ISSN 2395-695X (Print)
ISSN 2395-695X (Online)

Secure Big Data Storage and Sharing Scheme for Cloud Tenants Using Md6 Algorithm

J.JENIFER ROSLINE[1] , T.KUMARAGURUBARAN[2]
(Anna Univ Affiliated) Master of Engineering, Department of Computer Science and Engineering[1] ,
(Anna Univ Affiliated) Associate Professor,Department of Computer Science and Engineering[2]
Mohamed Sathak Engineering College,
Kilakarai, Ramanathapuram dist., TamilNadu
[1]jenifez20@gmail.com

Abstract: Cloud computing is an emerging technology that allows users to store and process big data for its tenants and uses classical security mechanisms based on encryption techniques that used for protecting big data in the cloud. There are many challenges in the cloud computing in terms of security such as data loss, data breaches, service tracking hijacking, denial of service etc. To overcome the challenges, we present an approach that divides the big data into sequential parts and stores them among the multiple storage providers. Our scheme protects the mapping of various data elements virtually to each provider using a function called trap door function, instead of big data protecting it. Our proposed scheme is efficient and it is secure for the big data of cloud tenants. Our experimental result is verified by the analysis, comparison and the simulation process.

Keywords: cloud computing; big data; storage and sharing; security

I. INTRODUCTION

In modern information technology, big data [1] is a term applied to data sets whose size is beyond the ability of commonly used software systems to store, manage, and process within a tolerable elapsed time. Big data sizes are a constantly moving target, currently ranging from a few dozen terabytes to many petabytes of data in a data center. A data center [2] mainly focuses on the storing and processing of big data sets, real-time data mining, and streaming media delivery etc. Data-intensive applications [3] and research will be integral to many future scientific endeavors, but will demand specialized security mechanisms to make data centers efficient and secure.

In addition, the research community now has the option of accessing storage and computing resources on demand, and the IT industry is currently building multiple big data centers for social networks and applications. Consequently, large amounts of clients' private and secret data (including meta-data) will be stored in data centers, and will need protection during processing and transmission.Thus, data centers should be able to provide efficient security, access, and update mechanisms to not only huge files running into petabytes, but also to small files that are only a few hundred bytes. In all the above cases, determining how to de-sign a secure and efficient scheme for tenants to access their data on the data center storage is crucial. The rest of the paper is organized as follows; in Section II, some existing solutions for data security and related works are described and in Section III, a secure big data protected scheme for cloud tenants is proposed. In Section IV, we give a detailed analysis and comparison among the proposed scheme and other schemes; In Section V, implementation and simulation of the related schemes are designed and realized; we concluded the paper in Section VI.

II. EXISTING SOLUTIONS AND RELATED WORKS

Data centers play an important role in modern information systems which always perform complex computations and retrieve large amount of datasets from data centers. In a distributed environment, an application may needs several datasets located in different data centers and therefore face some challenges such as data security, privacy protection and authentication. In order to enhance the security of cloud client data in data centers, some schemes [4-7] have been proposed, but these schemes mainly focused on designing some algorithms to keep data confidential, these algorithms are always costly and cannot be applied on big data efficiently.

Generally, there are four kinds of conventional security mechanisms to protect data. The first scheme related to the file-level data security[8], which can be implemented on the host. For some applications, this security method can cause performance problems; at the same time, it will introduce some limitations for data backup operations, especially for database backup. In particular, file-level encryption introduces challenges with respect to key management and thus causes low efficiency for data centers. Furthermore, file-level encryption may be inefficient because often only a small fraction of a file will contain in-formation that needs to be protected.The second scheme mainly focused on database-level data security[9], which can be applied when the data stored in a database. This deployment mechanism is known as column-level encryption in that it encrypts the data in each column of the database table. This scheme may be economical for companies who find that all their sensitive data are stored on one or two columns of a database. However, this process will lead to a decline in system performance because the encryption and decryption is generally performed by the software rather than hardware. At the same time, this encryption method must face the challenge that many encrypted data entries may have the same value. The third data security scheme for data centers is media level security technology[10], which is a new method involving static data encryption on storage equipment such as hard disks and tapes. Although media-level encryption provides a high degree of transparency for users and applications, the protective effect is very limited because the data has not been encrypted during transmission. In this scheme, data will be encrypted only after reaching the storage device, and media-level encryption can only guard against the theft of physical storage media. In addition, use of this technology in a heterogeneous environment will require use of multiple key management software, which will increase the complexity of key management, thereby increasing the risks for data recovery. An embedded encryption device is the fourth security scheme for data centers. Embedded encryption devices can encrypt the data sent via them to storage devices, and decrypt the data that is retrieved. Embedded encryption devices are a good solution for point-to-point configurations, but it is difficult to extend, and the overhead is very large.

The final scheme is application-level data encryption technology[11]; this technology is an end-to-end encryption solution. It can ensure that only certain users get to access the data through a particular application. This scheme will be very costly because it must maintain many parameters and data structures.

As the complexity, variety, and popularity of many advanced information services grows, data centers has formed the backbone of these services offered via the Internet including load-hosting, e-commerce, social networking, and a variety of more general services such as software as a service (SaaS), platform as a service (PaaS), and other forms of grid/cloud computing. Of all the advanced network technology, cloud computing is very popular and has been a hot issue recently. The privacy-preserving[12], data security[13], and authentication and encryption technique[14] for cloud computing have been studied recently, these researches mainly focused on using traditional technology to solve cloud security, and it always not efficient for cloud big data.

Related technology and schemes of cloud data storage and security were discussed in Ref. [15] , which focused on aspects for providing security for data storage in cloud and key points for proving security for data storage. In cloud computing, virtualization is the key to provide many of these cloud services, and is being increasingly used within data centers to achieve better server utilization and more flexible resource allocation. However, virtualization also makes many aspects of cloud data center management more challenging; the above cloud services provide high-bandwidth access to cost-effective storage and computing services. However, there are still many key issues which should be studied deeply, especially, secure storage and sharing mechanism for cloud big data.

III. THE PROPOSED SCHEME

In our proposed system, we use MD6 encryption algorithm for the data owner key generation which is highly secured than the existing system. The big data will be separated sequentially into many parts before storage. Then it will be stored on different storage media that was owned by the cloud storage providers. There are two types of data they are public data and confidential data. In order secure the confidential data we use above mentioned encryption technique to secure from the unauthorized access. And then the data are mapped to the cloud by using the cryptographic virtual mapping called Trapdoor function.

In cloud computing, big data storage services represent a basic function for their tenants. In the proposed scheme, firstly, tenants' big data will be separated into many sequenced parts before storage, and then will be stored on different storage media owned by different cloud storage providers. When tenants access their data, the data parts in different data centers will be collected together and then be restored into original form based on the sequenced number of each data part. Generally, the tenants' big data which is stored in cloud storage can be classified into public data and confidential data. There are no extra security requirements for public data, and each tenant can access these data freely, on the other hand, confidential data should always be kept secret and inaccessible to irrelevant persons or organizations.

As described above, the traditional network data security schemes cannot be efficiently applied to protect the cloud tenants' big data. For some threats, especially the security threat of abusing private information and data is fatal for the tenant. Currently, the storing of tenants' data on a cloud platform is a popular practice, and this is becoming more complicated and diversified than ever. In modern advanced information society, people have a variety of personalized requirements about their data

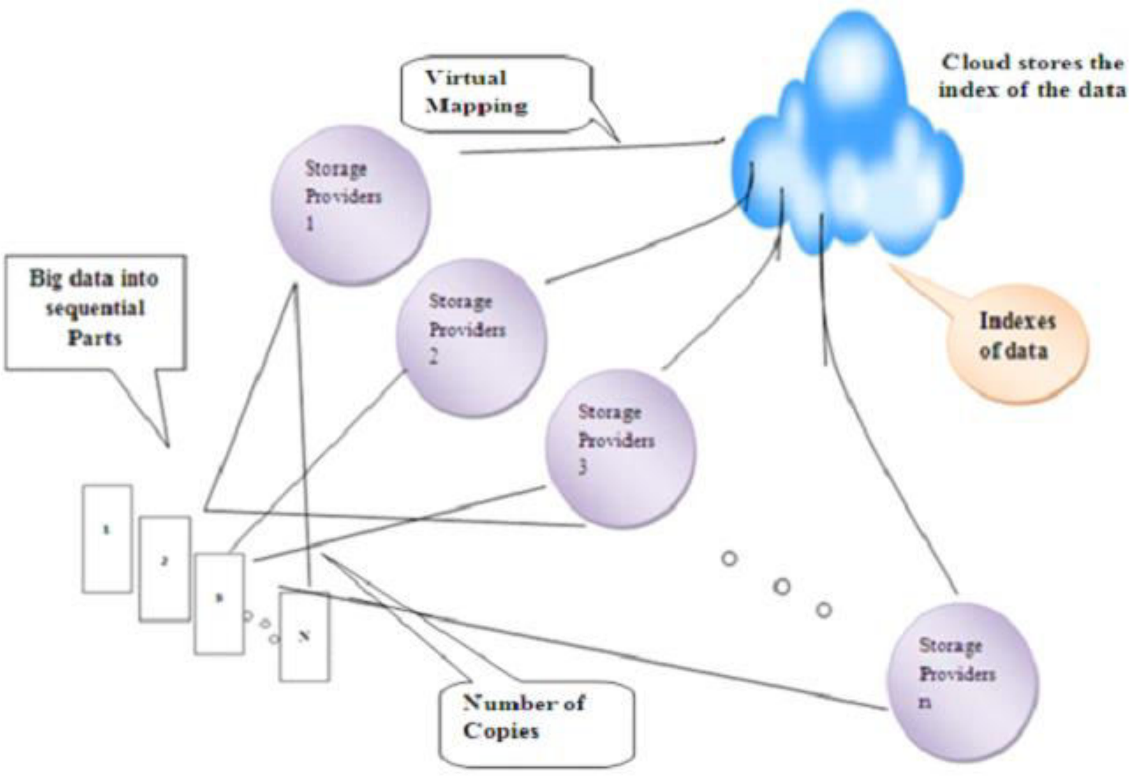

Fig.1 *The proposed Cloud Storage scheme for Tenants' Big Data*

information. Undoubtedly, privacy and security of personal data information is the most important concern for tenants when they store their confidential data on cloud storages.

In order to make the confidential big data of tenants secure, we propose a secure cloud big data storage scheme based on cryptographic virtual mapping of the big data, and the concept is shown in Fig1.

In the proposed scheme, we divide the big data or big data set into sequential data parts according certain principles, such as same data type block or IP-resembled (Internet Protocol) data packets.

In cloud storage of the proposed scheme, the big data of tenants will be separated into a sequence of n parts, where each part can be denoted by *parti* ($i \in (1, n)$) ,and they will be stored at m different storage providers, where each provider is identified as *providerj* ($j \in (1, m)$) . Evidently, n is always far greater than m, these m storage providers belong to different organizations, such as Google, Amazon and Yahoo. Each data part stored on certain cloud storage providers will be allocated to some physical storage media that belongs to the storage provider, so, when big data of a tenant is stored, it will form a unique storage path for the big data given as $Mapping_{Storage\_Path}=\{Data.((P_1(M_1,M_2 \ldots M_r)$ $(P_2(M_1,M_2 \ldots M_S)$; ... $(P_n(M_1,M_2 \ldots M_t)\}$; where, P denotes the storage provider, and M denotes the physical storage media.

We first introduce the trapdoor function before describing the proposed scheme. A trapdoor function' is a function that is easy to compute in one direction, yet believed to be difficult to compute in the opposite direction (finding its inverse) without special information, called the "trapdoor". Trapdoor functions are widely used in cryptography. In mathematical terms, if f is a trapdoor function there exists some secret information y, such that given f(x) and y it is easy to compute x.

In order to protect the confidential big data of a tenant (such as A) from unauthorized access, because big data are always enormous and impossible to encrypt them as a whole, we only need to encrypt the storage path of the big data, and then we can get a cryptographic value which can be called cryptographic virtual mapping of big data. Instead of protecting the big data itself, the proposed scheme protects the mapping of the various data elements to each provider using a trapdoor function. For some special application situation, the proposed scheme will encrypt some key data parts to enhance the data security. At the same time, the proposed scheme will distribute all data parts in different storage service providers, and each provider holds some of the data parts. In order to improve the availability and robustness of big data, the proposed scheme will store more copies for each data on cloud storage providers. The owner of big data will keep the storage index information for each data parts, so, when some data parts on the cloud storage providers lost, we can try to find another copies of the data parts according to their storage index information. Actually, when the proposed scheme store the sequenced data parts of big data into cloud storage providers, certain amount data copies will be stored on different cloud storage providers to enhance the robustness of the big data. As the owner, the tenant will keep the storage path(including data redundant information) secretly, and will only share the path mapping with the authenticated tenants. On the other side, even adversaries can get some data parts, they can not understand the relation with the whole big data, furthermore, the proposed scheme will encrypt some data parts randomly. Generally, the path of the big data is k bytes level and can be stored, processed and transferred very easily. At the same time, in order to protect the confidential big data, the pro-posed scheme will encrypt the path of the big data using trapdoor function, the encrypted result is denoted as *FTrapdoor* (*MappingS torage Path*), and we use *In f oA* to denote the secret information kept by tenant A, and given *FTrapdoor* (*MappingS torage Path*) and *In f oA* it is easy to recovery (*MappingS torage Path*) , actually, *In f oA* is always kept secretly by tenant A.

We have mentioned about that the $Info_A$ proposed scheme will distribute all data parts (some crucial data parts have been encrypted) in different storage service providers, and each provider hold some of the data parts. Generally, individual data part is not so significant for adversaries to learn the privacy of big data. Furthermore, the key parts of privacy have been encrypted already, so, it is even very difficult for all of the providers to collude to recovery the big data because of some encrypted data parts. in fact, the possibility for all providers to collude together is very low for the law restriction and operation infeasibility.

In the proposed scheme, other tenants must be authenticated by tenant A before getting secret information *In f oA* from A. It is easy to share the secret information *In f oA* with other tenants or organizations based on identity encryption algorithm, if someone else, such as tenant(organization) B, wants to share the secret information *In f oA* with tenant A, B must provide his/her identity number ID_B to get the private key k_B calculated and sent by the owner of the secret information *In f oA* based on identity based encryption algorithm, the private key stored in a single storage service provider when tenants directly store their big data on cloud, and this will bring many problems, for example, the storage service error, system crash, the tenants will not be k_B is calculated by identity based encryption algorithm. Afterwards, B can share the secret information *In f oA* with A using the private key k_B. (Note: when A authenticates B using identity based encryption, *In f oA* is the private key of A k_A).

IV. ANALYSIS AND COMPARISON

4.1 Efficiency of the proposed scheme

It is well known that even when we transfer big data locally, the overhead of the transfer-ring big data is often unendurable. Currently, as an efficient and economical data processing and storage service model, cloud computing can provide its tenants with the best data management and service. Recently, many large companies or enterprises focus on some problems related with their data processing and maintenance, these kinds companies or enterprises often spend a lot of time and over-head on their data management, but the effects sometimes are frustrating, especially when these companies or enterprises have not so much business correlation with data management, so, in such situation, these companies or enterprises would like to find a reliable and professional technical institutions which deal with their data storage, management and main-tenance rather than they do it by themselves.

Evidently, big data transmission and its remote storage will cause many problems, such as, the time of processing is too long, the system crashes sometime, and some uncertain system failure. In cloud computing environment, the tenants such as some companies or enterprises, when they transfer and store their big data on cloud storage center directly, it maybe arise some serious problems such as system crash or failure, however, in the pro-posed scheme, the big data of tenants will be divided into some smaller data blocks, these smaller data blocks will be stored in cloud storage media one by one, because these data blocks are smaller than the primitive big data ,they are very efficient for remote-distance data transmission and storage. Under the same network conditions, the transmission failure probability of the proposed scheme is lower than when the big data store and transmit directly. The proposed scheme main focused on the storage and sharing for confidential big data. Generally, cloud tenants will not put their big data on the cloud storage provider when they want to access them frequently. For public big data, it is no need to depend on the proposed scheme because every cloud tenants can access them randomly, on the contrary, the proposed scheme aimed to protect the privacy of confidential big data for cloud tenants and they will not be access. able to access the stored data or the big data lose. In our proposed scheme, the divided data blocks are stored in different storage media.

In the proposed scheme, it is easy to protect the tenants' confidential big data from unauthorized access when the owner of the big data only need to encrypt the storage path of them because it is impossible to encrypt the big data directly in practical applications, and we can call the cryptographic value as cryptographic virtual mapping of big data. It is very efficient when the tenant only encrypt the path of the big data because it is almost in k bytes level. In the proposed scheme, it is easy to share the big data with other tenants by distributing the secret information to other tenants based on some secret sharing scheme, such as identity encryption algorithm.

4.2 Security of the proposed scheme

Now we compute the vulnerability of security of proposed scheme. Let x be an adversary who want to acquire the storage path of the big data illegally, according to the proposed scheme, the adversary can observe the whole big data only when he/she gets the storage paths of all data blocks. So, we can judge the security of storing part of big data on different cloud storages only by computing the probability that x knows the storage paths all of data blocks.

In the proposed scheme, big data will be separated into n parts and these n parts will be stored at m different storage providers. Since no need to consider some extreme situations, we assume that the adversary x can know the storage paths of the parts of big data in probability $p(0<p<1)$. So, in this case, the probability that adversary x maybe achieve all the storage paths of the parts of big data is p^m, and when the probability p is low and the number of providers is large, the probability p^m will be very low. Next, we consider the probability that when we store more than one copy data parts on cloud storage service providers. Similarly, big data of the tenants will be separated into n parts and stored at m different storage providers; we assume the multiple copies of the data parts are q. So, in such case, the best situation for the adversary x to achieve the storage paths of the data parts is each data part has not the same copy on any storage provider,so the probability in such situation is $_p^{n/}$ nq/m, on the other side, the worst situation for the adversary x to achieve the storage paths of the data parts is each data part almost has the same copy on any storage provider, so, the adversary x must acquire all the storage paths of the data parts, and the probability in such situation is p^m as well. From the above analysis, we can conclude that the probability of the adversary x can achieve the all storage paths of data parts is very low in general situation. Now, we can assume that the adversary x will attempt to gain the path mapping file of the big data when it can not achieve all of the storage paths of the data parts. Even this situation is true, the proposed scheme shows excellent performance because it is always impossible for the adversary to get the secret information from the big data owner, the adversary can get the path mapping only after passing the authentication from the owner of the big data. In the pro-posed scheme, we use the trapdoor function to encrypt the storage paths and such trapdoor function is easy to compute in one direction, and almost can not compute in the opposite direction without the secret information which is kept secretly by the big data owner.

4.3 Comparison with other schemes

As more and more big data of individuals, companies and enterprises are placed in the cloud, some concerns are beginning to arise just how safe such kind environment it is. De-spite of there is a lot of promises came from the cloud providers; some tenants are still reluctant to deploy their big data in the cloud. Security is one of the major issues which reduces the growth of cloud computing and complications with data privacy and data protection continue to plague the market. In work [16], a survey of the different security risks that pose a threat to the cloud is presented and the work focused on the different security is-sues and concerns that have emanated due to the nature of the service delivery models of a cloud computing system.

In addition to the classical encryption schemes for protecting data centers mentioned above, various novel schemes[17-18] have been proposed that employ variations on Rabin's n-out-of-m secret sharing scheme, but they all suffer from high computational overhead that will be prohibitive in a big data setting. There are also a number of previous approaches to ensure security of data stored in distributed systems through dispersal of information. Jaatun et al. [19] propose a system for storing data at honest but curious cloud providers, but due to the extensive use of encryption on the storage path between cloud tenants and providers, their scheme is not suitable for big data. Furthermore, their scheme does not provide a convenient way to share data with other users. Spillneretal. [20] present the NubiSave system, which offers a Redundant Array of Optimal Cloud Storage Providers (RAOC) for desktop users; this solution is also not suited to big data. Wang et al. [21] study the problem of ensuring the integrity of data storage in Cloud Computing. In particular, the paper considers the task of allowing a third party auditor (TPA), on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud. The introduction of TPA eliminates the involvement of the client through the auditing of whether his data stored in the cloud are in-deed intact, which can be important in achieving economies of scale for Cloud Computing. Because the third party auditor (TPA) in cloud computing will be complicated and costly, this scheme must solve some important issues before practical applications.

V. SIMULATION

To describe the simulation of the proposed scheme, we will design a procedure to establish communication key between the tenant of cloud big data and other cloud users.

From here on, we use Z_q to denote the group $\{o,...q\text{-}1\}$ under addition modulo q. For a group G of prime order we use G^* to denote the set $G^*=G|O$ where O is the identity element in the group G. We use Z^+ to denote the set of positive integers. We give first some definitions for the proposed key establishment scheme, and the detail of identity based encryption algorithm (IBE) can be referred in [22].

Definition 2.1 A map $\hat{e} : G1 \times G1 \rightarrow G2$ *is called a bilinear pairing if, for all* $x, y \in G1$ *and all* $a, b \in Z$, *we have* $\hat{e}(x^a , y^b) = \hat{e}(x, y)^{ab}$.

Definition 2.2 The Bilinear-Diffie-Hell-man problem (BDH) for a bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ such that $| G1 | = | G2 | = q$ is prime is defined as follows: given $g, g^a, g^b, g^c \in G_1$, compute $\hat{e} (g, g)^{abc}$, where g is a generator and $a, b, c \in Z$. An algorithm A is said to solve the BDH problem with advantage ε if $\Pr[A(g, g^a , g^b , g^c) = \hat{e} (g, g)^{abc}] \geq \varepsilon$ *where the probability is over the random choice of a,b,c,g, and the random bits of A.*

Definition 2.3 A randomized algorithm G that takes as input a security parameter $k \in Z^+$ *is a BDH parameter generator if it turns in time polynomial in k and outputs the description of two groups* G_1, G_2 *and a bilinear function* $\hat{e} : G_1 \times G_1 \rightarrow G_2$ *, with for some prime q. Denote the output of the algorithm by* $G(1^k) =< G_1 , G_2 , \hat{e}, q >$.

Based on the above definitions and IBE algorithm, in Algorithm 1, we designed a procedure to establish communication key between the tenant of cloud big data and other cloud users. From the above definitions, we can conclude that $K_1 = (Y_B)^{XA} \bmod q = \eta^{XA\ XB} \bmod q= (Y_A)^{XB} \bmod q = K_2$.

Algorithm 1. The procedure of establishing communication key between tenant A and other cloud user, for example B

```
Tenant A computes parameter $Y_A$:
    Choose $X_A <q$
    Compute $Y_A = \eta^{XA} \bmod q$ .
User B computes parameter $Y_B$ :
    Choose $X_B <q$
    Compute $Y_B = \eta^{XB} \bmod q$ .
Tenant A encrypts $Y_A$ ,$ID_A$ and $ID_B$ using IBE
   algorithm and then send to B: Encrypt ($Y_A$ ,$ID_A$ and $ID_B$)→B
User B encrypts $Y_B$ ,$ID_B$ and $ID_A$ using IBE
    algorithm and then send to B: Encrypt ($Y_B$ ,$ID_B$ and $ID_A$)→A
Tenant A decrypts message and computes $K_1 = (Y_B)^{XA} \bmod q$ User B decrypts message and computes $K_2 = (Y_A)^{XB} \bmod q$
```

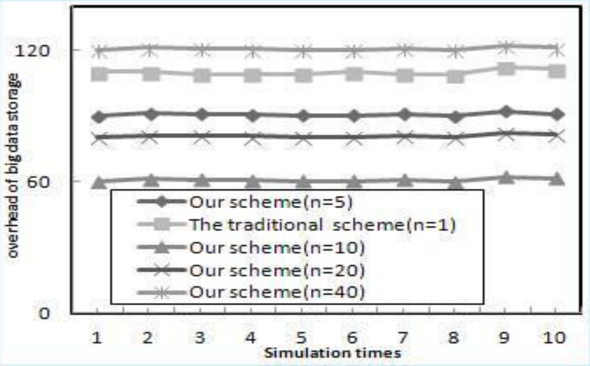

Fig.2 *Overhead of two schemes to deal with big data storage*

Therefore, when any tenant wants to share his/her cloud big data with others, he/she can establish communication key with other cloud user bases on algorithm 1.(2).We construct a distributed storage system based on Hadoop on 20 computers with Federal 10.0, Intel core i5-2400 CPU processor and 4G DDR memory, each computer can provide 100G free hard disk storage space. We decompose a 10G big data into k (k>1) sequenced data part based on programming using Interactive Data Language (IDL). In simulation, the sequenced data parts will store in available cloud storage providers randomly. In general situation, the big data of cloud client is always stored in one single storage service provider; we call it as the traditional scheme.

In simulation, we mainly evaluate the efficiency and data storage robustness of the proposed scheme and the traditional scheme under two different simulation scenarios.

i) In simulation, the overhead of the traditional scheme mainly includes the time of big data transmission from the client to cloud storage platforms, and in the proposed scheme, the overhead includes the pre-processing cost of big data, the maximum transmission time of all the data parts. We test the proposed scheme in 4 different situations (our scheme (n=x)), which x denotes the number of big data being decomposed; and we test each situation of the proposed scheme and the traditional scheme 10 times individually; the simulation results are shown in Figure 2.

ii) Generally, the cloud service providers have certain probability to be failure or deny serving the cloud clients. As the traditional scheme, the big data will lose or be unavailable during the failure period of the cloud storage providers. On the other side, the proposed scheme can be robust because there are more than one copy data parts storing on different cloud storage providers. In the simulation, we test the two different schemes under the five situations that the cloud storage providers fail with a probability of 5%, 10%, 15%, 20% and 25%. The encryption technology in the proposed scheme is identity-based encryption algorithm; and we stored the each data parts in m (m>1) copies on different cloud storage providers in simulation. We assume that the client's big data cannot be available in the cloud when any part of the big data missed in the proposed scheme. In this simulation scenario, we test the two schemes 50 times under each probability and the simulation results are shown in Figure 3.

The simulation results in Figure 2 and Figure 3 coincide in the analysis in Section IV very well; and from the simulation results in scenario 1, we can conclude that when we choose suitable data parts number, the pro-posed scheme can be more efficient than the traditional scheme, at the same time, we can find that when the number exceeds a certain value, the proposed scheme will become less efficient because it need more cost to preprocess the big data. From the simulation results, we find that there is a better balance between the overhead of data partition and data uploading. So, in real application of the proposed scheme, the suitable data size depends on some factors such as data type, network performance and client terminal's computation capacity (concurrency). In order to reflect the real cloud computing situation, we set certain percentage of cloud storage providers (computers) to be off work in simulation. Figure 3 shows the data robustness of two schemes when cloud storage providers fail under different probability, the traditional scheme shows bad performance when the probability of failure is great. While the proposed scheme can achieve wonderful performance when we choose suitable copy of data, the improvement of cloud data robustness of the proposed scheme is very evident. So, in the above two simulation scenarios, the proposed scheme achieve better results and more efficient than the traditional scheme.

VI. CONCLUSION

The size of big data is growing constantly, we need to secure the big data, for this the big data will be separated sequentially into many parts and then it will be stored on different storage media that was owned by the cloud storage providers. When tenants access their big data, the data are collected from the data center by the sequence number of each data part and send to the tenants. Here the data must be stored as sure we develop a scheme called Secure cloud big data storage scheme based on cryptographic virtual mapping of the big data. A trapdoor function is a technique that used for the cryptographic virtual mapping function, which is highly secure for the storage purpose.

.

REFERENCES

[1] Cheng Hongbing; Coll. of Comput. Sci., Zhejiang Univ. of Technol., Hangzhou, China ; Rong Chunming ; Hwang Kai ; Wang Weihong "Secure big data storage and sharing scheme for cloud tenants", IEEE Transactions on, 2015,Communications, China ,Vol:12 , Issue: 6.

[2] D. Kusnetzky. What is "Big Data?" [Online]. Available:http://blogs.zdnet.com/virtualization/?p=1708.

[3] K. Kant, "Data center evolution: A tutorial on state of the art, issues, and challenges," Com-puter Networks, vol. 53, no. 17, pp. 2939– 2965, 2009, virtualized Data Centers. [Online]. Avail-able: http://www.sciencedirect.com/science/article/pii/S1389128609003090

[4] M. L. Norman and A. Snavely, "Accelerating da-ta-intensive science with gordon and dash," in Proceedings of the 2010 TeraGrid Conference, ser. TG '10. New York, NY, USA: ACM, 2010, pp. 14:1–14:7.[Online]. Available: http://doi.acm. org/10.1145/1838574.1838588.

[5] Curran R J, Haskin R L. File level security for a metadata controller in a storage area network: U.S. Patent 7,840,995[P]. 2010-11-23.

[6] Shmueli, Erez, et al. "Database encryption: an overview of contemporary challenges and de-sign considerations." ACM SIGMOD Record 38.3 (2010): 29-34.

[7] Sabahi F. Virtualization-level security in cloud computing[C]//Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on. IEEE, 2011: 250-254.

[8] Popa R A, Stark E, Helfer J, et al. Building web applications on top of encrypted data using Mylar[C]//USENIX Symposium of Networked Systems Design and Implementation. 2014.

[9] Cao N, Wang C, Li M, et al. Privacy-preserving

multi-keyword ranked search over encrypted cloud data[J]. Parallel and Distributed Systems, IEEE Transactions on, 2014, 25(1): 222-233.

[10] Soundararajan O M, Jenifer Y, Dhivya S, et al. Data Security and Privacy in Cloud Using RC6 and SHA Algorithms[J]. Networking and Com-munication Engineering, 2014, 6(5): 202-205.

[11] Singla J S. Cloud Data Security using Authentication and Encryption Technique[J]. Global Journal of Computer Science and Technology, 2013, 13(3).

[12] Spoorthy V, Mamatha M, Kumar B S. A Survey on Data Storage and Security in Cloud Comput-ing[J]. 2014.

[13] S. Subashini， V. Kavitha， A survey on security issues in service delivery models of cloudcom-puting ， Journal of Network and Computer Applications,vol 34, Issue 1, January 2011, pp. 1–11.

[14] A. Shamir, “How to share a secret,” Com-mun. ACM, vol. 22, pp. 612–613, November 1979. [Online]. Available: http://doi.acm. org/10.1145/359168.359176.

[15] Y. Singh, F. Kandah, and W. Zhang, “A secured cost-effective multicloud storage in cloud com-puting,” in Computer Communications Work shops (INFOCOM WKSHPS), 2011 IEEE Confer-ence on, April 2011, pp. 619 –624.

[16] M. G. Jaatun, G. Zhao, A. Vasilakos, A . A. Nyre, Alapnes, and Y. Tang, “The design of a re-dundant array of independent net-storages for improved confidentiality in cloud computing,” Journal of Cloud Computing: Advances, Sys-tems and Applications, vol. 1, no. 1, p. 13, 2012. [Online]. Available: http:// www.journalofcloudcomputing.com/content/1/1/13.

[17] J. Spillner, G. Bombach, S. Matthischke, J. Muller, Tzschichholz, and A. Schill, “Information dis-persion over redundant arrays of optimal cloud storage for desktop users,” in Utility and Cloud Computing, 2011 Fourth IEEE International Conference on, Dec. 2011, pp. 1 –8.

[18] Qian Wang; Cong Wang; Kui Ren; Wenjing Lou; Jin Li. Enabling Public Auditability and Data Dy-namics for Storage Security in Cloud Comput-ing. IEEE Transactions on Parallel and Distribut-ed Systems, vol. 22(5), 2011, pp.847-859.

[19] Boneh D and Franklin M, “Identity-based en-cryption from the Weil pairing” [J]. in Advances in Cryptology, CRYPTO 2001, Lecture Notes in Computer Science, vol. 2139, pp. 213-229,2001.