

Secure File Management, System and the Method Thereof for Cloud Storage

Field of the Invention

5 **[0001]** The present invention relates to file management. More particularly, the present invention relates to a system and method adapted for secure file management on cloud storage.

Background

[0002] The boost in the demand for scalable, reliable and fault-tolerant
10 computing has led to the increasing popularity of cloud computing. Cloud computing delivers over-the-Internet computing services allowing users to access the data or even software through the networks.

[0003] One of the most common services provided by cloud computing is cloud storage. Cloud storage conveniently allows users to deposit and access their digital data
15 without any geographical limitations. These cloud storage service providers typically implement certain security measures to protect the stored data. However, once the security measures implemented by the service providers are breached, the data is no longer secured.

[0004] Further, such data security measures are implemented to protect against
20 third party attacks. The service providers are usually free to access the data hosted by them.

Summary

[0005] One aspect of the present invention seeks a secure data management service for offering a cloud computing service to users for storing documents thereon through users' devices over a network. The secure data management service comprises a cloud based security service provider for hosting a data storage for storing the documents in an encrypted form, wherein the encrypted documents are associated to respective licenses and fingerprints, the documents are stored in an encrypted form; and users' devices having a program module resided therein, the program module is adapted for remotely accessing the cloud computing service in an authenticated manner. The fingerprints are distributed and structured in accordance with circles of users, the circles are formed in a tree structure with circles from a lower level branched out from a higher level of the tree structure, when a file is to be accessed by a non-owner user, the service provider verifies a common ancestor circle at a higher level.

[0006] In one embodiment, each user's device has a digital certificate having a public-private key pair registered with the service provider, wherein a private key of the digital certificate is further encrypted by a passphrase that is stored on the service provider. The passphrase may be created by an authentication service of the service provider, not the user, and the passphrase is stored remotely from the digital certificate.

[0007] In another embodiment, the service provider is operationally identifying the respective circles where the relevant users belong, tracing the higher level of circles to identify the common ancestor circle, and selecting the next successive circle towards the circle of the owner of the files/documents to verify the fingerprints stored therein against that of the encrypted document.

[0008] Yet, the service provider may further comprise at least one bloom filter for each circle, wherein the bloom filters are adapted for document fingerprinting.

[0009] In another aspect, there is provided a method for offering a cloud computing service to users for storing documents therein through users' devices over a network. The method comprises the steps of authenticating users to remotely access the cloud computing service through the users' devices; generating respective licenses and fingerprints for the documents to be stored; encrypting the documents; storing the respective licenses, fingerprints and the encrypted documents on the cloud computing service; distributing and structuring the fingerprints in accordance with circles of users, the circles are formed in a tree structure with circles from a lower level branched out from a higher level of the tree structure; verifying a common ancestor circle at a higher level of the tree for allowing access to the encrypt document by a non-owner user.

[0010] In one embodiment, the method further comprises registering a digital certificate with the cloud computing service, the digital certificate having a public-private key pair for each user device; and encrypting the private key of the digital certificate by a passphrase that is stored on the service provider.

[0011] In another embodiment, the method further comprises creating the passphrase by an authentication service of the cloud computing service, not the user; storing the passphrase remotely from the digital certificate.

[0012] In yet a further embodiment, the method may further comprises identifying the respective circles where the relevant users belongs; tracing the higher level of circles to identify the common ancestor circle; selecting the next successive circle of the common ancestor circle towards the circle of the owner of the documents;

and verifying the fingerprints stored on the next successive circle against that of the encrypted document.

Brief Description of the Drawings

5

[0013] This invention will be described by way of non-limiting embodiments of the present invention, with reference to the accompanying drawings, in which:

[0014] **FIG. 1** illustrates a schematic diagram of a cloud computing service 110 in accordance with one embodiment of the present invention;

10 **[0015]** **FIG. 2** illustrates a sequence diagram for user login process in accordance with one embodiment of the present invention;

[0016] **FIG. 3** illustrates a sequence diagram of uploading documents to the cloud storage of a cloud computing service in accordance with one embodiment of the present invention;

15 **[0017]** **FIG. 4** illustrates a sequence diagram for downloading documents to the cloud storage of a cloud computing service in accordance with one embodiment of the present invention;

[0018] **FIG. 5** illustrates a sequence diagram of fingerprint query in accordance with one embodiment of the present invention; and

20 **[0019]** **FIG. 6** illustrates a further sequence diagram of a fingerprint registration process in accordance with one embodiment of the present invention.

Detailed Description

[0020] In line with the above summary, the following description of a number of specific and alternative embodiments are provided to understand the inventive features of the present invention. It shall be apparent to one skilled in the art, however that this invention may be practised without such specific details. Some of the details
5 may not be described at length so as not to obscure the invention. For ease of reference, common reference numerals will be used throughout the figures when referring to the same or similar features common to the figures.

[0021] FIG. 1 illustrates a schematic diagram of a cloud computing service **110** in accordance with one embodiment of the present invention. The cloud computing
10 service **110** is one of the features can be provided over a network, such as Internet **102**. Users may utilize the services offered under the cloud computer service **110** through the Internet **102**. One of the services provided under the cloud computing service **110** is cloud storage, which allows users to store, manage and share their digital files over the Internet **102**. Users of the cloud computing service **110** may be registered users **104**,
15 **106** to the cloud computing service **110** and non-registered users **108**. Typically, the non-registered users **108** are dependent on the registered users **104**, **106**. The registered users **104**, **106** must invite the non-registered users **108** for sharing files stored under their own accounts. The registered users **104**, **106** may limit files to be shared with the non-registered users **108** and certainly, they may also limit the access right to the shared
20 files at will. As shown in FIG. 1, the registered users may include individual registered users **104** and group-registered users **106**. A group-registered user may include an administrator for managing the cloud computing service for the sub-users under the same group.

[0022] These users access the cloud computing service **110** through computing devices **105**, for example, personal computers, laptops, tablets, or even mobile phones. It is possible that one user may possess more than one computing device **105**. These devices **105** may connect to the Internet **102** directly, or behind a server **107** or a local
5 area network. The cloud computing service **110** may allow the users **104**, **106** as well as an administrator **106** to manage and control file storing, access and sharing through graphical user interface (GUI). Such GUI may be available under the operating system, a dedicated program module or web-based applications that can be accessed through any web browsers, which are resided on the devices.

10 [0023] In one preferred embodiment, the users' devices may be deployed with a local dedicated program module, herein after also referred to as gatekeeper application, for carrying out the features offered by the cloud computing service **110**. Such gatekeeper application may offer better flexibility, more control, and higher security over the digital files stored at the cloud computing service **110**. It is desired that some
15 of the security features are carrying out locally and they are not shared with the system hosting the cloud computing service **110**, such that the cloud computing service **110** provider will not be able to decrypt the files and documents thereby maintaining the privacy of the users over the files stored at the cloud computing service **110**. Preferably, only the users authorized by the original owner of the files and documents
20 are given the access to these files and documents. The owner of the files and documents are provided with means for controlling the access right.

[0024] For the purpose of this application, the users' devices may also be referred herein as Endpoint. All registered users are assigned with their own user's

name and password for accessing the cloud computing service **110**, and each endpoint will be assigned an Endpoint ID and an Endpoint Key. The Endpoint ID is assigned by the cloud computing service **110**, whilst the Endpoint Key is generated by the devices through local endpoint devices' (hardware and software) configurations. In the special
5 case of the NULL endpoint, the Endpoint ID and Endpoint Key may be always 0 (zero). NULL endpoint occurs, for example, when there is no physical endpoint or when endpoint is unregistered. NULL endpoint can also occur when user accesses the cloud computing service through web browser, instead of dedicated client application. Preferably, the cloud computing service accessed through web browser is limited to
10 some administrative purposes only, not for accessing the documents.

[0025] For each User-Endpoint combination, except for the NULL endpoint, a digital certificate having a public-private key pair, such as an OpenPGP RSA certificate, is generated and registered with the cloud computing service **100**. The certificate may be generated by the local device, or the cloud computing service **100**.
15 Each certificate contains a public key and a private key. The private key is further encrypted by a passphrase, which is the standard practice in public key infrastructure.

[0026] The original certificate is installed on the endpoint to which it is assigned. The cloud computing service **100** will contains the public key as well as the passphrase.

20 **[0027]** For each user, the cloud computing service **110** further keeps a unique numbers or random bits, such as cryptographic salt. Salt is a random value generated when the user is registered with the system.

[0028] **FIG. 2** illustrates a sequence diagram for user login process in accordance with one embodiment of the present invention. The login process involves the user **202**, a gatekeeper **204**, an authentication service **206** and a session manager **208**. The gatekeeper **204** is a program module resides on the user's device while the authentication service **206** and the session manager **208** are resided at the cloud computing service **110**. At start, the user **202** logs in to the cloud computing service **110** with a predefined username and password through the gatekeeper **204**. The gatekeeper **204** then requests for a timestamp and a user's salt from the authentication service **206** of the cloud computing service **110**. Once the requested timestamp and salt is obtained from the authentication service **206** by the gatekeeper **204**, the gatekeeper **204** returns with the login user name, password, a Nonce, an Endpoint ID, the timestamp, and a signature to the Authentication Service **206**. To compute the signature for login, the gatekeeper **204** retrieves the Endpoint ID from its configuration file of the device, and computes the Endpoint Key from the configuration of the endpoint it is running. Thereafter, it generates a random string such as cryptography nonce. Based on these information, the gatekeeper application then computes a key, K, as follows:

[0029]
$$K = H(\text{Salt} + \text{Password}) \text{ XOR EndpointKey}$$

[0030] where, H is a hash function of the salt concatenated with the user password, and computes the exclusive-OR (XOR) of the hash value and the Endpoint Key. The hash function H can be any publicly available cryptographic hash function, such as SHA family of hash functions, for example.

[0031] Subsequently, the signature is being compute as follows:

[0032] $\text{Signature} = E(\text{Timestamp} + \text{Username} + \text{EndpointID} + \text{Nonce}, K)$

[0033] A string is constructed by putting together the timestamp retrieved from the authentication service **206**, the username supplied by the user **202**, end endpoint retrieved from its configuration and a random nonce, and then encrypt the result using
5 the key K generated above.

[0034] For the purpose of this invention, cryptographic salt can be suitably used for generating a key for a given user password. A new cryptographic salt is needed when a new password is provided. The cryptographic nonce, on the other hand, is suitable used for each service request, such as, login, logout, document
10 download/upload, query for access rights, etc.

[0035] The authentication service **206** further sends a session creation request to the session manager **208** with the user name and a session key. The session key can be generated randomly by the cloud computing service. It is desired that the session key is sufficiently long and difficult to guess. It is further desired that the size of the session
15 key matches the underlying cryptographic algorithm that performs the encryption/signature. An AES (advanced encryption standard) key, for example, can be adapted herewith. The session manager **208** accordingly returns a session ID to the authentication service **206** conclude the login. User may start to utilize the cloud computing service **110** upon receiving the login confirmation. No session ID is
20 generated if the user fails to login to the cloud computing service.

[0036] The SessionID and SessionKey allow the user/application to communicate securely later without referring to certificates or passwords. Each login

session will expire after a configurable period of time. Such period of time can be pre-set as default setting, or pre-defined by the users or administrator, and once set, stored on the cloud computing service in the respective users' configuration file. Before the session expires, the user (i.e. the gatekeeper **204**) may renew the session using the SessionID and SessionKey. When a login session is renewed, the SessionID remains the same but the SessionKey will be replaced by a new SessionKey.

[0037] Unlike a typical RSA certification, a passphrase is adapted for the login authentication and the passphrase is created by the authentication service **206**, not the user. In the present embodiment, the passphrase is generated by systems, not user, and is located remotely from the certificate.

[0038] The present invention ensures that the passphrase generated to protect the private key is a strong passphrase that cannot be easily guessed. Such passphrase can also be easily removed or revoked in case where the endpoint (i.e. the device) is lost, stolen, or any terminated from the cloud computing service **110** for any reasons. In such case, even an attacker who hacks into the device and obtain the certificate, but without the passphrase, which is not stored on the same device, the attacker would not be able to access the cloud computing service **110**. Accordingly, the user's data can be safeguard even when the cloud computing service's **110** securities are compromised since the private keys are not stored on the server.

[0039] In the cloud computing service **110**, the EndpointID and EndpointKey are regarded as added factors for users' authentication. Each user will be only allowed to login from a set of pre-registered devices.

[0040] In another embodiment of the present invention, additional factors can be integrated to the system for better user authentications. For example, if the user has a USB token that stores another key K' , the new EndpointKey' can be derived from:

[0041] $\text{EndpointKey}' = f(\text{EndpointKey}, K')$

5 [0042] wherein the EndpointKey is the original EndpointKey derived from hardware and software configurations, and the function “f” can be any cryptographic process provided by the USB token. If the USB token is passive (i.e., it provides only a key), then the function “f” can be anything cryptographic one-way function defined by the gatekeeper 202, such as a SHA-1 hash or any other suitable function.

10 [0043] FIG. 3 illustrates a sequence diagram of uploading documents to the cloud storage of a cloud computing service in accordance with one embodiment of the present invention. The cloud computing service includes a gatekeeper application 302, a client DLP engine 304, a client crypto engine 306, an authentication service 308, a document service 310, a license service 312 and a server DLP service 314. The

15 gatekeeper application 302, the client DLP service 304 and the client crypto engine 306 are resided at the users' device, while the authentication service 308, the document service 310, the license service 312 and the DLP service 314 are resided at the cloud computing service. When the user uploads a document to the cloud storage, the user is prompted for a login request, if the user has not yet logged in to the cloud computing

20 service. The login procedure can be similar to that in FIG. 2. The gatekeeper application 302 will transmit the user ID, terminal ID and prove of identity. The proof of identity may be a combination of user identity and digital certificate issued from the user end through the gatekeeper application to the authentication service 308 for

authentication. The authentication service **308** then responds with a session credentials allowing the user to upload the document once the information is authenticated. Such authentication process may require being done only once until the session ends or logout.

5 **[0044]** Once the user is authenticated, the original document to be uploaded is encrypted through the client cryptography engine **306**. To encrypt the document, the gatekeeper application **302** generates a random key and then encrypts the random key by the certificate's public key as a license. Both the encrypted document and the self-generated license are uploaded. In other embodiment, additional licenses, such as
10 rights manager license, may be uploaded as well. These licenses can be required for downloading and viewing the files stored on the cloud computing service.

[0045] Before the document is uploaded, the gatekeeper application **302** sends the document to the client DLP Engine **304** to compute a fingerprint for that document. Thereafter, the client crypto engine **306** encrypts the document and then returns an
15 encrypted document with a user license and a rights manager license. The rights manager license is a license for a rights manager. The rights manager for a given user is one who assigns access rights on the user's documents to other users. The rights manager needs to know all the document keys. Accordingly, when the user uploads a document, the rights manager license is also uploaded such that document key can be
20 encrypted by the rights manager's public key. The gatekeeper application **302** then upload the encrypted document to the document service **310**. The document service **310** then returns the gatekeeper application **302** with a new document ID. The gatekeeper application **302** then uploads the fingerprint to the DLP service **314** and

once uploaded, a confirmation is returned to the gatekeeper application **302**. The gatekeeper application **302** further uploads the license to the license service **312** and once uploaded, a confirmation is returned to the gatekeeper application **302**.

[0046] It is understood that the encrypted document, the fingerprints and the
5 license can be uploaded at any sequence, without limitation. In another embodiment, it is desired that the encrypted document is uploaded only after the licenses and the fingerprints are uploaded to ensure that the licenses and fingerprints are always present with the associated documents.

[0047] **FIG. 4** illustrates a sequence diagram for downloading documents to the
10 cloud storage of a cloud computing service in accordance with one embodiment of the present invention. This diagram provides a downloading session to download a document uploaded through the operation illustrated in **FIG. 3**. Similarly, the downloading session involved the gatekeeper application **302**, the client crypto engine **306**, the authentication service, the document service **310** and license service **312**. The
15 downloading session further involves an access controller **420**, which is resided at the cloud computing service.

[0048] Similarly, when the user downloads a document from the cloud storage of the cloud computing service, the user is prompted for a login request, if the user is no in a login or active session on the cloud computing service. Similarly, the login
20 request will require the user ID, terminal ID and prove of identity. This information will be transmitted directly to the authentication service **308** for authentication. The authentication service **308** then responds with a session credentials allowing the user to upload the document once the information is authenticated. Such authentication

process may be required to be done only once until the session ends or logout, as mentioned before.

[0049] Once the user is authenticated, a request for downloading document from the cloud storage is sent to the document service **310** from the gatekeeper application **302**. Such request may be initiated automatically when the user is attempting to drag and drop the document at from the cloud storage. The document service **310** checks with the access controller **420** for the type/range of access rights of the user to access the document to be downloaded, and once a confirmation is received from the access service **420** confirming the right to access the document to be downloaded, the document service **310** releases the document in an encrypted form to the user. Once the encrypted document is received, the gatekeeper application **302** sends a request for license to the license service **312**. The license service then respond to with a user license to obtain the license generated when the document was uploaded. The gatekeeper application **302** then decrypt the document with the client crypto engine **306**. Once the encrypted document is decrypted, the original document is returned to the gatekeeper application.

[0050] FIG. 5 illustrates a sequence diagram of fingerprint query in accordance with one embodiment of the present invention. The fingerprint query process may vary in accordance with the type of documents involved, which can be detected by the client DLP engine. The fingerprint query requires a client fingerprint engine **522**, a document converter **524**, a text analyzer **526** and a multimedia analyzer **528**, which can be resided at either the gatekeeper application, or at the cloud computing service or both. In the present embodiment, documents are being classified in two categories: textual

documents and multimedia documents. In either category, these documents need to be converted to a specific form before further processing. For textual documents, the client fingerprint engine **522** sends the documents to the document converter **524** to extract the textual data. The textual data is then returned to the client fingerprint engine **522**. The client fingerprint engine **522** then sends the extracted textual data to the text analyzer **526** for processing.

[0051] In one embodiment, the text analyzer may processes the text document with the following steps:

[0052] 1. Removing white spaces and other format information that does not contain actual content;

[0053] 2. Splitting them into smallest processing units; it is configurable by user. The units may include paragraphs, sentences, data cells (from text based spreadsheets), etc.;

[0054] 3. Removing insignificant data units. For example, paragraphs that only contain a few words, a sentence with only one word, etc.;

[0055] 4. Computing a MD-5 hash value, or any cryptographic one-way function for each data unit to produce the fingerprint of the processed document.

[0056] For multimedia documents, the document converter **524** extracts the features of the multimedia document that are being processed. It produces a set of feature vectors, which can be considered as a concise version of the original content. The actual algorithms depend on the type of the multimedia data. For example, for

image data, a scaled-down version of the image can be produced. For videos, segments of the video are to be extracted with the known method, and then extract key frames from the segments. Then, the key frames may be processed in a same way as processing the images. The features extracted by the document converter **524** are taken
5 for generating the fingerprints. A MD-5 hash function, for example, may be used to generate the required fingerprint.

[0057] **FIG. 6** illustrates a further sequence diagram of a fingerprint registration process in accordance with one embodiment of the present invention. The fingerprint associating the encrypted document with user groups that the owner of the encrypted
10 registration is provided for uploading fingerprints with the encrypted document and document belongs. The fingerprint registration also starts from the gatekeeper application **302** of the user's device. The fingerprint registration involves a DLP service **604**, a DLP circle service **606** and a bloom filter **608**.

[0058] The gatekeeper application **302** is the same or substantially the same
15 gatekeeper application of FIG. 2 and 3. It is provided on the user side for allowing the user to communicate with the cloud storage. The DLP service **604** is provided for deciding which circle of the fingerprints it belongs based on user/document information. The fingerprints tagged to the encrypted documents helps the cloud computing service to identify the encrypted documents without the need of revealing or
20 decrypting the full contents of the encrypted documents.

[0059] To manage the documents effectively while protecting the users privacy, the associated fingerprints are distributed among circles, which are organised in a tree structure as exemplified in FIG. 7. In this example, the tree structure has three levels

comprising a circle C1 placed at level 1, which is the highest level; circles C2 and C3, branched out from the circle C1 to form the level 2; and circles C4, C5, C6 and C7 that form level 3, wherein circle C4 and C5 are branched out from circle C2 and C6 and C7 are branched out from circle C3. Each user of the cloud computing service belongs to one of the circles and the documents are tied to the users. The associated fingerprints of the documents are stored in the relevant circles. Preferably, the relevant circles include the originating circle where the user belongs as well as all the circles of the immediate preceding level along the same branch/line till the circle at the highest available level.

10 **[0060]** Say, for example, a User A belongs to circle C2 and a User B belongs to circle C3, and User A intends to send a document to User B through the cloud computing service. The system will first identify a common ancestor circle of C2 and C3 by backtracking the circles of the preceding higher level along the same branch. In the given example, the circle C1 is a common ancestor of circles C2 and C3. Once the common ancestor is found, the system select the next successive circle of the same line of the originating circle, i.e. circle C2, to check the fingerprints therein against the fingerprints of the document to be sent. In another example, when user B is in circle C6, the common ancestor is also C1, and next successive circle to be checked is still circle C2. In a further example, when user A is in circle C4, and user B is in circle C6, 15 the common ancestor circle will be identified as circle C1, and the next successive circle will also be circle C2. In yet another example, when user A is in circle C4 and user B is in circle C5. The common ancestor will be circle C2 and from circle C2 the

next successive circle is circle C4 and therefore, circle C4 will be checked for fingerprints.

[0061] Documents fingerprinting can be used for managing small changes in documents and performance in terms of matching accuracy and speed. The present invention combines the two, and with an aid of Bloom Filters, the present system is able to manage confidential documents in a very fine-grained manner, with high-speed and the privacy of the documents is protected.

[0062] The DLP circle service **606** on the other hand is adapted for mapping circles to the Bloom filters

10 [0063] The Bloom filter **608** is an in-memory probabilistic data structure that allows constant time insertion and query for membership. A piece of arbitrary data, for example, can be inserted to the bloom filter and later query for its existence. The operations can be carried in constant time.

[0064] The fingerprint registration comprises uploading/querying for fingerprint from the DLP service **604**; dispatching the fingerprints to the DLP circle service **606**; registering the fingerprints with the Bloom filter **608**; providing confirmation of the fingerprints registration from the Bloom filters **608** to DLP circle service **606** to DLP service **604** and finally to the gatekeeper application **302**.

[0065] In another embodiment, the cloud computing service may require two Bloom filters for each circle of user group. The circles can be organized in to a hierarchical tree structure, and so are the Bloom filters. When the fingerprints of a

document are uploaded, the fingerprint can be inserted into the document's owner's circle bloom filter, and all its ancestors in the tree.

[0066] Once the fingerprints are registered with the cloud computing service, it is possible to query for fingerprints using a fingerprint web service provided by the cloud computing service. The overall process is essentially the same as the registration process.

[0067] To determine if a given document (including plain text data such as email) contains confidential information from a given circle, the fingerprints from the document in question are generated and query against the bloom filter that is associated with the given circle. In one embodiment, all documents uploaded onto the cloud computing service may be segmented into overlapping pieces and a fingerprint is generated for each overlapping piece. In such case, partial content of the documents can be detected by the cloud computing service.

[0068] After that, a heuristic policy can be used to determine if the data is indeed contains confidential information. For example, if 1000 fingerprints are generated from the document in question (recall that each fingerprint is just an MD-5 hash), and more than 200 fingerprints match those in the Bloom Filter (i.e., the Bloom Filter answers "Yes" for these 200 fingerprints), it can be concluded that the document contains confidential data. The number 200 exemplifies herein is a configurable threshold that can be adjusted according to the needs. It is also possible to give different thresholds for consecutive matched fingerprints. For example, out of the 1000 fingerprints, only 10 are matched, but these matches are consecutive, it can also conclude that the document contains confidential information.

[0069] To avoid false alarms, a second negative Bloom Filter that stores data that are explicitly marked as non-confidential. For example, those template documents email and forms that may not do not contain actual data, and many documents generated from these templates are similar. The system may ignore such fingerprints.

5 **[0070]** Another use of the negative Bloom Filter is when the system administrator finds a false alarm on some documents that have no confidential data but regarded as so, the administrator can allow these document to pass the confidential document detection system by explicitly putting them into the negative Bloom Filter.

[0071] While specific embodiments have been described and illustrated, it is
10 understood that many changes, modifications, variations and combinations thereof could be made to the present invention without departing from the scope of the invention.

Claims

1. A secure data management service for offering a cloud computing service to users for storing documents thereon through users' devices over a network, the secure data management service comprising:

5 a service provider for hosting a data storage for storing the documents in an encrypted form, wherein the encrypted documents are associated to respective licenses and fingerprints, the documents are stored in an encrypted form; and

users' devices having a program module resided therein, the program module is adapted for remotely accessing the cloud computing service in an authenticated manner,

10 wherein the fingerprints are distributed and structured in accordance with circles of users, the circles are formed in a tree structure with circles from a lower level branched out from a higher level of the tree structure, when a file is to be accessed by a non-owner user, the service provider verifies a common ancestor circle at a higher level.

15

2. The secure data management service according to claim 1, each users' device has a digital certificate having a public-private key pair registered with the service provider, wherein a private key of the digital certificate is further encrypted by a passphrase that is stored on the service provider.

20

3. The secure data management service according to claim 2, wherein the passphrase is created by an authentication service of the service provider, not the user, and the passphrase is stored remotely from the digital certificate.

4. The secure data management service according to claim 1, wherein the service provider operationally identifying the respective circles where the relevant users belong, tracing the higher level of circles to identify the common ancestor circle, and
5 selecting the next successive circle towards the circle of the owner of the files/documents to verify the fingerprints stored therein against that of the encrypted document.

5. The secure data management service according to claim 4, wherein the service
10 provider further comprises at least one Bloom filter for each circle, wherein the Bloom filters are adapted for document fingerprinting.

6. A method for offering a cloud computing service to users for storing documents therein through users' devices over a network, the method comprising:
15 authenticating users to remotely access the cloud computing service through the users' devices;

generating respective licenses and fingerprints for the documents to be stored;

encrypting the documents;

storing the respective licenses, fingerprints and the encrypted documents on the
20 cloud computing service;

distributing and structuring the fingerprints in accordance with circles of users, the circles are formed in a tree structure with circles from a lower level branched out from a higher level of the tree structure;

verifying a common ancestor circle at a higher level of the tree for allowing access to the encrypt document by a non-owner user.

- 5 7. The method according to claim 6, further comprising:
 registering a digital certificate with the cloud computing service, the digital
 certificate having a public-private key pair for each user device; and
 encrypting the private key of the digital certificate by a passphrase that is stored
 on the service provider.

10

8. The method according to claim 7, further comprises creating the passphrase by an authentication service of the cloud computing service, not the user; storing the passphrase remotely from the digital certificate.

9. The method according to claim 6, further comprising:
15 identifying the respective circles where the relevant users belongs;
 tracing the higher level of circles to identify the common ancestor circle;
 selecting the next successive circle of the common ancestor circle towards the
 circle of the owner of the documents; and

 verifying the fingerprints stored on the next successive circle against that of the
20 encrypted document.

10. The method according to claim 6, wherein the cloud computing service further comprises at least one bloom filter for each circle, wherein the bloom filters are adapted for document fingerprinting.