Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 174 (2017) 1340 – 1355

Procedia Engineering

www.elsevier.com/locate/procedia

13th Global Congress on Manufacturing and Management, GCMM 2016

Research on data security technology based on cloud storage

Rongzhi Wang*

Institute of Computer
Hulunbuir College
Hulunbuir, Inner Mongolia,China
rongzhiwang@163.com

Abstract

With the development of cloud storage system and its application in complex environment, its data security has been more and more attention. On the one hand, node crashes or external invasion are likely to lead to incomplete data; on the other hand, when the data is incomplete, because the cloud service provider deliberately concealed or other factors, the user cannot be promptly informed of the change. In view of the above problems, this paper makes a deep research, and puts forward a secure storage system based on how to ensure the data availability when data integrity and data are not complete. In this paper, we begin with the availability of data; the research focuses on the confidentiality of data, the loss of data recovery and data recovery. In this paper, we propose a data secure storage scheme based on Tornado codes (DSBT) by combining the technique of symmetric encryption and erasure codes. Program uses boot password to solve the traditional data encryption in the problem of key preservation and management; system design by correcting Tornado data redundancy code delete code in order to solve problems and recover lost data; through a hash keyed to Tornado code with error correction function so as to solve the problem of data tampering. On this basis, the paper continues to carry out research on data retrieval (POR). Based on the classic POR algorithm based on BLS short signature, the trusted log is introduced, and the trusted log is used to provide the user with the test results. Finally, combined with the DSBT scheme, the computational efficiency of the POR algorithm is optimized, which has nothing to do with the file size, which can achieve the calculation complexity of the constant level. According to the above scheme, this paper implements a secure cloud storage prototype system based on Cassandra. The test shows that the system can provide strong data loss recovery ability, effectively resist the Byzantine fault, in the back of the desirable detection ability is also prominent, but also has very high computation efficiency, especially in the face of large files. This paper studies the modeling and analysis methods of some key problems of data security in cloud storage, such as encryption storage, integrity verification, access control, and verification and so on. Through the data segmentation and refinement rules algorithm to optimize the access control strategy, using the data label verification cloud data integrity, using replica strategy to ensure the data availability, the height of authentication to strengthen security, attribute encryption method using signcryption technology to improve the algorithm efficiency, the use of time encryption and DHT network to ensure that the cipher text and key to delete the data, so as to establish a security scheme for cloud storage has the characteristics of privacy protection.

Peer-review under responsibility of the organizing committee of the 13th Global Congress on Manufacturing and Management

Peer-review under responsibility of the organizing committee of the 13th Global Congress on Manufacturing and Management
doi:10.1016/j.proeng.2017.01.286

Keywords Cloud storage; data security; strategy optimization; data redundancy; Tornado code

1. Introduction

Cloud storage resources as a service available to users through the Internet, cloud computing infrastructure as a service (Infrastructure as a Service LAAS) is a kind of important form, and is Fl Yi deep into our life. Cloud storage has many traditional storage incomparable advantages, such as to the complexity of the user shielding bottom hardware management, whenever and wherever possible data access, on-demand resource deployment. In recent years, some well-known IT companies have an important part and started to provide cloud storage service as its cloud computing, such as Amazon J Simple Storage 100 Service (S3), Google Cloud Storage, and Rackspace Cloud Files. Also appeared on the cloud storage infrastructure services, such as the upper application services [1], such as EMC company's online document storage and backup services, Mozy, etc. Although it has many advantages and has been sought after by many IT companies, however, the cloud storage has not been widely used, one of the important reasons is that data security issues. Gartner company in 2009 survey results show: more than 70% of respondents believe that the recent CTO is not the primary reason for cloud storage is the cloud storage data security and privacy concerns. In cloud storage, the user data stored on the server and storage devices, cloud storage service providers, these storage devices are no longer subject to regulation and control of direct users, equipment failure, administrator disoperation, internal leakage, the server was hacked and other reasons may lead to leakage of user, sensitive and important data loss or damage. On the one hand, users are concerned about the confidentiality of their data cannot be guaranteed. The weak point of confidentiality refers to the intruder cannot obtain the data stored in the cloud of meaningful information, confidentiality strong point refers to even cloud storage service providers are unable to obtain meaningful information of user data. Strong confidentiality is very necessary, because some special data such as the company's commercial secrets, the government's confidential information or other data related to the user's privacy needs to ensure that such strong confidentiality. On the other hand, users also worry about the availability of data, that is, cloud storage service providers can provide timely and correct data access services [2]. In recent years, the continuous burst of cloud storage accidents even more drama people's concerns, such as the March 2009 Google documents leaked user documents, in April 2011 EC2 Amazon failure, etc. As a new type of data storage, data security in cloud storage not only covers the traditional data storage security, but also extends its own characteristics. In cloud storage, the user's data is stored on a remote storage device. Although the cloud storage service provider has the responsibility to provide reliable storage service for users, but for various reasons may lead to user data security and integrity is destroyed, and the service provider may have to avoid the loss of economy and reputation of the user while trying to hidden full of the destruction. Also in the cloud storage, data is usually cut into slices and encrypted stored in different storage nodes, so as to avoid the loss of data caused by a single node of the original integrity of the information leakage. But in this case, when a node collapse, the loss of the patch will cause a lot of data cannot be restored. These features make for some of the traditional network security and storage security technology in cloud storage environment cannot be fully applicable, for example in the traditional storage technology, using message digital signature to ensure the integrity of the file, but in the cloud storage, the data stored in the remote server, the data retrieve and regularly check the signature to all detection of data integrity is not feasible. In addition to cloud storage using stored data divided into fixed size pieces, so there is a need for a reliable and efficient method of fault tolerance, can ensure that even if the plurality of slices is lost, rely on remaining can also restore the integrity of files [3]. The data security schematic diagram based on the cloud storage is shown in Figure 1 and 2.

The key technology of data security in cloud storage is a broad concept, which contains many aspects, so it is necessary to explain the main content of this paper. Study of data security in cloud storage has a lot of research work, cloud storage in access control, etc. This paper studies the cipher text search; the main concern is data availability and integrity. Our study focuses on three basic problems: 1) how users fast and simple timely informed of their data stored in the cloud is intact; 2) if the data is not in good condition, how to restore it; 3) implementation of the key technology to solve the problems in the actual cloud storage environment how efficient. In the cloud storage environment, the data is usually not completely stored in a node, but is split into multiple slices, and then randomly stored on a number of storage nodes. This method is similar with paging memory management, can effectively reduce the file size of node storage fragments do not bring, to improve the utilization of storage space, and can

prevent the node attack causes the user's data all leaked. However, the fatal flaw of this method is that a node is damaged and the loss of the fragmentation can lead to a large number of files are not complete. There is a need for the data redundancy slice can tolerate a certain proportion of the loss, so even if some nodes collapse slice in the Qing, the storage node involved can still access the file.

Fig. 1. Cloud storage data security structure.

Fig. 2. Now has taken the cloud security measures.

2. Secure storage scheme based on Tornado code

2.1. Storage model overview

For users, data confidentiality is one of the most popular topics in the field of cloud storage. Data is stored in a remote server, and the user has no actual control over it. In the entire storage life cycle of the data, the user cannot monitor the behavior of the cloud service provider. Whether external intruders, or internal service providers, are likely to have a threat to the confidentiality of the data. For cloud service providers, how to ensure the availability of data is essential, which is directly related to their business interests. Part of the storage node power, machine failure, and network failure cannot be caused by the collapse in anti-fed or external invasion of external factors such as the data is incomplete or incorrect the case can still continue to provide correct data access service to users, is the key research content in the field of cloud storage. The above two points can be divided into three questions: 1) how to ensure strong confidentiality of data; 2) how to ensure that when the accident caused the loss of data, the user can still complete data recovery; 3) how to ensure the data when encountering malicious tampering, users can detect and correct the errors of the data. In order to solve these problems [4], we propose a data security scheme based on Tornado code DSBT (data security scheme based on Tornado Code A). Scheme based on Tornado codes as the core,

and the introduction of related technologies in the field of information security, focusing on data confidentiality, data loss and data recovery and tamper recovery exhibition? they will be regarded as a whole, the establishment of an integrated security system. Data redundancy technology is closely related to the underlying storage, which determines the different data redundancy technology. For the data redundancy system, the underlying storage should be distributed storage. The definition of distributed storage is not simply defined as the distribution of the underlying storage nodes, but the file itself to be precut. The file is first cut into pieces, and then the slices are spread to different storage nodes. This storage mode is used by most of the existing storage systems. Cloud storage data model commonly used as shown in Figure 3 and 4.

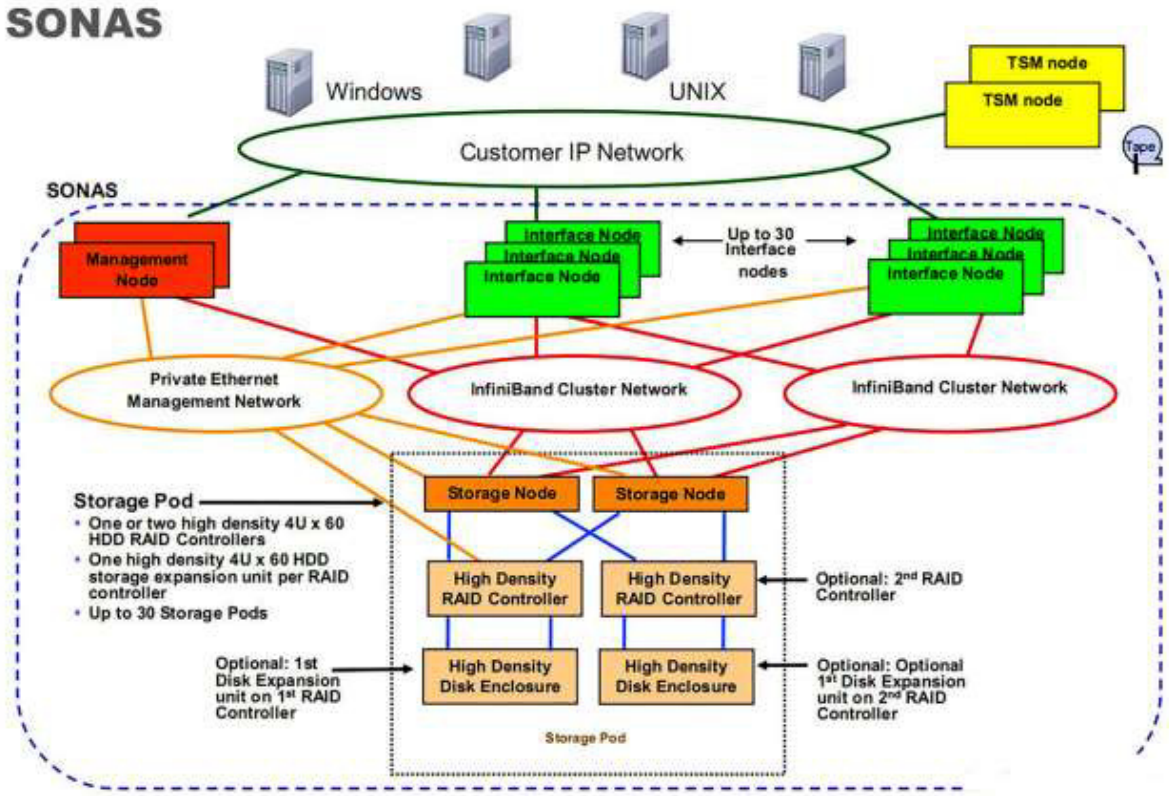

Fig. 3. Universal cloud storage data structure.

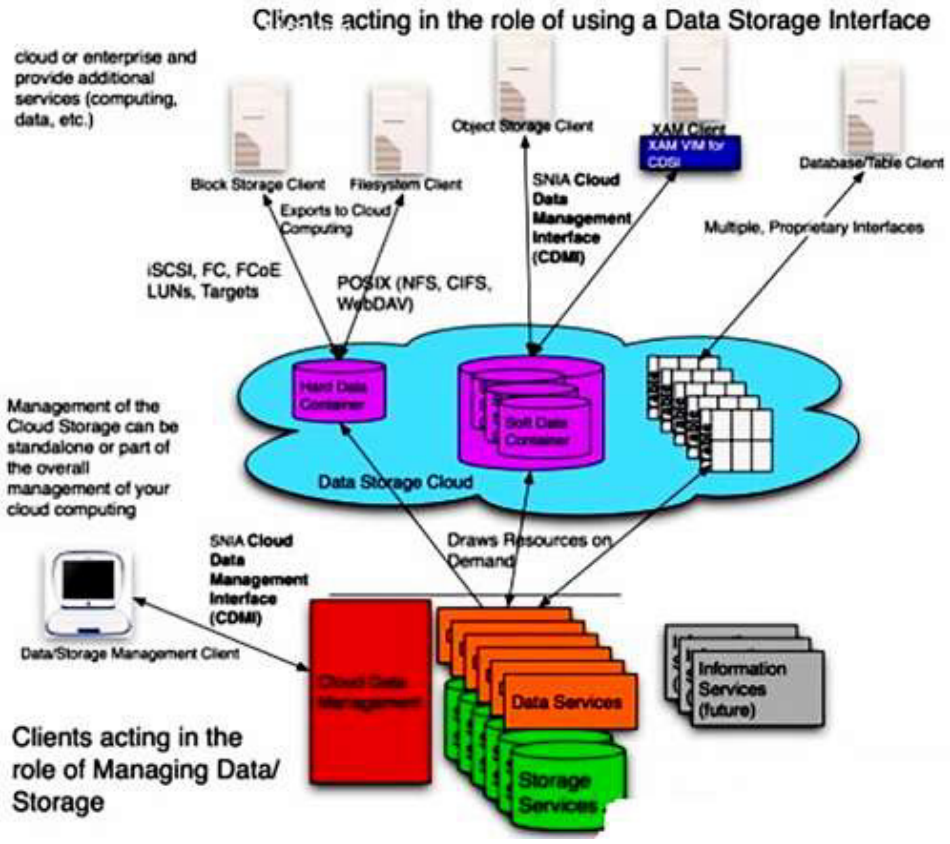

Fig. 4. A common cloud storage data structure for enterprises.

2.2. data confidentially

This section focuses on how to achieve strong confidentiality of the data, that is, in addition to the user's own, no one can access the data of meaningful information, even if it is cloud service providers. In order to achieve this kind of security, the traditional cloud is responsible for the encryption architecture has not been applied. Data in the upload to the cloud before the need for local encryption, and encryption of all users of copper. Non symmetric encryption algorithm is slow, in the face of a large file powerless, it is difficult to achieve the desired speed, so we use symmetric encryption algorithm, the user uploads the data in before using the traditional symmetric encryption technology (such as AES) to encrypt the data. Since the data is encrypted in the local storage to the cloud, you need to be in the download back to the local use, which will involve the process of key. Therefore, the local encryption method is bound to cause the problem of secret preservation. This will lead to the problem described above: users have to keep their own encryption and decryption keys and cannot use the cloud storage, unless the key is also migrated everywhere. From this and the cloud storage is designed, and it is not convenient for users. The boot code is a common password string similar to a mailbox or MSN password, which is equivalent to a compact steel to generate a seed, to generate the desired encryption. The boot code is made up of two parts. The first part is the memory password entered by the user. The second part is the unique identifier of the file name or any other file which is encrypted by the system [5]. So the user can generate different boot password through the same password memory for different files, and then generate different encryption keys, so as to avoid in order setting a different encryption Milang to different files need to memorize a lot of different memory key. Compared to the direct use of

encryption, the use of more easy to remember the password can be used to avoid the management of secret Hu difficult problems, users only need to remember the memory of their input password can be. The advantage of using the boot code is simple and convenient, do not need to make big changes to the system, will not increase the complexity of the system, not only to achieve data confidentiality, but also to avoid the management of dense copper problems. The lack of data confidentiality is no longer entirely dependent on the security of AES, and reduced to the security of the boot password. But the security is not think so unbearable, for a 16 bit long without any social law password, without considering other ways such as poisoning caused by password leak, a common PC using exhaustive attack to get the password needed time to million years. This is acceptable with respect to almost no management problems that are brought about by the direct use of cryptographic dense steel. The key generation function should have two characteristics: first is the same and only as long as the input, the output will have the same characteristics; second is it should be a variable length boot password to encrypt the divergence of fixed length key, the shortest as required for AES encryption key length of 128 bits. Obviously, the hash function MD5 or SHA1 have these two characteristics, can act as our key generating function. Users upload (download) file each time, you need to enter the memory password, assembly into a boot password, through the hash function to calculate the encryption key, and then encrypt the file (decryption). Only the user knows the memory password [6], so even if the cloud storage service provider cannot get the meaningful information. Cloud storage commonly used data encryption algorithm schematic diagram shown in Figure 5 and 6.

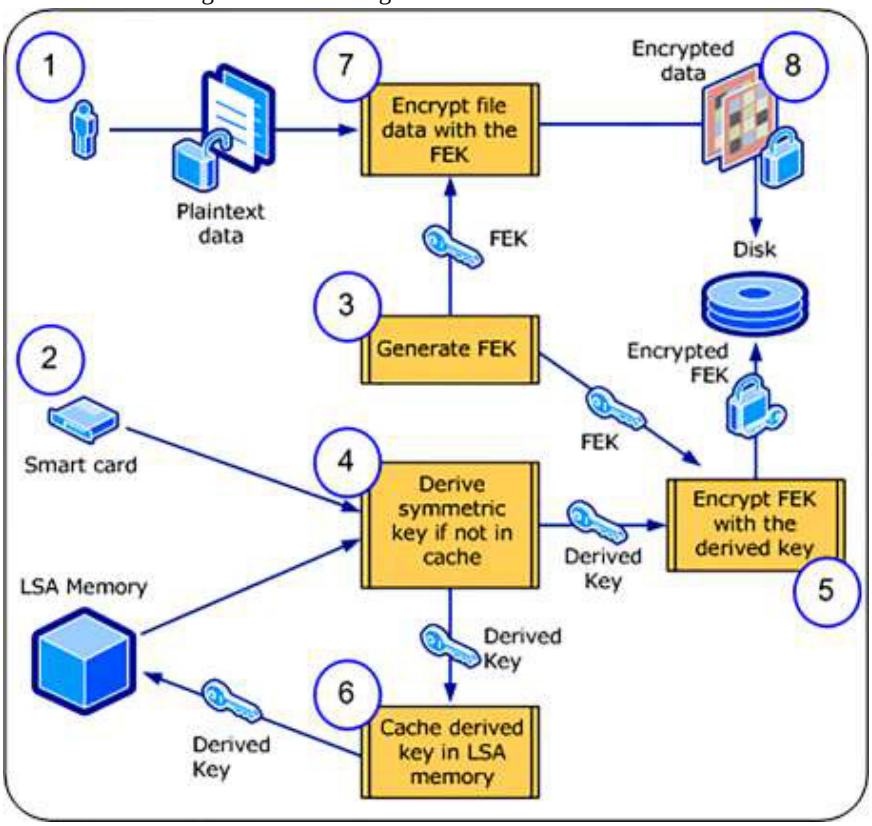

Fig. 5. Sketch of a general data encryption algorithm.

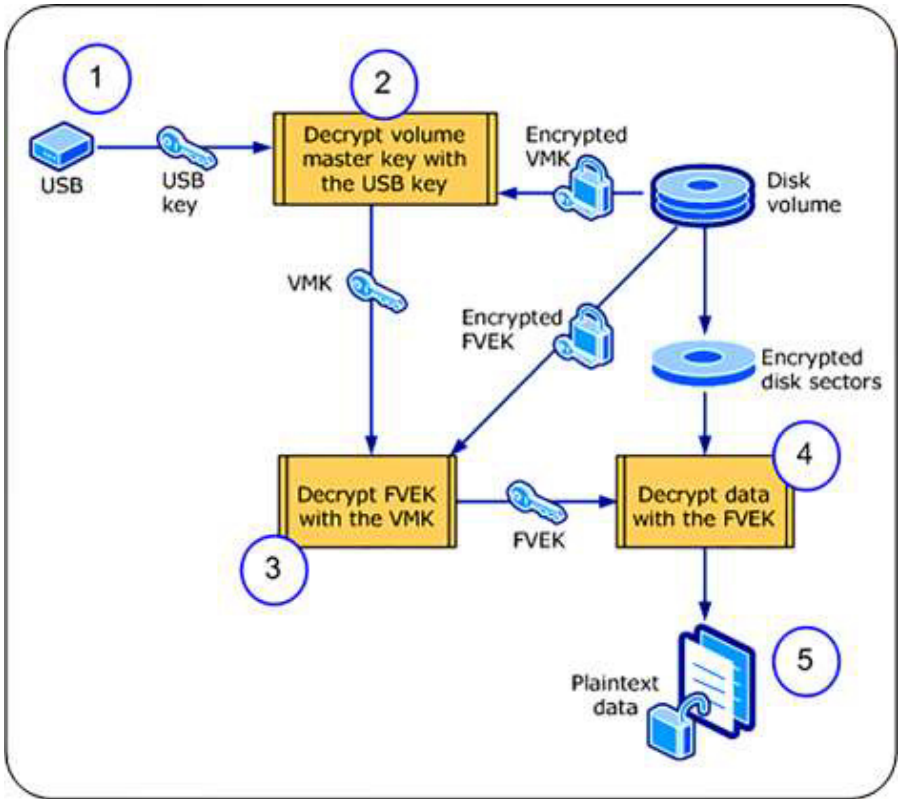

Fig. 6. Schematic diagram of a mobile data encryption algorithm.

2.3. Data loss recovery

Data loss recovery is in fact the data redundancy, which is one of the core contents of distributed storage, and it determines the reliability of a distributed storage system. In a distributed storage system, the storage node for some reason temporarily or permanently leave the system, so the design of distributed storage platform is the most important in the storage node unreliable to achieve reliable storage service. In distributed storage system, data redundancy technology is the most basic method to ensure system reliability and improve data availability and persistence. Through multiple instances of the same data storage file (unit for a file or sheet) to different storage node availability to strengthen data, ensure that even if some nodes are not available, the remaining storage node also has the ability to recover the original data integrity. The system uses a threshold redundancy scheme based on erasure code, which is about to split the original data into M blocks, and generates n (n>m) block coded data blocks according to the redundancy, so that the original data can be recovered as long as the R (n>r>m) block coding blocks are retrieved. And the Tornado code is the first choice for our redundancy strategy with its efficient encoding and decoding speed, low storage redundancy and strong erasure correcting ability. Tornado code is a randomly generated two points diagram based on XOR operation to basic encoding. It is not MDS encoded, but very close to the MDS storage efficiency, which can be considered progressive MDS (MDS asymptotically). Tornado code cannot be like the traditional RS code in a certain degree of error to ensure 100 percent recovery of data, but can guarantee the possibility of a high probability of recovery. In other words, it sacrifices a little bit of security in exchange for high efficiency, which is worth [7], Tornado code 100~10000 times faster than the RS code, depending on the size of the file. Different redundancy strategies have to be set at two points: one is how to create redundant data; the two is how to reconstruct the data when the data is invalid. Popular, that is, the redundant algorithm coding

and decoding algorithm. The following two 8 and 7 show the coding and decoding principle of Tornado code, the left side of the two diagram represents the original data block, and the right side represents the redundancy check block. Tornado code first need to generate a series of cascade two sub graphs, encoding and decoding are based on these plans to complete, about the two-point map generation please refer to the literature. The encoding process is very simple, according to the relationship between the two nodes encoding about sub graph of cascade level and each of the two points Toury, compute the parity blocks of each layer can be. Tornado decoding process is slightly more complex. If a calibration block is connected to a data block, it is clear that the value of the check block is equal to the data block that is connected to it. Therefore, to restore a data block, you can try to find a check block that is only connected to it. We define the degree of a node in the two sub graph as the number of edges connected to the node, and the decoding process is to find a check block, which is connected with the data blocks that need to be recovered. So the decoding process can be regarded as the process of the edge drop in the two sub graph. The following is a detailed decoding process. 1 the complete data block received, according to the XOR operation to the check node connected to it, at the same time from the two maps to remove the variable nodes and the variable nodes are connected by edges. 2 from the new look for a check node degree is 1; the value of the check node is the information node it is connected to the value of the data node is restored. The check node is removed and the corresponding edge is removed. 3 the value of the recovered data node is added to the value of the check node it is connected to, and it is removed from the graph as well as the edge it is connected to. 4 repeat 2 and 3 until all the nodes in the graph have been recovered or cannot find the degree of 1.

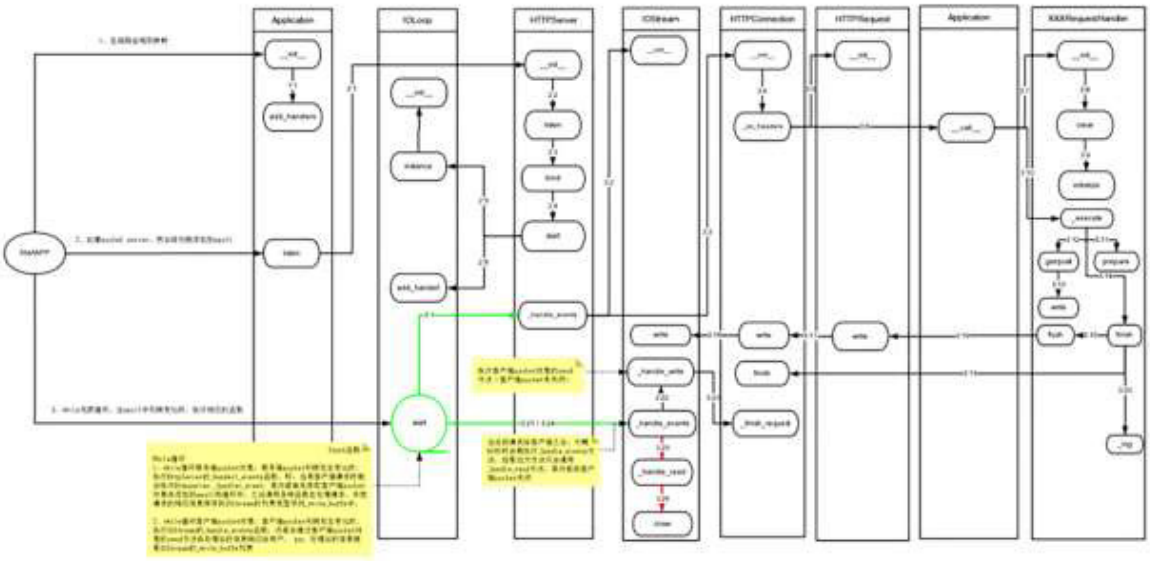

Fig. 7. Schematic diagram of Tornado code.

```
class HTTPConnection(object):

    def _on_write_complete(self):
        if self._request_finished:
            self._finish_request()

    def _finish_request(self):
        if self.no_keep_alive:
            disconnect = True
        else:
            connection_header = self._request.headers.get("Connection")
            if self._request.supports_http_1_1():
                disconnect = connection_header == "close"
            elif ("Content-Length" in self._request.headers
                    or self._request.method in ("HEAD", "GET")):
                disconnect = connection_header != "Keep-Alive"
            else:
                disconnect = True
        self._request = None
        self._request_finished = False
```

Fig. 8. Algorithm core code.

3. Efficient POR system based on trusted log

3.1. POR system description

More and more individuals or companies choose to store data in the third party. In some cases, users use explicit storage services such as Amazon S3 online storage service to back up their data. In other cases, as part of the trend of software as a service (SAAS), the user data is stored implicitly in the third party web site. For example, Salesforce.com web site provides a set of online tools to help manage certain aspects of the company's sales strategy, and store all the submitted sales records. In these applications, the user confirms that the data is still valid and can be retrieved when needed is very necessary. For example, he wanted to make sure that the server did not lose his e-mail or sales records. This is also very important for storage service providers, and users need an audit mechanism to let them know that the data is in good condition, so that they can be at ease. The research content of this chapter is aimed at these scenes, try to find a way [8], the user need to download files can also be convenient and efficient and effective detection of their documents can be retrieved. At the same time, further, looking for a way to use the user does not have to personally detect whether the file can be retrieved, that is, can be detected by others. This system is called POR system. In a POR system, the data storage center must prove to the inspector that it does hold all the data from the user. The proof is to determine whether the data stored on the remote server is in good condition through a kind of similar knowledge proof protocol. The biggest challenge is to build a system that is both efficient and safe, and that the system should be able to ensure that users can take their data from a storage system that is tested. Juels and lalisli give a formal definition of POR system in a POR system in the literature, if a storage service provider can through the audit, and then certainly can remove the data file from it. A simple solution is to allow the server to send

all the data it has stored for the user to go back, but the audit method because of its high cost and no value. The schematic diagram of POR algorithm is shown in Figure 9.

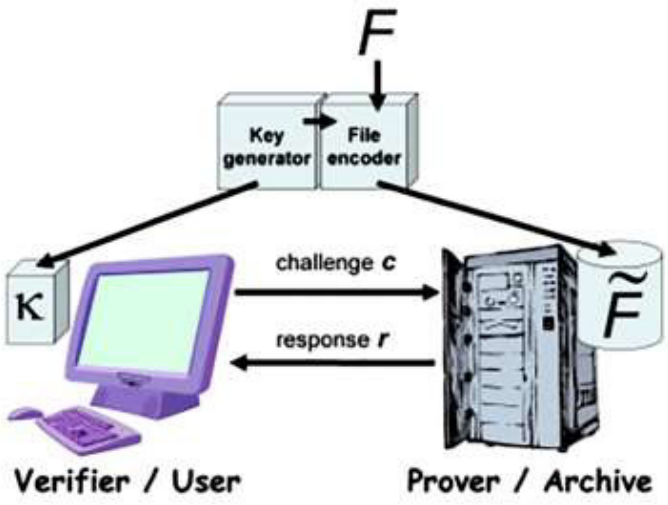

Fig. 9. Schematic diagram of POR system

3.2. System security model

As with most POR or PDP systems, the model of this paper also introduces an additional third party, as a trusted entity, to help users to carry out the retrieval of documents. So the security model of the system becomes the three party models from the two parties of the cloud and the user side. The system model contains three roles: cloud storage service providers, cloud storage users and a trusted third party verifier. 1 cloud: cloud storage service providers, to storage as a form of service allows users to store data storage resources; the cloud need to ensure the integrity and availability of user data, but in this model, it is not completely reliable [9]. 2 users: to buy or hire a cloud storage service provider's storage services, the data is stored in the cloud, is the real master of the data; in the model, the user is considered credible. 3 trusted third party (TTP): used to replace the user's data on the user can retrieve the audit, while providing customers and cloud storage service providers can also allow users to convince the audit results. In this model, the user and the third party verifier is credible, the only threat from the cloud storage service provider. Users and service providers to fully trust third party audit results, the third party will not falsify tun the results, at the same time it provides audit results can also resist the attack from the other or tampered with. The core work of the system consists of two parts: data can be retrieved to detect and generate audit F1. Can retrieve the detection results will be store in the form of trust in the cloud Zhi yen, for user authentication and access. In the traditional POR system, only to explore the retrieval algorithm, and no detailed study of how to feedback the results to the user. The framework of this paper uses the form of a trusted log; the TTP audit results are stored in the cloud. In this way, the user to verify the results of the process, only need to deal with the cloud, does not need to carry out data transmission with TTP. At the same time, TTP itself does not need to burden the storage of audit results, and reduce the burden of TTP storage. Two provides a convenient trusted log: 1 public audit that is to master the TTP public steel case, any party (including other users) can see through, and verify the integrity of the trusted log file learned. Therefore, even if the POR algorithm itself does not have the characteristics of public audit, the trusted log can make up for this defect. 2 offline audit, the user does not need to personally conduct audit, only when needed, was on the line to see the file integrity test results. Algorithm storage, challenge and response constitute the data can retrieve the audit mechanism [10], the algorithm generates F1 and log verification constitutes the audit P1 Chi

generation and verification mechanism. Therefore, the data retrieval of audit system can achieve two modules: data retrieval of Tun meter module and audit module of yen. This module is divided into the implementation of the architecture to provide significant flexibility. In the design of a more efficient POR algorithm only need to replace the data can retrieve the audit module. The security model of the system is shown in Figure 10.

Fig. 10. A common data security model

3.3. Audit log module

Data can be retrieved from the audit module output results will be stored in the form of a trusted log in the cloud. As one of the two core modules of the system architecture, the R will play an important role in the system. It provides a gauge function and file Tun audit results view function separation mechanism, in this mechanism, the trusted third party will bear as little as possible to the audit function without the need to take charge of the store and supply plan, make the design of lightweight trusted third party become possible. Users directly from the cloud to obtain audit results, without the need to communicate with TTP. The audit F1 blog module includes two parts of log generation and verification of H records. The following first describes the log data structure, and then introduces the log generation, update and other operations, as well as R chi. Is 11 ", according to the results of the audit can be retrieved to generate the log, each corresponding to an audit results, the same file of all L five will form a log chain. 1< below by six field components, where res/r is the holding of the examination results[11], the results for the 1 file intact, the result is 0 Ume for the log file that is damaged; the generation time, identifies the Tun meter time, at the same time to ensure the freshness of the log entries and cannot be copied; Eid the identifier entries; FI will mark prev_eid for the same file before a corresponding plan for the formation of Tun, sign is using TTP log list; the key to the front field with RSA signature. LH is always equal to the latest LE: an integer, the initial 0, after each detection plus 1, so e/t in the iLif is always the biggest that e/d. If the audit log as a list, Zif is similar to the operation of all head nodes, always start at the beginning and similar nodes, the list pointer, it will log all by series. The new log is always inserted in the back of the head, and has the biggest assumption of a log of efcf for e, then it's next log of eW for E-1, it's the first log of Eid is e+l. The schematic diagram of the audit log module is shown in Figure 11.

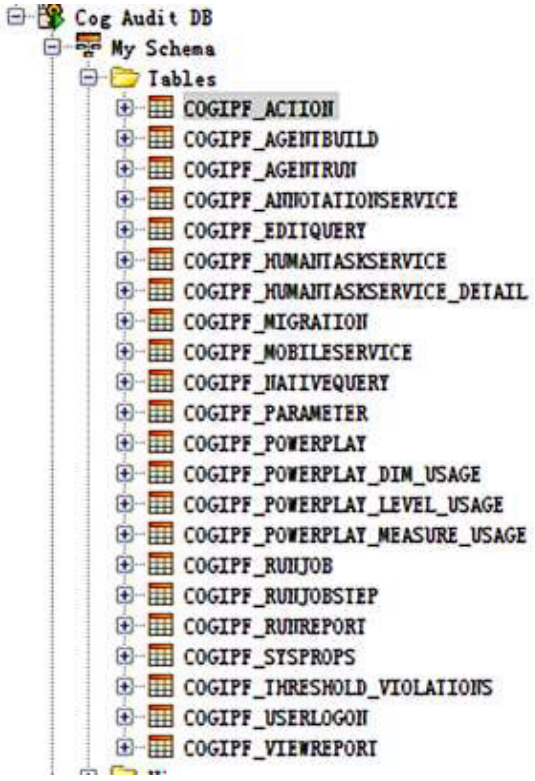

Fig. 11. Schematic diagram of audit log module

4. Design and implementation of prototype system

4.1. Cassandra software profile

Storage platform is the infrastructure of the system, and its design is one of the most important parts in the system implementation. According to the second chapter summary storage model description, storage platform first if distributed; then it is access speed should be high; finally, should be able to deploy a simple, simple extension, simple backup. Integrated the above requirements, the system uses Cassandra as the bottom of the distributed storage database. Cassandra is a mixed type of non-relational database, the main features are distributed, and Column based structured and high stretch. The main feature of Cassandra is that it is not a database, but by a database node constitute a distributed network service, a write operation on the Cassandra, will be copied to other nodes, the read operation of Cassandra, will be routed to a node to read it. Cassandra cluster has no center node, and each node has the same status. The architecture of the. Cassandra system is based on DHT (distributed hash table) of the complete P2P architecture, with the traditional database cluster based on Sharding compared to add or remove a node Cassandra is very convenient, no need to stop the cluster, there is no need to transfer the data manually, very suitable for node stability low compared to the scene. In consistency, availability and network partition tolerance (CAP) compromise on the issue, Cassandra also provides a more flexible strategy [12]; the user can specify different strategies, such as bias strong consistency or availability bias. Cassandra cluster has no center node, and the status of each node is exactly the same. They are maintained by a protocol called Gossip. By Gossip, each node can know which nodes in the cluster, and the state of the node, which makes each node in the Cassandra cluster can complete any key routing, every node is not available would have disastrous consequences. The Gossip algorithm is also called anti entropy, the entropy is a concept in physics, on behalf of the anti-entropy is out of order, and seek out of

order, which fully shows the characteristics of Gossip: in a bounded network, each node randomly and other nodes, through a communication out of order. Eventually all the nodes will reach agreement. Each node may know all the other nodes, it may only know a few neighbors, as long as these nodes can be connected through the network, and ultimately their state is consistent. Gossip is a redundant fault-tolerant algorithm, and further, Gossip is a final consistency algorithm. It cannot guarantee that all nodes are in the same state at a certain time, but it can be guaranteed that all nodes are consistent in the "final time". "The final" is a reality, but it cannot be proved in theory. Therefore, Gossip is not suitable for high consistency requirements of the scene The system ultimately include three roles, users, cloud, and TTP. The DSBT scheme and Realization of POR system based on trusted yen will depend on the coordination between the different roles. But DSBT only needs the user and the cloud, does not require the participation of POR, and TTP systems need to participate in the three parties. DSBT programs and POR systems need to be split into different parts, deployed in different roles. In addition to the core part of the DSBT and POR systems, the system should also provide other basic services, to ensure the normal operation of the core system. The common data structure of Cassandra is shown in Figure 12.

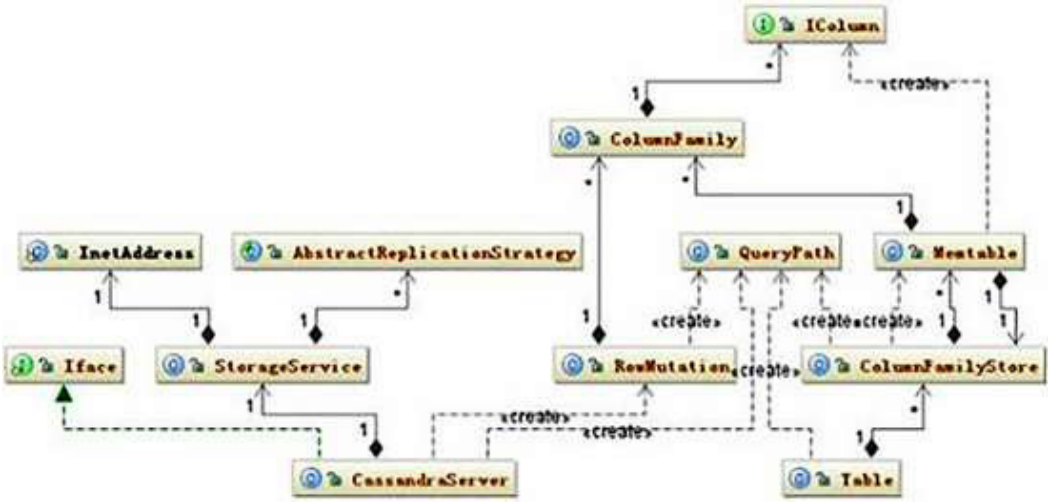

Fig. 12. Cassandra common data structure

4.2. Module design

Subsystems in the system, such as the DSBT scheme or the POR system based on the trusted log, are not only running on one of the roles, but covering two or even three characters. They span multiple roles, and rely on a unified communication interface to coordinate the operation of each other. So the module of the system is not based on the role of physical segmentation, but based on the fact that the subsystem of the logic of the segmentation. The module design of the system is based on the following principles. (1): the system should first meet the backward compatibility and closed principle: modify the closed to open for extension. It should not be a static system, but should have a high openness, when there is a better plan in the future, such as better redundant algorithm or better POR algorithm, the system can easily be extended. In order to achieve this, we first abstracted the interface of each module, and then aimed at the interface instead of the implementation of the program. This can reduce the maximum degree of the module between the call, after which one of the modules to replace, will not affect other modules [13]. (2) using a unified data format: only the design of the interface and cannot completely meet the low said, because across the different subsystems, such as data redundancy from the POR system will still have some affinity. In essence, the system is designed around the data, the data processing and protection is the core idea of the system. Therefore, in order to further achieve the low call between the different subsystems, we designed a unified data interface, which provides the input and output data structure for each seed system. We use the file split to pass the data, the design is based on the status quo: almost all of the redundant system and POR system are to be treated as a

unit. (3) the protocol stack form: protocol stack refers to the sum of the protocol in the network, the image of the reaction process of files in a network transmission by upper layer protocol agreement in the end, and then from the bottom to the upper protocol agreement, similar to a stack. The key module is designed in this form, the user uploads a start down the layers of processing, from the upper module output; users download files is a reverse process, from the beginning of a layer to the bottom module for processing, file. Of course, on both sides of the module stack height is not always the same, can upload more a part of treatment, and may also download more a part of treatment. This module design form and system of the actual work flow more perfect consistent and, at the same time also can be relatively good vertical or horizontal compatibility. Module stack is similar to a filter channel, the channel is the module list, the file is uploaded or downloaded to go through it, and get the final output. As for what module, as well as the module inside the algorithm is what is not a document of concern. Linked list module is not always the same, you can always replace a module, and you can always replace the implementation details of a module. (4) to focus the public service into modules: in each module, there is a large number of repeated but necessary behavior. This part of the system will be repeated to extract the behavior of the package into a module, as the system's underlying public libraries to other modules to call. Three main public services are: unified data access interface, unified communication interface and unified security library interface [14].

4.3. Implementation of module

The paper does not intend to describe all the modules in detail, but only describe some key modules, such as the public module or modules directly related to DSBT and POR system. (1) a unified data access module: the project focus is data redundancy algorithm and POR algorithm, and therefore not too much attention to the underlying storage platform, the bottom using a single Cassandra database set up, without the use of virtualization technology to different storage devices connected to. But we still need to design a unified data access layer, in order to avoid each; one using the database module is repeated writing and database interaction. (2) security algorithm library: as a focus on the safety of the system, there must be a large number of internal security algorithm is used. In order to achieve the purpose of reuse, we unify the security algorithm in a module, for each module call. Part of the security algorithm library from the original JAVA security API (mainly security package) package, the other part is the open source security algorithm; the last part comes from its own implementation. (3) data redundancy module: the data redundancy module contains two modules: data redundancy encoding and decoding module. Because the encoding and decoding are usually the correspondence, the data after the data redundancy algorithm code can only be restored by the decoding algorithm of the algorithm. (4) 4 POR modules: the same data redundancy module, POR module also contains a number of modules: POR data processing module, POR response module and POR verification module. The reason also is the same as the data redundancy module, these modules often have the corresponding relations; the module can work together under the same kind of POR algorithm. Module overall still uses the simple factory design pattern, it brings the advantage is, using the same POR algorithm name can generate the same POR instance [15]. (5): the log module contains two log generation and verification module. Log part of the system is relatively stable; there will not be a variety of R will appear, so only for them to define the public interface, not using the factory method. The log generation module is deployed on the TTP, which is directly related to the verification module. The parameters received by the log generation module are the unique identifier of the file and the audit results of the file.

5. Conclusion

Cloud storage in the rapid development at the same time also brings a series of negative issues, especially data security issues, which seriously hindered the further extensive application of cloud storage. In this paper, in-depth study of issues surrounding the detection can retrieve the data availability and data, first introduced the research status at home and abroad related to the problem, and compares and analyzes the advantages and disadvantages between them, where a detailed analysis of the shortcomings of the current research in this field and can be improved. Then aiming at the problem of data availability and data detection, the corresponding solution method is proposed, which is based on DSBT scheme and POR system based on trusted log. DSBT adopts more efficient and

reliable erasure codes as the core model, starting from solving data confidentiality, supplemented by related cryptographic techniques in the field of information security, the establishment of a can effectively ensure data confidentiality and availability of the system; the POR system based on trusted log is based on the traditional POR system, trusted the log, and combined with DSBT scheme, trusted architecture to achieve lightweight, computational efficiency but also enhance the quality, reach a constant level. At last, it introduces the implementation of DSBT system based on trusted log and POR system as the core of the cloud storage prototype system. The system uses a simple three party security model, with Cassandra as the underlying distributed storage platform; the subsystem needs to carry out the logic module. It focuses on the design concept of DSBT and POR module and the interaction between them. The key part of this paper also gives a detailed flow chart and interface diagram. System performance test is also put in this part, first introduced the test parameters and the environment, and then for each function of the program design test case, the final result analysis.

Acknowledgements

Acknowledgements and Reference heading should be left justified, bold, with the first letter capitalized but have no numbers. Text below continues as normal.

References

[1] Huang Ruwei, Gui Lin, Yu Si, Zhuang Wei. Cloud environment in support of the privacy protection can be calculated encryption method [J]. Journal of the computer. 2011 (12).

[2] Mao Jian, Li Kun, Xu Xiandong. Privacy protection scheme in cloud computing environment [J]. Journal of Tsinghua University (NATURAL SCIENCE EDITION). 2011 (10).

[3] Lv Zhiquan, Aman Chang, Feng Dengguo. Cloud storage access control scheme [J]. computer science and exploration. 2011 (09).

[4] Sun Guozi, Dong Yu, Li Yun. Data access control of cloud storage based on CP-ABE algorithm [J]. Journal of communication. 2011 (07).

[5] Xu Jian, Zhou Fucai, Chen Xu, Zhu Zhiliang. A data outsourcing authentication model based on authentication data structure in cloud computing [J]. Journal of communication. 2011 (07).

[6] Hong Cheng, Aman Chang, Feng Dengguo. An efficient and dynamic cipher text access control method for cloud storage [J]. Journal of communication. 2011 (07).

[7] Zhang Fengzhe, Chen Jin, Chen Haibo, Zang Binyu. Calculation of the cloud data privacy protection and self destruction of [J]. computer research and development. 2011 (07).

[8] Hou Qinghua, Wu Yongwei, Zheng Weimin, Yang Guangwen. A method of protecting the privacy of user data cloud storage platform [J]. Journal of computer research and development. 2011 (07).

[9] Side root Qing, Takamatsu, Shao Bilin. For cloud storage distributed storage security architecture [J]. Journal of Xi'an Jiao Tong University. 2011 (04).

[10] Feng Dengguo, Aman Chang, Zhang Yan, Xu Zhen. Research on the security of cloud computing [J]. software journal. 2011 (01).

[11] Zhao Chunhong, Liu Guohua, Wang Ning, He Lingling. The integrity detection scheme of [J]. micro computer system of text data in the outsourced database model. 2010 (09).

[12] Xian cranes, Feng Dengguo. Research and development of integrity detection scheme of [J]. computer in the outsourced database model. 2010 (06).

[13] Aman Chang, Hong Cheng, Chen Chi. A server transparent outsourcing database query verification method [J]. computer research and development. 2010 (01).

[14] Tan Shuang, Jia Yan, Han red. Data integrity verification of cloud storage in [J]. Journal of computer research and progress. 2015 (01).

[15] Wang Yuding, Yang Jiahai, Xu Cong, Ling Xiao, Yang Yang. Research on the access control technology of cloud computing [J]. Journal of software. 2015 (05).