of the cloud service provider module:** In the process of data outsourcing, for each CT_{kw_i} in CT_{KWS} , the cloud service provider module needs to compute the cloud-side encrypted data by computing

$$k_{SE,kw_i} = \mathcal{H}_k \left(e(CT_{kw_i}, shk_{DO}) \right), I_{kw_i} = \{y_i, C_{y_i} = SE.Enc(k_{SE,kw_i}, y_i)\}.$$

- 5 Therefore, n_{kw} bilinear pairing operations are required for the data storage where n_{kw} represents the number of keywords in the data.

- [095] In the process of data access, the cloud service provider module is involved in search and transformation. In the search process, it needs to compute a bilinear pairing operation, i.e., $k'_{SE,kw} = \mathcal{H}_k(e(ToK_{kw}, shk_{DC}))$. Furthermore, for each item $\{CT_{Data}, I_{KWS}\} \in CT_{Cloud}$ and each $I_{kw_i} \in I_{KWS}$, it performs the matching based on a symmetric encryption (or hashing). Therefore, N_{index} symmetric encryption (or hashing) operations are required for each search where N_{index} represents the total number of ciphertext indexes in the cloud. In the transformation process, it needs to compute $F_i = \frac{e(C_{3,i}, D_t)}{e(D'_t, C'_{3,i})}$ for each leaf node i in T with $attr(i)$ being an attribute of the data consumer. Accordingly,
- 15 $2n_{leaf}$ bilinear pairing operations are required for a ciphertext transformation where n_{leaf} is the number of leaf nodes in the access policy tree T .

- [096] **Computation cost of the data consumer module:** In the process of data access, the data consumer module first generates a search token $ToK_{kw} = \mathcal{H}_1(kw)^{x_{DC}}$, and hence only one exponentiation operation is required. In addition, the data consumer module needs to perform decryption by computing

$$r_d = \frac{C_1}{C'_2{}^{dk_{DC}}}, m_d = C_0 \oplus \mathcal{H}_3(r_d), s_d = \mathcal{H}_0(r_d, m_d),$$

and checking if $C_1 = r_d \cdot h_2^{s_d}$ and $C'_2 = h_2^{\frac{s_d}{z}}$. Therefore, three exponentiation operations are required for decryption.

25

Security analysis

- [097] **Data confidentiality:** In the process of data outsourcing, the data owner module first chooses a symmetric key to encrypt the data, and then encrypts the symmetric key based an access policy. As for the encryption of the symmetric key, the data owner module chooses a polynomial $q_i(x)$ of degree d_i for every node i of T , and the polynomial associated with the root node is $q_{i_{root}}(x)$. The value of $s = q_{i_{root}}(0)$ is used in the data encryption by computing $h_2^s = r \cdot e(g_1, g_2)^{\alpha s}$. In this case, only the data consumer whose attribute set satisfies the access policy tree of the ciphertext data can succeed in decryption. In other words, only authorized data consumers are able to decrypt the ciphertext data.

35

- [098] **Keyword privacy:** In the data uploading process of the data outsourcing process, the data owner module generates index generation requests. For each $kw_i \in KWS$, the data owner module

computes $CT_{kw_i} = \mathcal{H}_1(kw_i)^{x_{DO}}$ as the corresponding index generation request. Because x_{DO} is only known by the data owner, under the hardness assumption of the discrete logarithm problem and the collision-resistance of hash functions, the privacy of the keyword kw_i is preserved. In the process of data access, the data consumer module provides the search token $ToK_{kw} = \mathcal{H}_1(kw)^{x_{DC}}$ to the cloud service provider module. Similarly, the privacy of kw is preserved.

[099] **Correctness of keyword search:** According to the system design, for a keyword kw_i , the corresponding index is $I_{kw_i} = \{y_i, C_{y_i} = SE.Enc(k_{SE,kw_i}, y_i)\}$, where $y_i \in Z_p$, $k_{SE,kw_i} = \mathcal{H}_k(e(CT_{kw_i}, shk_{DO})) = \mathcal{H}_k(e(\mathcal{H}_1(kw_i), g_2)^x)$. In the process of search, the cloud service provider module can compute $k'_{SE,kw} = \mathcal{H}_k(e(ToK_{kw}, shk_{DC})) = \mathcal{H}_k(e(\mathcal{H}_1(kw), g_2)^x)$. Obviously, if and only if $kw_i = kw$, then $k_{SE,kw_i} = k'_{SE,kw}$ which is completely determined by the keyword and the master secret key of the key generation center. Therefore, the correctness of keyword search is ensured. Those skilled in the art would appreciate that any registered data consumer in the system is able to search over encrypted data provided by any data owner.

[100] **User revocation:** In the system of the present invention, the cloud service provider module performs searches over encrypted data based on the user search key corresponding to the data consumer. Furthermore, in the process of transformation, the cloud service provider module performs the ciphertext transformation based on the transformation key corresponding to the data consumer. Accordingly, either the user search key or the transformation key corresponding to the data consumer is deleted, the data consumer is revoked from the system. Therefore, the user revocation is enabled in the present invention, and can be realised in an immediate effect.

Example II

[101] In practical applications, the cloud service provider module usually provides services to a large number of users. This example II aims to further improve the efficiency of data access process, especially the efficiency of transformation at the cloud service provider module.

Initialization process

[102] The initialization component of the key generation center module chooses a bilinear mapping $e: G_1 \times G_2 \rightarrow G_T$, and a secure symmetric encryption scheme $SE = (K, SE.Enc, SE.Dec)$ where K , $SE.Enc$ and $SE.Dec$ respectively represent the secret key space, the symmetric encryption algorithm and the symmetric decryption algorithm. Then, it randomly chooses $g_1 \in G_1, g_2 \in G_2, \alpha, \beta, x \in Z_p$, where g_1 and g_2 are respectively a generator of groups G_1 and G_2 of order p , and cryptographic hash functions

$$\mathcal{H}_0: \{0,1\}^* \rightarrow Z_p, \mathcal{H}_1: \{0,1\}^* \rightarrow G_1, \mathcal{H}_t: \{0,1\}^* \rightarrow G_T, \text{ and } \mathcal{H}_k: \{0,1\}^* \rightarrow K.$$

[103] Next, the initialization component of the key generation center module sets

$$PP = \{(G_1, G_2, G_T, e), \mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_t, \mathcal{H}_k, SE, g_1, g_2, h_1, h_2, h_3\}$$

as the public parameter and $MSK = \{\alpha, x\}$ as the master secret key, where $h_1 = g_1^\beta, h_2 = g_2^\beta, h_3 = e(g_1, g_2)^\alpha$.

5 [104] At last, the initialization component of the key generation center module secretly keeps MSK and provides PP to the local storage components of data owner modules, the local storage components of data consumer modules and the search component and the transformation component of the cloud service provider module.

10 Registration process

[105] For the registration of a data owner, the registration component of the key generation center module takes as inputs the system public parameter PP and the master secret key MSK , randomly chooses $x_{DO} \in Z_P$, sets the data owner's owner query key as $qk_{DO} = x_{DO}$ and the corresponding owner

15 search key as $shk_{DO} = g_2^{\frac{x}{x_{DO}}}$. Then, the registration component of the key generation center module chooses an identity $u_{DO} \in \{0,1\}^*$ and provides $\{u_{DO}, qk_{DO}\}$ and $\{u_{DO}, shk_{DO}\}$ to the index generation component of the data outsourcing component of the data owner module and the search component of the cloud service provider module through secure channels, respectively.

20 [106] For the registration of a data consumer, the registration component of the key generation center module takes as inputs the system public parameter PP and the master secret key MSK , randomly chooses $x_{DC} \in Z_P$, sets the data consumer's user query key as $qk_{DC} = x_{DC}$ and the corresponding user

search key as $shk_{DC} = g_2^{\frac{x}{x_{DC}}}$. Then, the registration component of the key generation center module chooses an identity $u_{DC} \in \{0,1\}^*$ and provides $\{u_{DC}, qk_{DC}\}$ and $\{u_{DC}, shk_{DC}\}$ to the search token component of the data access component of the data consumer module and the search component of the cloud service provider module through secure channels, respectively. In addition, the registration component of the key generation center module takes as inputs the system public parameter PP and the master secret key MSK , and the data consumer's attribute set S , which is from the data consumer module, randomly chooses $v, z \in Z_P$, and sets the decryption key dk_{DC} of the data consumer and the corresponding transformation key tk_{DC} as $dk_{DC} = \{S, D_1, D_2, \{D_j\}_{j \in S}\}$, $tk_{DC} = \{S, z\}$, where

$$D_1 = g_2^{\frac{\alpha}{z} + \beta v}, D_2 = g_2^{vz}, D_j = \mathcal{H}_1(j)^{vz}.$$

[107] Finally, the registration component of the key generation center module provides dk_{DC} and $\{u_{DC}, tk_{DC}\}$ to the decryption component of the data access component of the data consumer module and the transformation component of the cloud service provider module, respectively.

Data outsourcing process

[108] The data outsourcing process includes data uploading process and data storage process, which are separately discussed below.

5

[109] **Data uploading:** When a data owner with identity u_{DO} intends to outsource its data $Data$ with a set of keywords KWS , the encryption component of the data outsourcing component of the data owner module does the following:

10 [110] The encryption component of the data outsourcing component of the data owner module specifies an access policy T which is a tree. Each node has a unique index i and the index of the root node is denoted as i_{root} . The parent node of node i is denoted as $parent(i)$. Each leaf node has a threshold value 1 and it is associated with an attribute. Each non-leaf node i is a threshold node with a corresponding natural number pair $(k_{nl,i}, n_{nl,i})$, where $n_{nl,i}$ represents the number of child nodes of the non-leaf node i and $k_{nl,i} \leq n_{nl,i}$ is the threshold value. Especially, the non-leaf node becomes an "AND" node if $k_{nl,i} = n_{nl,i}$ and an "OR" node if $k_{nl,i} = 1$. In the subsequent procedures, a polynomial of degree $k_{nl,i} - 1$ will be chosen for each node.

20 [111] The encryption component of the data outsourcing component of the data owner module takes as inputs the system public parameter PP , the access policy T , randomly chooses $r \in Z_p, m \in G_T$, and computes $s = \mathcal{H}_0(r, m)$. Then it chooses a polynomial $q_i(x)$ of degree d_i for every node i of T , where

$$q_i(x) = \begin{cases} \sum_{l \in [1, d_{i_{root}}]} a_{i_{root}, l} x^l + s, & \text{if } i = i_{root}, \\ \sum_{l \in [1, d_i]} a_{i, l} x^l + q_{parent(i)}(i), & \text{if } i \neq i_{root} \text{ and is a non-leaf node}, \\ q_{parent(i)}(i), & \text{if } i \text{ is a leaf node} \end{cases}$$

with the coefficients $\{a_{i, l} \in Z_p\}_{l \in [1, d_i]}$ being randomly chosen. Note that $d_i = k_{nl,i} - 1$ if node i is a non-leaf node and $d_i = 0$ if node i is a leaf node. Suppose Y is the set of leaf node indexes of T and the function $attr(y)$ is to get the attribute of the node $y \in Y$. Then, it sets $CT_{ABE} =$

$$\left\{ T, C_0, C_1, C_2, \{C_{3,y}, C'_{3,y}\}_{y \in Y} \right\} \text{ and } k_{SE} = \mathcal{H}_k(m), \text{ where } C_0 = m \oplus \mathcal{H}_t(r), C_1 = r \cdot h_3^s = r \cdot e(g_1, g_2)^{as}, C_2 = g_1^s, C_{3,y} = g_2^{r_y}, C'_{3,y} = g_1^{\beta q_y(0)} \mathcal{H}_1(attr(y))^{r_y}.$$

30 [112] The encryption component of the data outsourcing component of the data owner module takes as inputs the system public parameter PP , the symmetric encryption key k_{SE} and the data $Data$, and generates a symmetric ciphertext $CT_{SE} = SE.Enc(k_{SE}, Data)$.

[113] For each keyword $kw_i \in KWS$, the index generation component of the data outsourcing component of the data owner module takes as inputs the system public parameter PP , the owner query

key $qk_{DO} = x_{DO}$, and kw_i , computes $CT_{kw_i} = \mathcal{H}_1(kw_i)^{x_{DO}}$. It sets $CT_{KWS} = \{CT_{kw_i}\}_{kw_i \in KWS}$ which is used as the index generation request.

[114] The data owner module sets the user-side encrypted data as $CT_{DO} = \{u_{DO}, CT_{Data}, CT_{KWS}\}$, where $CT_{Data} = \{CT_{ABE}, CT_{SE}\}$, and provides CT_{DO} to the cloud service provider module. Specifically, the encryption component of the data outsourcing component of the data owner module provides CT_{Data} to the search component of the cloud service provider module, and the index generation component of the data outsourcing component of the data owner module provides $\{u_{DO}, CT_{KWS}\}$ to the search component of the cloud service provider module.

10

[115] **Data storage:** Upon receiving the user-side encrypted data $CT_{DO} = \{u_{DO}, CT_{Data}, CT_{KWS}\}$, the search component of the cloud service provider module retrieves the owner search key shk_{DO} associated with u_{DO} , takes as inputs the system public parameter PP , the owner search key shk_{DO} , and CT_{KWS} , and does the following:

15 For each CT_{kw_i} in CT_{KWS} , it randomly chooses $y_i \in Z_p$, and computes

$$\begin{aligned} k_{SE, kw_i} &= \mathcal{H}_k \left(e(CT_{kw_i}, shk_{DO}) \right) = \mathcal{H}_k \left(e \left(\mathcal{H}_1(kw_i)^{x_{DO}}, g_2^{\frac{x}{x_{DO}}} \right) \right) \\ &= \mathcal{H}_k(e(\mathcal{H}_1(kw_i), g_2)^x), \\ I_{kw_i} &= \{y_i, C_{y_i} = SE.Enc(k_{SE, kw_i}, y_i)\}, \end{aligned}$$

where $SE.Enc$ also can be replaced by a hash function to get an alternative embodiment.

20 Then, it sets $I_{KWS} = \{I_{kw_i}\}_{CT_{kw_i} \in CT_{KWS}}$. The search component of the cloud service provider module provides $\{CT_{Data}, I_{KWS}\}$ to the cloud storage component of the cloud service provider module.

[116] The cloud storage component of the cloud service provider module sets the cloud-side encrypted data as $CT_{Cloud} = \emptyset$ at first. Upon receiving $\{CT_{Data}, I_{KWS}\}$, it sets $CT_{Cloud} = CT_{Cloud} \cup \{CT_{Data}, I_{KWS}\}$.

25

Data access

[117] The data access process includes query process, search process, transformation process, and decryption process, which are separately discussed below.

30

[118] **Query:** When a data consumer with identity u_{DC} intends to get the data with a keyword kw , the search token component of the data access component of the data consumer module takes as inputs the system public parameter PP , the user query key $qk_{DC} = x_{DC}$, and kw , computes the corresponding search token $ToK_{kw} = \mathcal{H}_1(kw)^{x_{DC}}$, and provides $\{u_{DC}, ToK_{kw}\}$ to the search component of the cloud service provider module.

35

[119] **Search:** Upon receiving $\{u_{DC}, ToK_{kw}\}$, the search component of the cloud service provider module takes as inputs the system public parameter PP , and the user search key shk_{DC} associated with u_{DC} , and sets the target ciphertext data $TarCT_{Data} = \emptyset$ at first. Then, it does the following:

The search component of the cloud service provider computes

$$5 \quad k'_{SE,kw} = \mathcal{H}_k(e(ToK_{kw}, shk_{DC})) = \mathcal{H}_k\left(e\left(\mathcal{H}_1(kw)^{x_{DC}}, g_2^{\frac{x}{x_{DC}}}\right)\right) = \mathcal{H}_k(e(\mathcal{H}_1(kw), g_2)^x).$$

For each item $\{CT_{Data}, I_{KWS}\} \in CT_{Cloud}$ from the cloud storage component of the cloud service provider module, and for each $I_{kw_i} \in I_{KWS}$, it parses $I_{kw_i} = \{y_i, C_{y_i}\}$, computes $C'_{y_i} = SE.Enc(k'_{SE,kw}, y_i)$, and sets $TarCT_{Data} = TarCT_{Data} \cup CT_{Data}$ if and only if $C'_{y_i} = C_{y_i}$.

10 [120] Finally, the search component of the cloud service provider module provides $\{u_{DC}, TarCT_{Data}\}$ to the transformation component of the cloud service provider module.

[121] **Transformation:** The transformation component of the cloud service provider module sets the transformed ciphertext data as $TranCT_{Data} = \emptyset$. If and only if $TarCT_{Data} \neq \emptyset$, for each $CT_{Data} \in TarCT_{Data}$, it takes as inputs the system public parameter PP , $CT_{Data} = \{CT_{ABE}, CT_{SE}\}$, the transformation key $tk_{DC} = \{S, z\}$ associated with the data consumer u_{DC} , and parses $CT_{ABE} = \{T, C_0, C_1, C_2, \{C_{3,y}, C'_{3,y}\}_{y \in Y}\}$. If and only if S satisfies T , it sets $C'_2 = C_2^z$, and $TranCT_{Data} = TranCT_{Data} \cup \{T, C_0, C_1, C'_2, \{C_{3,y}, C'_{3,y}\}_{y \in Y}, CT_{SE}\}$. Finally, the transformation component of the cloud service provider module provides $TranCT_{Data}$ to the decryption component of the data access component of the data consumer module.

[122] **Decryption:** Upon receiving $TranCT_{Data}$, the decryption component of the data access component of the data consumer module does the following:

25 If $TranCT_{Data} = \emptyset$, it knows no suitable data exist in the cloud service provider module. Otherwise, $TranCT_{Data} \neq \emptyset$, it takes as inputs the system public parameter PP , its decryption key dk_{DC} and $TranCT_{Data}$, and does the following:

For each tuple $\{T, C_0, C_1, C'_2, \{C_{3,y}, C'_{3,y}\}_{y \in Y}, CT_{SE}\} \in TranCT_{Data}$, it does the following:

1) If node i is a leaf node, then lets $t = attr(i)$, and does the following:

a) If $t \in S$, where S is the data consumer's attribute set and it is given in dk_{DC} , computes

$$30 \quad F_i = \frac{e(C'_{3,i}, D_2)}{e(D_t, C_{3,i})} = \frac{e(g_1^{\beta q_i(0)} \mathcal{H}_1(attr(i))^{r_i}, g_2^{vz})}{e(\mathcal{H}_1(t)^{vz}, g_2^{r_i})} = e(g_1, g_2)^{\beta v z q_i(0)}.$$

b) If $t \notin S$, it sets $F_i = \perp$.

2) If i is a non-leaf node, then does the following:

a) For each child node j of i , it sets $t = attr(j)$, if $t \notin S$, it sets $F_j = \perp$. Otherwise, it computes

$$F_j = \frac{e(C'_{3,j}, D_2)}{e(D_t, C_{3,j})} = \frac{e(g_1^{\beta q_j(0)} \mathcal{H}_1(attr(j))^{r_j}, g_2^{vz})}{e(\mathcal{H}_1(t)^{vz}, g_2^{r_j})} = e(g_1, g_2)^{\beta v z q_j(0)}.$$

35 b) It sets $S_i = \{j \mid F_j \neq \perp\}$ where j is child node of i .

- c) If the size of S_i is less than $d_i + 1$, then it returns $F_i = \perp$. Otherwise, it supposes S_i has $d_i + 1$ elements and computes

$$\begin{aligned}
 F_i &= \prod_{j \in S_i} F_j^{\Delta_{j,S_i}(0)} \\
 &= \prod_{j \in S_i} (e(g_1, g_2)^{\beta v z q_j(0)})^{\Delta_{j,S_i}(0)} \\
 &= \prod_{j \in S_i} e(g_1, g_2)^{\beta v z q_j(0) \Delta_{j,S_i}(0)} \\
 &= e(g_1, g_2)^{\beta v z \sum_{j \in S_i} q_j(j) \Delta_{j,S_i}(0)} \\
 &= e(g_1, g_2)^{\beta v z q_i(0)},
 \end{aligned}$$

where $\Delta_{j,S_i}(0) = \prod_{a \in S_i, a \neq j} \frac{0-a}{j-a}$. For the root node i_{root} , $F_{i_{root}} = e(g_1, g_2)^{\beta v z s}$ or $\perp$ since $q_{i_{root}}(0) = s$. Let

10 $B = F_{i_{root}}$ and if and only if $B \neq \perp$, it computes

$$\frac{e(C'_2, D_1)}{B} = \frac{e(g_1^{sz}, g_2^{\frac{\alpha}{2} + \beta v})}{e(g_1, g_2)^{\beta v z s}} = e(g_1, g_2)^{\alpha s},$$

and sets

$$r_d = \frac{C_1}{e(g_1, g_2)^{\alpha s}} = \frac{r \cdot e(g_1, g_2)^{\alpha s}}{e(g_1, g_2)^{\alpha s}}, m_d = C_0 \oplus \mathcal{H}_t(r_d), s_d = \mathcal{H}_0(r_d, m_d).$$

Then, it checks if $C_1 = r_d \cdot h_3^{s_d}$ and $e(g_1, g_2)^{\alpha s} = h_3^{s_d}$. If no, it fails to decrypt. Otherwise, it computes the symmetric key $k_{SE} = \mathcal{H}_k(m_d)$ and recovers the data $Data = SE.Dec(k_{SE}, CT_{SE})$. Finally, the decryption component of the data access component of the data consumer module puts all the recovered data into the local storage component of the data consumer module.

Revocation

20

[123] To revoke a data consumer u_{DC} from the system, the revocation component of the key generation center module provides u_{DC} as the revocation request to the cloud service provider module. Then, the search component of the cloud service provider module deletes the data consumer's search key shk_{DC} , and/or the transformation component of the cloud service provider module deletes the data consumer's transformation key tk_{DC} .

25

Performance analysis

[124] **Transformation cost of the cloud service provider module:** In the transformation process, for each $CT_{Data} = \{CT_{ABE}, CT_{SE}\} \in TarCT_{Data}$ where $CT_{ABE} = \{T, C_0, C_1, C_2, \{C_{3,y}, C'_{3,y}\}_{y \in Y}\}$, the cloud service provider only needs to compute an exponentiation operation, i.e., $C'_2 = C_2^Z$. Accordingly, the example II is more efficient than the example I in the transformation process.

35

Example III

[125] Example III is proposed to further improve the search performance. Most of the procedures in the example III are the same as those in example I or II, only the different procedures are described below.

The process of data outsourcing

[126] **Storage:** Upon receiving the user-side encrypted data $CT_{DO} = \{u_{DO}, CT_{Data}, CT_{KWS}\}$, the search component of the cloud service provider module retrieves the owner search key shk_{DO} associated with u_{DO} , takes as inputs the system public parameter PP , the owner search key shk_{DO} , and CT_{KWS} , and does the following:

For each CT_{kw_i} in CT_{KWS} , it randomly chooses $y_i \in Z_p$, and computes

$$\begin{aligned} k_{SE,kw_i} &= \mathcal{H}_k \left(e(CT_{kw_i}, shk_{DO}) \right) = \mathcal{H}_k \left(e \left(\mathcal{H}_1(kw_i)^{x_{DO}}, g_2^{\frac{x}{x_{DO}}} \right) \right) \\ &= \mathcal{H}_k(e(\mathcal{H}_1(kw_i), g_2)^x). \end{aligned}$$

Then, it sets $I_{KWS} = \{I_{kw_i} = k_{SE,kw_i}\}_{CT_{kw_i} \in CT_{KWS}}$. The search component of the cloud service provider module provides $\{CT_{Data}, I_{KWS}\}$ to the cloud storage component of the cloud service provider module.

[127] The cloud storage component of the cloud service provider module first sets the cloud-side encrypted data as $CT_{Cloud} = \{CT_{Cloud,kw_i}\}$, where $CT_{Cloud,kw_i} = (Index_{kw_i}, CTData_{kw_i}) = (\emptyset, \emptyset)$. It also sets $Index_{Cloud} = \cup_i Index_{kw_i}$.

[128] After receiving $\{CT_{Data}, I_{KWS}\}$, for each $I_{kw} \in I_{KWS}$, the cloud storage component of the cloud service provider module does the following:

If $I_{kw} \notin Index_{Cloud}$, it sets $Index_{kw} = I_{kw}$, $CTData_{kw} = CT_{Data}$, $Index_{Cloud} = Index_{Cloud} \cup I_{kw}$ and $CT_{Cloud} = CT_{Cloud} \cup CT_{Cloud,kw}$, where $CT_{Cloud,kw} = (Index_{kw}, CTData_{kw})$.

If $I_{kw} \in Index_{Cloud}$, supposes $I_{kw} = Index_{kw_i}$, it then updates CT_{Cloud} by setting $CTData_{kw_i} = CTData_{kw_i} \cup CT_{Data}$.

Data access process

[129] **Search:** Upon receiving $\{u_{DC}, ToK_{kw}\}$, the search component of the cloud service provider module takes as inputs the system public parameter PP , and the user search key shk_{DC} associated with u_{DC} , and sets the target ciphertext data $TarCT_{Data} = \emptyset$ at first. Then, it computes

$$k'_{SE,kw} = \mathcal{H}_k(e(ToK_{kw}, shk_{DC})) = \mathcal{H}_k \left(e \left(\mathcal{H}_1(kw)^{x_{DC}}, g_2^{\frac{x}{x_{DC}}} \right) \right) = \mathcal{H}_k(e(\mathcal{H}_1(kw), g_2)^x).$$

It sets $TarCT_{Data} = TarCT_{Data} \cup CTData_{kw_i}$ if and only if $k'_{SE,kw} = Index_{kw_i}$, where $Index_{kw_i} \in Index_{Cloud}$ is given by the cloud storage component of the cloud service provider module.

[130] Finally, the search component of the cloud service provider module provides $\{u_{DC}, TarCT_{Data}\}$ to the transformation component of the cloud service provider module.

Performance analysis

[131] **Search cost of the cloud service provider:** In the search process, the cloud service provider module needs to compute a bilinear pairing operation, i.e., $k'_{SE,kw} = \mathcal{H}_k(e(ToK_{kw}, shk_{DC}))$. To retrieve all the ciphertext associated with kw , it only needs to perform N_{kw} comparisons to check whether two strings are the same, where N_{kw} is the total number of keywords in the cloud. Accordingly, the example III is more efficient than the example I and II in the search process.

[132] While the disclosure contained herein has set forth some of the preferred embodiments of the present invention, as well as exemplary examples, it is not intended to limit or confine the present invention only to such embodiments or examples. It will be appreciated by those skilled in the art that variations to the modules and steps disclosed may be made without departing from the scope and spirit of the invention. Such variations may include, but are not limited to, selection of alternate practical modules, steps, and application of the process for other purposes or occasions that are not specifically mentioned above.

REFERENCES:

1. Boneh D, Franklin M. Identity-based encryption from the Weil pairing, Proceedings of Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2001
2. Goyal V, Pandey O, Sahai A, et al. Attribute-based encryption for fine-grained access control of encrypted data, Proceedings of the 13th ACM Conference on Computer and Communications Security, 2006
3. Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption, Proceedings of IEEE Symposium on Security and Privacy, IEEE, 2007
4. Boneh D, Di Crescenzo G, Ostrovsky R, et al. Public key encryption with keyword search, Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2004

CLAIMS

1. A cloud data sharing system, including at least one key generation center module, at least one data owner module, at least one cloud service provider module, and at least one data consumer module, wherein

- 5 the at least one key generation center module is configured to-
- generate a system public parameter and a master secret key, and distribute the system public parameter to the at least one data owner module, the at least one cloud service provider module, and the at least one data consumer module;
 - generate and distribute an owner query key to each of the at least one data owner module, and an owner search key associated to the owner query key to the cloud service provider module, based on the system public parameter and the master secret key;
 - 10 - generate and distribute a user query key to each of the at least one data consumer module, and a user search key associated to the user query key to the cloud service provider module, based on the system public parameter and the master secret key;
 - 15 - generate and distribute a decryption key to each of the at least one data consumer module, and a transformation key associated to the decryption key to the cloud service provider module, based on the system public parameter, the master secret key and attribute set of the corresponding data consumer; and
 - generate and distribute revocation information to the at least one cloud service provider module;

20 the at least one data owner module is configured to-

 - generate user-side encrypted data for a data, based on at least one access policy, at least one keyword of the data, the owner query key and the system public parameter received from the at least one key generation center module; and
 - 25 - upload user-side encrypted data to the at least one cloud service provider module;

 the at least one cloud service provider module is configured to-

 - generate cloud-side encrypted data, based on the user-side encrypted data received from a data owner module, the system public parameter and the corresponding owner search key received from the at least one key generation center module;
 - 30 - store the cloud-side encrypted data;
 - perform search on the cloud-side encrypted data, based on at least one search request received from a data consumer module and the corresponding user search key received from the at least one key generation module, and retrieve target cloud-side encrypted data found in the search;
 - 35 - transform the target cloud-side encrypted data to transformed ciphertext data, based on the corresponding transformation key received from the at least one key generation center module;
 - provide the transformed ciphertext data to the corresponding data consumer module; and
 - revoke one or more data consumers based on the revocation information received from the key generation center module; and

40 the at least one data consumer module is configured to-

- generate a search token for a specified keyword, based on the system public parameter and the user query key received from the at least one key generation center module;
 - provide at least one search request including at least one search token to the at least one cloud service provider module; and
 - 5 - decrypt the transformed ciphertext data received from the cloud service provider module, based on the system public parameter and the decryption key received from the key generation center module.
2. The cloud data sharing system according to claim 1, wherein the at least one cloud service provider
10 module revokes one or more data consumers by deleting the corresponding user search key/s and/or transformation key/s.
3. The cloud data sharing system according to claim 1 or 2, wherein the user-side encrypted data
15 contains ciphertext data and index generation requests.
4. The cloud data sharing system according to claim 3, wherein the at least one cloud service provider
module is further configured to generate at least one ciphertext index, based on the index generation
requests received from the at least one data owner module, the system public parameter and the
corresponding owner search key received from the at least one key generation center module; and
20 wherein the cloud-side encrypted data contains ciphertext index and ciphertext data.
5. The cloud data sharing system according to claim 4, wherein the at least one cloud service provider
module transforms the ciphertext data of the target cloud-side encrypted data to transformed ciphertext
data only if attribute set associated with the transformation key matches access policy of the ciphertext
25 data of the target cloud-side encrypted data.
6. The cloud data sharing system according to any of claims 1 - 5, wherein the at least one key
generation center module includes at least one initialization component, at least one registration
component, and at least one revocation component.
30
7. The cloud data sharing system according to any of claims 1 - 6, wherein the at least one data
owner module includes at least one local storage component and at least one data outsourcing
component.
8. The cloud data sharing system according to claim 7, wherein the at least one data outsourcing
35 component includes at least one encryption component and at least one index generation component.
9. The cloud data sharing system according to any of claims 1-8, wherein the at least one cloud
service provider module includes at least one cloud storage component, at least one search component,
40 and at least one transformation component.

10. The cloud data sharing system according to any of claims 1-9, wherein the at least one data consumer module includes at least one local storage component and at least one data access component.

5

11. The cloud data sharing system according to claim 10, wherein the at least one data access component includes at least one search token component and at least one decryption component.

12. A method for sharing data using a cloud data sharing system according to any of claims 1-11, including at least five processes: initialization process, registration process, data outsourcing process, data access process and revocation process, wherein

- the initialization process includes-
- generating and providing a system public parameter, by at least one key generation center module, to at least one data owner module, at least one cloud service provider module and at least one data consumer module; and
 - generating a master secret key by the at least one key generation center module, and keeping the master secret key confidential in the at least one key generation center module;
- the registration process includes-
- generating and providing at least one owner query key, by the at least one key generation center module, to each of the at least one data owner module, based on the system public parameter and the master secret key;
 - generating and providing one owner search key associated to each owner query key, by the at least one key generation center module, to the at least one cloud service provider module, based on the system public parameter and the master secret key;
 - generating and providing at least one user query key, by the at least one key generation center module, to each of the at least one data consumer module, based on the system public parameter and the master secret key;
 - generating and providing one user search key associated to each user query key, by the at least one key generation center module, to the at least one cloud service provider module, based on the system public parameter and the master secret key;
 - generating and providing at least one decryption key, by the key generation center module, to each of the at least one data consumer module, based on the system public parameter, the master secret key and corresponding data consumer's attribute set; and
 - generating and providing one transformation key associated to each decryption key, by the key generation center module, to the at least one cloud service provider module, based on the system public parameter, the master secret key and corresponding data consumer's attribute set;
- the data outsourcing process includes-
- generating and providing user-side encrypted data, by the at least one data owner module, to the at least one cloud service provider module, based on at least one access policy, at least

one keyword of the data to be encrypted, the system public parameter and the at least one owner query key received from the at least one key generation center module; and

- generating and storing cloud-side encrypted data, by the at least one cloud service provider module, based on the user-side encrypted data received from the at least one data owner module, the system public parameter and the owner search key associated to each of the at least one owner query key received from the at least one key generation center module;

the data access process includes-

- generating and providing at least one search request including at least one search token, by the at least one data consumer module, to the at least one cloud service provider module, based on the system public parameter and the at least one user query key received from the at least one key generation center module;

- performing search on the cloud-side encrypted data, by the at least one cloud service provider module, based on at least one search request received from a data consumer module and the corresponding user search key received from the at least one key generation module, and retrieving target cloud-side encrypted data found in the search;

- transforming the target cloud-side encrypted data to transformed ciphertext data, by the at least one cloud service provider module, based on the transformation key and providing the transformed ciphertext data to the corresponding data consumer module; and

- decrypting the transformed ciphertext data, by the corresponding data consumer module, based on the decryption key received from the at least one key generation center module; and

the revocation process includes-

- providing revocation information, by the at least one key generation center module, to the at least one cloud service provider module; and

- revoking at least one relevant data consumer, by the cloud service provider module, based on the revocation information received from the at least one key generation center module.

13. The method for sharing data according to claim 12, wherein the at least one cloud service provider module revokes the at least one data consumer by deleting the corresponding user search key/s and/or transformation key/s.

14. The method for sharing data according to claim 12 or 13, wherein the user-side encrypted data contains ciphertext data and index generation requests.

15. The method for sharing data according to claim 14, wherein at least one ciphertext index is generated by the at least one cloud service provider module, based on the index generation requests received from the at least one data owner module, the system public parameter and the corresponding owner search key received from the at least one key generation center module; and

wherein the cloud-side encrypted data contains ciphertext index and ciphertext data.

16. The method for sharing data according to claim 15, wherein the at least one cloud service provider module transforms the ciphertext data of the target cloud-side encrypted data to transformed ciphertext data only if attribute set associated with the transformation key matches access policy of the ciphertext data of the target cloud-side encrypted data.