A Distributed Load-Based Big Data Security Management System

Xie Ming Chen Zubin
Information Department of Guangxi Power Grid Co., Ltd.

Key words: cloud computing; big data; security management
Abstract: The continuous growth of multi-source heterogeneous large scale data and the continuous improvement of the real time requirement increased the complexity of the data processing brought the new technology challenges to the traditional data security transmission management. This paper proposes a data security management system using multiple server nodes constitute the cloud server cluster and permanent storage and backup files on a network; build the client cluster with multiple user nodes, save a copy of the files which is storaged in the server and to provide users with download service and update service, cluster server to user nodes were control and Arbitration. The system achieves a large data load of two parties in the cloud and the user side. The improved data download and update process can effectively improve the scalability of the system and reduce the cost of the service. The method was applied to the national invention patent and the Guangxi power grid was applied. The average success rate of daily data backup was increased by 0.99%.

Introduction

With the big improvement of Energy Revolution and “Internet +”, big data has significant impact on management model and value function structure, thus it takes widespread study and apply of electric big-data. Guangxi Power Grid Co., Ltd owns excellence data analysis experience based on the customer behavior analysis. Via deeply analysis the character and rules of Customer’s data-flow, we notify customer’s potential consumption, valuable improvement and business level. However, the data of multi-source heterogeneous keeps increasing and requires good performance on real-time transaction handling, which takes complicated difficulties while handling the data and brings big challenge of traditional data security and data transmission.

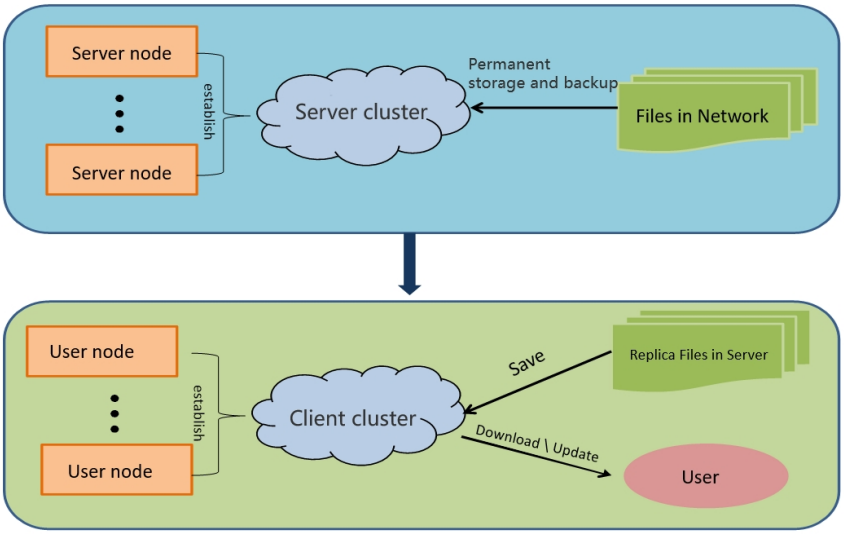
Refer to the traditional data transmission of cloud computing, it is using cloud storage align with the cloud path sequence to transfer huge amount of data accordingly. The advantage of this method is getting rid of the hardware resource limitation. But when we need to backup or recover mass data in a period, and the data reach to hundreds of T, it takes high stress of cloud computing. And once the capacity of cloud storage reach any limitation, it will encounter data overflow and get external attack security crisis. Whereas, such kind of the issue mentioned doesn’t have valid solution till now.

In order to solve above existing issue, this thesis aims to provide a big-data security management methodology and system, using multi-customer nodes to build up the customer client cluster, save the copy of file from server, ensure the data transmission security and deduct the web service loading stress.

Architecture design

This part puts forward the architecture of a management method of big data security, as shown in the graph 1, it uses for distributed storage and access to file data in the cloud platform, its features include: a cloud server cluster using multiple server nodes, for permanent storage and file backup in the network. To build the client cluster, it use multiple users nodes to storage files, and provide download service and update service to user. Server cluster control and arbitration to the user node. The cloud server cluster includes multiple server nodes and server nodes through Chord network interconnection and mutual backup server node responsible for document indexing and backup, arbitration and conflict. The server node also storages the addresses of files in the master client node and the latest version.

Each file corresponding to a client node which is as a virtual server included in the user node, and the master client node is responsible for file Index and Download，the latest version of the file storage and server nodes of the file and download all the node addresses. The master client node and server nodes maintain two-way connection and intercourse information periodically. It also includes downloading nodes and keeps the latest version of file. The download nodes storage the addresses of file system servers and master client node and communicate with master client node periodically, and is also responsible for downloading files. The replica user node storages the copies of the files, the client server nodes and the master node address, the replica user node doesn't provide download service for other user nodes. The replica user node can request for being the download node to the master client node, and the download node also can request for being a replica user node to the master client node. Each file contains a server node, a master client node and numbers of download nodes and copies of nodes.



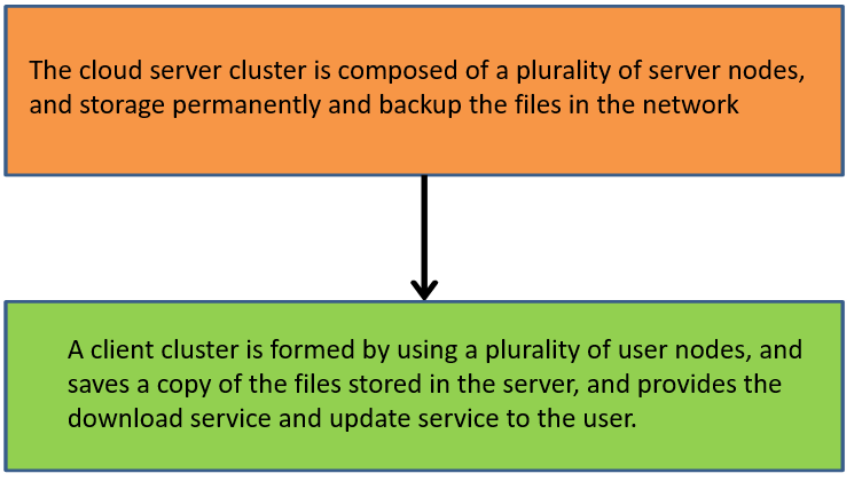
Graph 1 Big data security management framework

Method and System design

This article presents a big data security management system for distributed storage and access to the data files in the cloud platform, as shown in Graph 2, its features include: cloud server cluster module, which is composed of several server nodes for files in the network for permanent storage and backup, and controls and arbitrates the user node. The client cluster module is constructed by a plurality of user nodes to save the copies of files in the servers, and provides download service and update service to the user.

Compared with the existing technology, this system has the following advantages:

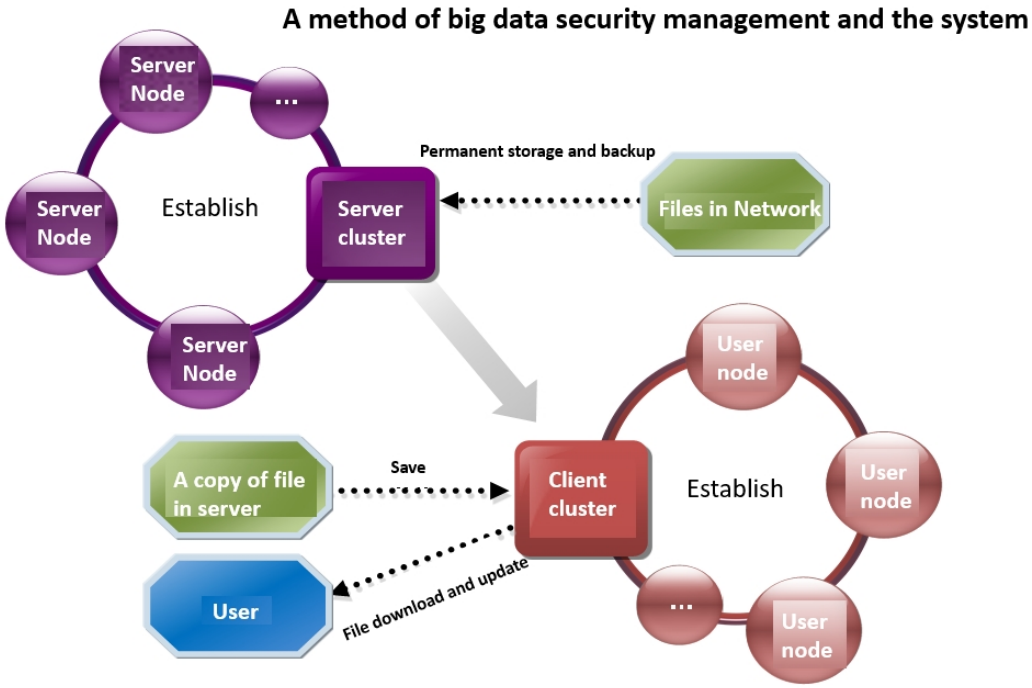
It realized in the cloud and the user side of the two directions of big data load dispersion, improved the efficiency of data downloading and updating in the process and the scalability of system, and reduced the cost of service providers.



Graph 2 the flow chart of method of big data security management

There will be described that the big data security management and prevention in following application cases in this article.

The graph 3 shows the big data security management system data provided in this paper is the safety management system. This system is constituted with the user nodes, and the extension of cloud server cluster. By this way, it can share the running load from cloud to the user terminal, and realize the load dispersion between the cloud and the user side in both directions, and also reduce the cost and improve the scalability of the system. The cloud server cluster which is provided by service provider is reliable and trustable, and it can control and arbitration of nodes in the network. The cluster provides permanent storage and backup for files in the network, so that there will improve the reliability of the network.



Graph 3 Big data security management system

The implementation of the method is as follows:

This system is a bilayer structure, namely the upper layer is a reliable cloud server cluster provided by the service providers and the lower layer is the client cluster built by the user nodes.

The cloud server cluster includes several server nodes, and the server nodes connect to each other and mutual backup through the Chord network. The server node is mainly responsible for files indexing, backup, arbitration and conflict. The server node storages the address of files in the master client server node and the latest version of the files.

The user node includes a virtual server that is master client node: the master client node consists of high processing ability and bandwidth user nodes. Each file corresponding to a master client node, and is responsible for the document indexing and download. The master client node storages the latest version of the file and the corresponding file server nodes and all the addresses of download nodes. The client side and server nodes maintain two-way connection, and the periodic interaction information.

The user node includes download node which is always keep the latest version of the file as a user node. The download node storage the addresses of server nodes and master client node, and it also maintains interaction information and is responsible for the file download.

The user node also includes a copy of itself, and the replica node storages a copy of the file and the addresses of server node and master client node, but the copy of file storage in the replica node may not be the latest version, so it can not provide download service for other nodes. If a replica node often access the file, and need to track the changing of the file, it need to ask the master client node to become a download node. When the replica node no longer needs real-time tracking, it can also apply to convert back to the replica user node.

The four kinds of nodes are the data backup node, all nodes storage the copy of files, and different place is the server node, the master node and all client download nodes always storage the latest version of the files, but the version of files in replica nodes may be expired. A copy of each file has a server node, a master client node and a plurality of download nodes and replica nodes. Among them, the server node is the cloud server provided by the service provider and belong to the upper, and the master client node, download nodes and the replica nodes are the user nodes which belong to the lower.

Algorithm implementation

If the network does not have any of the master client node for this file, then there is no download node and replica node, and the server node will provide download service for user and reset the node as a new client node in the end. Meanwhile, several nodes will connect each other to provide download service and resource to user.

File download process

First, when download files, users only need to connect to the cloud server cluster and find out the server node through dynamic hash table which is responsible for the files, and then get the address of the master client node of the file and request to download service, the master client node will randomly get a part of feedback from its addresses which storage the download nodes, by then, users can download directly by connecting the master client node and the download nodes. When the download is finished, the node will reset to replica node until file is deleted. If there is no any backup node for the files in the network, then means there is no master client node, download nodes and replica nodes, and the server node will provide download service directly to the user, when download is completed, the new node will be set to a master client node.

File download algorithm:

Initialize the LOCATION (Hash Cluster) / / when users need to download files, connect to the cloud server cluster, through dynamic hash table to find node server responsible for the document

Step 1 GET IP (Client Server) / / get the address of the main client node of the file from the server node;

Step 2 CONNECT (Client User) / client master node users connected to the file, download the file request;

Step 3 RESPONSE (Download List Client) / client master node from its storage node to download the address list randomly selected part of feedback;

Step 4 DOWNLOAD (Nodes User) / / users to download by connecting the main client node and the node download;

Step 5 COPYNODE (Copy / Node) //when the download is finished, the node set to copy file node, the node to delete the file.

The download process is basically running in the client cluster, thus it will greatly reduce the load of cloud server cluster, and users can download at the same time by connecting a plurality of nodes, thereby speed up the user's download process. Each node only needs to provide a part of resource for users to download which the load of running is spread on multiple nodes, so as to enhance the load balancing capability of the system. Even if all user nodes that keep the files in the network are offline, users can still get the resources from the server nodes which are belonged to the cloud server cluster, that will ensure further reliability and availability.

2. File update process

When the user needs to update the file, it needs to take different operations according to the type of the node itself. When the download node users need to update the file, they can update directly, because download nodes are always storaged with the latest version of the file. And it will update itself with the operations, the information of latest version and the hash value to the master client node. The node will first check the replica node information and the hash value, if it is a copy of the latest version, the hash value is correct and the update operation rightful, then perform the update, otherwise, it will refuse to renew the update request. After this, master client node will sent the update result, the latest version information and the hash value to the server node and all the download nodes, exclude the replica nodes. In addition, when the replica node users need to update the file, and the copy of file may be expired, users need to verify the version information of the master client node first, and once the master client node receives the request, it will compare the number of file version. If it is an outdated version, then return the latest version. After the replica node validated the version information, it will perform the same steps with download nodes

File update algorithm:

When the user needs to update the file, determine whether the type of the node is a download node or a copy of the node;

If need to update the file node for the download node, then the implementation step 1- step 3:

Step 1 UPDATE (Client Version) / direct update file, and update operations and their copy of the version information and the hash value is sent to the client host node;

Step 2 VERIFICATION (Version, Hash) / / client node should check the copy of the version information and the hash value, if it is a copy of the latest version, and the hash value is correct, then the update operation method, perform the update; otherwise, reject the update request;

Step 3 DISTRIBUTION (Download Nodes Client) / client master node update results and the new version number and the hash value is sent to the server node and all nodes without download, send copies of the node.

IF UPDATE(CopyNode ß FileNode) {

REPEAT STEP1 TO STEP3 / / if the need to update the file node for the replica node, step 1 - step 3:}

Step 4 VERIFICATION (Client Node) / / first to verify the version information of the client host node;

Step 5 COMPARE (Version) / / the main client node receives the check version information request, comparing file version number, if it is an outdated version, the latest version of the file is feedback;

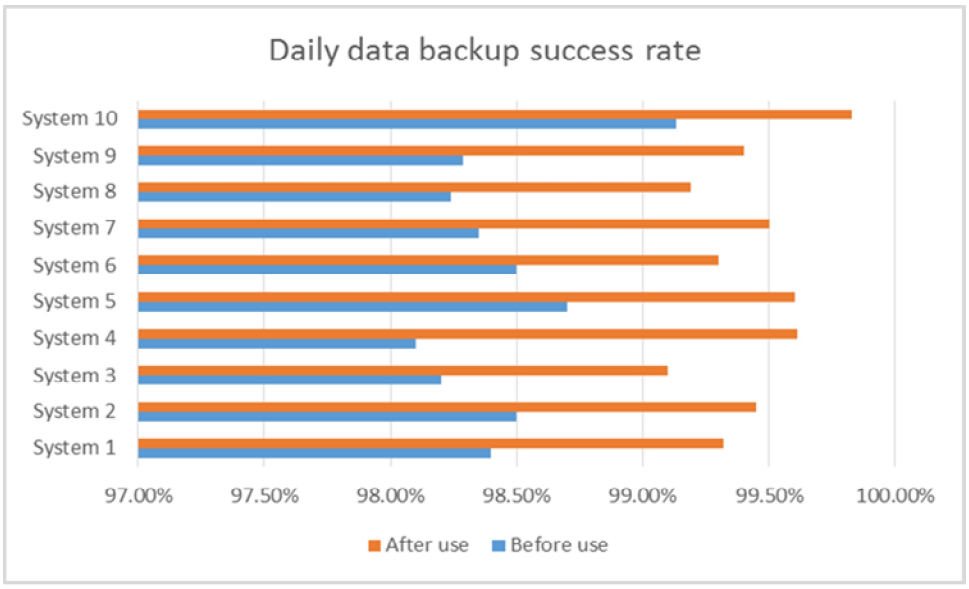
Step 6 UPDATE (Client) / replica node in the execution and verify the version information, download the same node update step in step 1- step 3

By this update method, the master client node only need to send the update information to the master client node which needs to use the files frequently and need to track the changes of files in real-time, in that case, it will not send the update information to the replica node, that will reduce the workload of master client node. Through passive receive updates, a copy of the file can be used directly from the download node, there will avoid frequent accessing to the master client to download files and reduces the traffic for the download nodes. This update method can avoid receiving useless updates information for the replica node, only need to update its own storaged copy from the client before using the file. By way of updating method, it can reduce the load of client clusters and ensure that each node in the using file can obtain the latest version of the file.

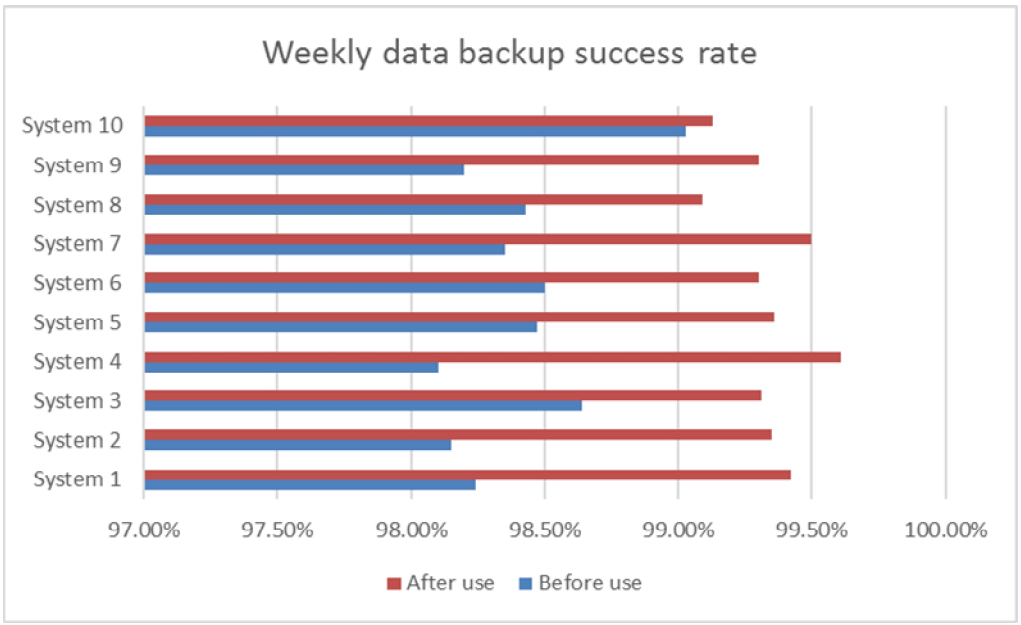
Obviously, because the mutual backup mechanism established between the server nodes, even if the server fails, it can still ensure the file is not lost. In the process of file download and update, the receiving node needs calculate hash value of file, only the hash values are the same, they will be storaged, this method can ensure the security of data transmission.

Experiment comparison

To test the effectiveness of big data security management which mentioned in article, we completed a series of work from September 2015 to October 2015, they included the experiment of success rate of data backup, data recovery rate with the financial system, human resources system, planning system, materials system, construction system, marketing system, cooperation system, GIS, integrated system, 4A system and ect. The whole experiment was according to the systems from 1 to 10 respectively and lasted 7 weeks. Among them, the frequency of data backup is 15 times each day and 7 times each week, and the frequency of data recovery is 4 times each day and 2 times each week, the result is the average value of the experimental data.

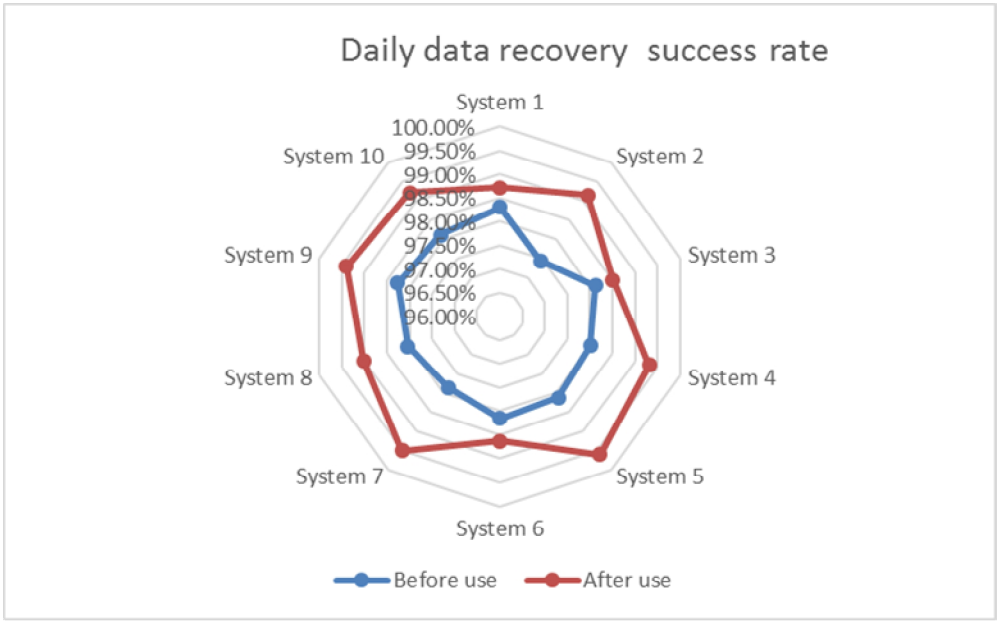


Graph 4 Daily data backup success rate comparison

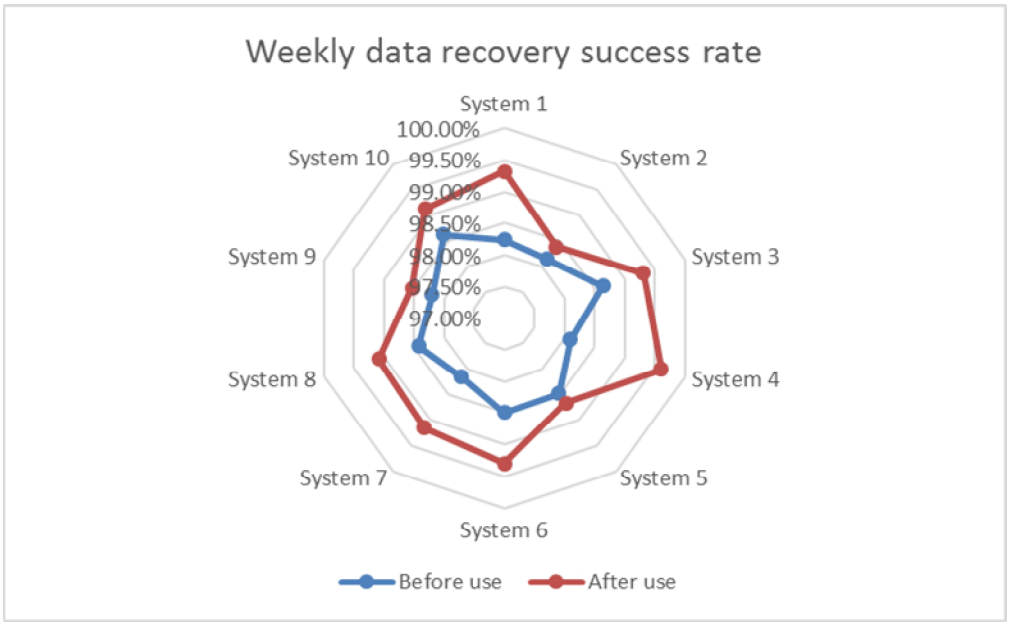


Graph 5 Weekly data backup success rate comparison

Experimental results show that, the average success rate of daily data backup is increased by 0.99%, and the average success rate of weekly data backup is increased by 0.93% by using the proposed the large data security management method and system mentioned in the article.



Graph 6 Comparison of daily data recovery success rate



Graph 7 Comparison of weekly data recovery success rate

Experimental results show that, the average success rate of daily data recovery is increased by 1.07%, and the average success rate of weekly data recovery is increased by 0.70% by using the proposed the large data security management method and system mentioned in the article.

Summary

In summary, this method and system mentioned in article improve the data transmission method based on cloud computing. In the cloud server cluster, which is provided by the service provider, the user nodes are used to construct a client cluster for handling the file download and update while the cloud server cluster is focused on providing reliable index and backup. By using this kind of bilayer structure, it realizes the load transfer from the cloud to the user side, and improves the system's availability and reliability. This method has been obtained the national invention patent in 2016, and there achieves a good effect after implementing the method in

Guangxi Power Grid Co., Ltd, the average success rate of daily data backup is increased by 0.99%. The next step will be applied to the whole power grid company in the field of intelligent security protection of big data security management.

Reference documentation

1. Akshaya Tupe, Amrit Priyadarshi, Data Mining with Big Data and Privacy Preservation, International Journal of Advanced Research in Computer and Communication Engineering , Vol.5, Issue 4, April 2016.

2. Pedro H. B. Las-Casas, Vinicius Santos Dias, Wagner Meira, Dorgival Guedes, A Big Data Architecture for Security Data and Its Application to Phishing Characterization, 2016 IEEE 2nd International Conference on Big Data Security on Cloud (Big Data Security),, Page(s): 36-41.

3. Gupta Palak, Tyagi Nidhi, Digital security implementation in big data using Hadoop, International Journal of Research Studies in Computing, Volume 5 Number 1, April 2016, Page(s):3-9.

4. Victor Changa, Yen-Hung Kuob, Muthu Ramachandrana, Cloud computing adoption framework: A security framework for business clouds, Future Generation Computer Systems , Volume 57, April 2016, Page(s):24-41.

5. Victor Chang, Muthu Ramachandran, Towards Achieving Data Security with the Cloud Computing Adoption Framework, IEEE Transactions on Services Computing,Volume:9 , Issue: 1, Page(s):24-41.

Author introduction

1, Chen Zubin, 1967, male, senior engineer, master's degree, electric power information technology

2, Xie Ming, 1978, male, senior engineer, PhD graduate, electric power information technology