www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 6 Issue 7 July 2017, Page No. 21889-21894
Index Copernicus value (2015): 58.10 DOI: 10.18535/ijecs/v6i7.03

Encrypted Bigdata Using AES Deduplication in Cloud Storage

N.B. Mahesh Kumar
[1]Assistant Professor (Sr.G), Dept. of Computer Science and Engineering, Bannai Amman Institute of Technology
Erode, Tamilnadu, India
mknbmaheshkumar@gmail.com

Abstract: Benefited from cloud computing, users can achieve an effective and economical approach for data sharing among group members in the cloud with the characters of low maintenance and little management cost. Meanwhile, it must provide security guarantees for the sharing data files since they are outsourced. Unfortunately, because of the frequent change of the membership, sharing data while providing privacy-preserving is still a challenging issue, especially for an untrusted cloud due to the collusion attack. Moreover, for existing schemes, the security of key distribution is based on the secure communication channel, however, to have such channel is a strong assumption and is difficult for practice. In this paper, a secure data sharing scheme for dynamic members was proposed. Firstly, a secure way for key distribution without any secure communication channels, and the users can securely obtain their private keys from group manager was proposed. Secondly, this scheme can achieve fine-grained access control, any user in the group can use the source in the cloud and revoked users cannot access the cloud again after they are revoked. Thirdly, the scheme from collusion attack, which means that revoked users cannot get the original data file even if they conspire with the untrusted cloud was protected. In the proposed scheme, by leveraging polynomial function, a secure user revocation scheme was achieved. Finally, this scheme can achieve fine efficiency, which means previous users need not to update their private keys for the situation either a new user joins in the group or a user is revoked from the group. The results will show effectiveness of the scheme for potential practical deployment, especially for big data deduplication in cloud storage.

Keywords: cloud computing, cryptography, deduplication, dropbox

1. Introduction

Cloud Storage is a model of data storage in which the digital data is stored in logical pools, the physical storage spans multiple servers (and often locations), and the physical environment is typically owned and managed by a hosting company. These cloud storage providers are responsible for keeping the data available and accessible, and the physical environment protected and running. People and organizations buy or lease storage capacity from the providers to store user, organization, or application data.

Cloud storage services may be accessed through a co-located cloud computer service, a web service application programming interface (API) or by applications that utilize the API, such as cloud desktop storage, a cloud storage gateway or Web-based content management systems.

Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate to deal with them. Challenges include analysis, capture, data curation, search, sharing, storage, transfer, visualization, querying, updating and information privacy. The term "big data" often refers simply to the use of predictive analytics, user behavior analytics, or certain other advanced data analytics methods that extract value from data, and seldom to a particular size of data set. "There is little doubt that the quantities of data now available are indeed large, but that’s not the most relevant characteristic of this new data ecosystem."

In cryptography, encryption is the process of encoding messages or information in such a way that only authorized parties can read it. Encryption does not itself prevent interception, but denies the message content to the interceptor. In an encryption scheme, the intended communication information or message, referred to as plaintext, is encrypted using an encryption algorithm, generating cipher text that can only be read if decrypted. For technical reasons, an encryption scheme usually uses a pseudo-random encryption key generated by an algorithm. It is in principle possible to decrypt the message without possessing the key, but, for a well-designed encryption scheme, large computational resources and skill are required. An authorized recipient can easily decrypt the message with the key provided by the originator to recipients, but not to unauthorized interceptors.

The purpose of encryption is to ensure that only somebody who is authorized to access data (e.g. a text message or a file), will be able to read it, using the decryption key. Somebody who is not authorized can be excluded, because he or she does not have the required key, without which it is impossible to read the encrypted information.

2. Related Work

M.Bellare, S.Keelveedhi, and T.Ristenpart proposed that the study of problem of providing secure outsourced storage and it supports deduplication and resists brute-force attacks. DupLESS, which combines a CE-type base MLE scheme with the ability to obtain message-derived keys with the help of a key server (KS), shared amongst a group of clients. The clients interact with the KS by a protocol for oblivious Pseudo Random Function (OPRFs), ensuring that the KS can cryptographically mix in secret material to the per message keys while learning nothing about files stored by clients.

These mechanisms ensure that DupLESS provides strong security against external attacks which compromise the Storage Service (SS) and communication channels (nothing is leaked beyond file lengths, equality, and access patterns), and that the security of DupLESS gracefully degrades in the face of comprised systems The substantial increase in security comes at a modest price in terms of performance, and a small increase in storage requirements relative to the base system. It can work transparently on top of any SS implementing a simple storage interface, as shown by this prototype for Dropbox and Google Drive [1].

In cloud environments [2], users store their data or files in cloud storage but it is not infinitely large. In order to reduce the requirement of storage and bandwidth, data deduplication has been applied. Users can share one copy of the duplicate files or data blocks to eliminate redundant data. Besides, considering the privacy of sensitive files, the users hope that the cloud server cannot know any information about those files. They often use certain encryption algorithms to protect the sensitive files before storing them in the cloud storage. Unfortunately, previous schemes have a security problem. These schemes did not satisfy semantic security. In this paper, a hybrid data deduplication mechanism which provides a practical solution with partial semantic security.

History-Aware Rewriting algorithm (HAR) [3] improves restore performance by 2.6X –17X at an acceptable cost in deduplication ratio. HAR out performs the state-of-the-art work in terms of both deduplication ratio and restore performance. The hybrid scheme is helpful to further improve restore performance in data- sets where out-of-order containers are dominant.

The ability of HAR to reduce sparse containers facilitates the garbage collection. It is no longer necessary to offline merge sparse containers, which relies on identifying valid chunks. A Container-Marker Algorithm (CMA) that identifies valid containers instead of valid chunks was proposed. Since the metadata overhead of CMA is bounded by the number of containers, it is more cost- effective than existing reference management approaches whose overhead is bounded by the number of chunks.

Deduplication is an effective technique for reducing storage costs in distinct storage environments, more specifically in backup, primary, RAM, and SSD storage systems. Although all existing deduplication systems can be classified by a common taxonomy, each storage type has different assumptions that lead to distinct design decisions. Then, as another contribution, the existing deduplication systems and classify them according to the storage type—for instance, backup, primary, RAM, and SSD was presented. Most of these systems assume immutable data and trade latency for deduplication and I/O throughput by mainly using inline deduplication approaches [4]. On the other hand, in primary storage, data is mutable and I/O latency is critical, so the number of inline deduplication systems is reduced and the percentage of offline approaches raised. In RAM deduplication, most systems scan memory for duplicates to avoid intrusive mechanisms for intercepting I/O calls. Moreover, even in backup deduplication where the amount of work is substantially larger, these issues and others, such as reference management, scalability, reliability, and security, can be further improved.

Digital signatures and hash functions already play an important role improvising data security to communication applications. A low-cost GPS digital signature architecture, which combines an optimized GPS algorithm design and an optimized SHA-1design, is proposed for low-cost RFID tags. RFID tags will be integral in the development of emerging mobile and ubiquitous computing applications. The proposed architecture can be used for device authentication to prevent tag cloning and to provide data authentication to prevent transmission forgery [5]. The design offers significant improvements over previous work on RFID digital signature architectures in term so area, power, and timing. The proposed low-cost SHA-1 architecture is based on an 8-bit data path, which results in a significant reduction in area (1200 gates) over previous work for the hash function. The net result is a digital signature architecture that is within close reach of current RFID tag deployment and will certainly be feasible for providing security in tags in the very near future.

Gao proposed that system is enhanced in security. Specifically, an advanced scheme to support stronger security by encrypting the file with differential privilege keys was presented. In this way, the users without corresponding privileges cannot perform the duplicate check. Furthermore, such unauthorized users cannot decrypt the cipher text even collude with the S-CSP (Secure-Cloud Service Provider). Security analysis demonstrates that the system is secure in terms of the definitions specified in the proposed security model. The user is only allowed to perform the duplicate check for files marked with the corresponding privileges. An advanced scheme to support stronger security by encrypting the file with differential privilege keys was presented [6]. Reduce the storage size of the tags for integrity check.

Halevi proposed that, the solution is based on a cryptographic usage of symmetric encryption used for enciphering the data file and asymmetric encryption for metadata files, due to the highest sensibility of this information towards several intrusions. In addition, thanks to the Merkle tree properties, it is shown to support data deduplication, as it employs a pre verification of data existence, in cloud servers, which is useful for saving bandwidth. Besides, the solution is also shown to be resistant to unauthorized access to data and to any data disclosure during sharing process, providing two levels of access control verification [7].

Kaaniche proposed that SecCloud, a privacy cheating discouragement and secure-computation auditing protocol for data security in the cloud. To the best of the knowledge, it is the first work that jointly considers both of data storage security and computation auditing security in the cloud. The concepts of uncheatable cloud computation were defined and privacy-cheating discouragement and proposed SecCloud to achieve the security goals. To improve the efficiency, different user's requests can be concurrently handled through the batch verification. By the extensive security analysis and performance simulation in the developed Sec HDFS (Hadoop File System), it is showed that the protocol is effective and efficient for achieving a secure cloud computing. In addition, this focus on the privacy preserving issues in the above computation. Furthermore, it is planned to implement them in the real cloud platform such as EC2 and OpenStack [8].

Lillibridge proposed that, the issue is coped by adding one additional layer of deterministic and symmetric encryption on top of convergent encryption. This additional encryption can be added by a component placed between the user and the cloud storage provider such as a local server or a gateway. This component will take care of encrypting/decrypting data from/to users. In order to allow the cloud provider to detect duplicates, encryption and decryption are performed with one unique set of secret keys [9].

These pairing-based schemes realize important new features, such as safeguarding the master secret key of the delegator from a colluding proxy and delegate. One of the most promising applications for proxy re-encryption is giving proxy capabilities to the key server of a confidential distributed file system; this way the key server need not be fully trusted with all the keys of the system and the secret storage for each user canal so be reduced. It is implemented that this idea in the context to the Chefs file system, and showed experimentally that the additional security benefits of proxy re-encryption can be purchased for a manageable amount of run-time overhead. The theoretical problem of finding a proxy re-encryption scheme that does not allow further delegations; that is, Bob (plus the proxy) cannot delegate to Carol what Alice has delegated to him [10]. Another challenging problem is to find unidirectional re-encryption schemes that allow cipher texts to be re-encrypted in sequence and multiple times. The practical problems of finding more efficient implementations of secure proxy re-encryption schemes, as well as conducting more experiment atleast in other applications was described.

Data deduplication is one of important data compression techniques for eliminating duplicate copies of repeating data, and has been widely used in cloud storage to reduce the amount of storage space and save bandwidth. To protect the confidentiality of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing. To better protect data security, this paper makes the first attempt to formally address the problem of authorized data deduplication. Different from traditional deduplication systems, the differential privileges of users are further considered in duplicate check besides the data itself. The several new deduplication constructions supporting authorized duplicate check in a hybrid cloud architecture was proposed. Security analysis demonstrates that this scheme is secure in terms of the definitions specified in the proposed security model. As a proof of concept, a prototype of the authorized duplicate check scheme and conduct test bed experiments using this prototype was implemented.

3. Proposed Work

In existing scheme based on data ownership challenge and Proxy Re-Encryption to manage encrypted data storage with deduplication was explained. It aims to solve the issue of deduplication in the situation where the data holder is not available or difficult to get involved. Meanwhile, the performance of data deduplication in this scheme is not influenced by the size of data, thus applicable for big data.

It is implemented that the proposed scheme and tested its performance. It is also describing the implementation and testing environments. A MySQL database is applied to store data files and related information. It is not taken into account the time of data uploading and downloading. Focus is done on testing the performance of the deduplication procedure and algorithms designed in this scheme. The scheme provides a secure approach to protect and deduplicate the data stored in cloud by concealing plaintext from both CSP (Cloud Service Provider) and AP (Authorization Party). The security of the scheme is ensured by PRE theory, symmetric key encryption, AES and Elliptic curve Cryptography theory.

3.1 Advantages

- Storage-based data deduplication reduces the amount of storage needed for a given set of files. It is most effective in applications where many copies of very similar or even identical data are stored on a single disk a surprisingly common scenario. In the case of data backups, which routinely are performed to protect against data loss, most data in a given backup remain unchanged from the previous backup. Common backup systems try to exploit this by omitting (or hard linking) files that haven't changed or storing differences between files. Neither approach captures all redundancies, however. Hard-linking does not help with large files that have only changed in small ways, such as an email database; differences only find redundancies in adjacent versions of a single file (consider a section that was deleted and later added in again or a logo image included in many documents).
- Network data deduplication is used to reduce the number of bytes that must be transferred between endpoints, which can reduce the amount of bandwidth required.
- Virtual servers benefit from deduplication because it allows nominally separate system files for each virtual server to be coalesced into a single storage space. At the same time, if a given server customizes a file, deduplication will not change the files on the other servers—something that alternatives like hard links or shared disks do not offer. Backing up or making duplicate copies of virtual environments are similarly improved.

3.2 Module Description

User Module:

Registration:

In this module, each user registers his user details for using files. Only registered user can able to login in cloud server.

File Upload:

In this module, user upload a block of files in the cloud with encryption by using his secret key. This ensures the files to be protected from unauthorized user.

Download:

This module allows the user to download the file using his secret key to decrypt the downloaded data verify the data and re-upload the block of file into cloud server with encryption. This ensures the files to be protected from unauthorized user.

Re-upload:

This module allow the user to re-upload the downloaded files of blocked user into cloud server with resign the files(i.e.) the files is uploaded with new signature like new secret with encryption to protected the data from unauthorized user.

Admin Module:

View Files:

In this module, public auditor view the all details of upload, download, blocked user, reupload.

File Upload:

In this module, admin can also upload a block of files in the cloud with encryption by using his secret key. This ensures the files to be protected from unauthorized user.

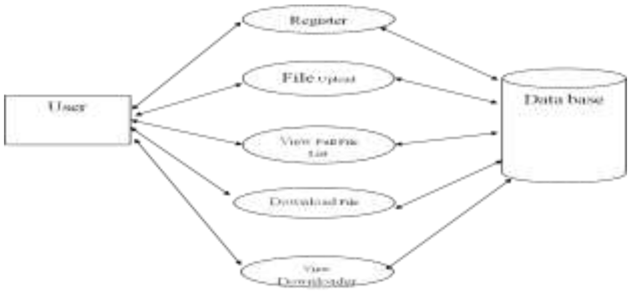


Figure. 3.2.1 Admin Module.

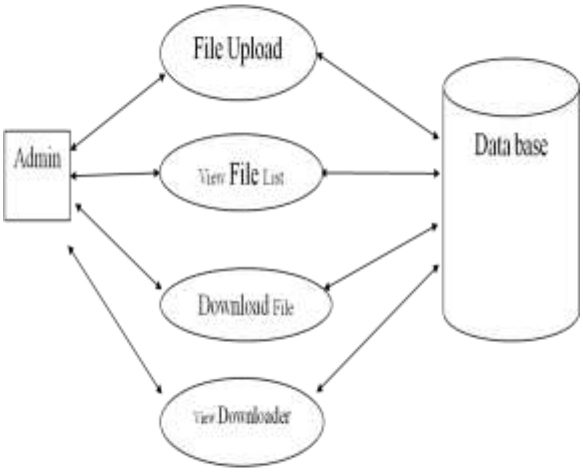


Figure. 3.2.2 User Module.

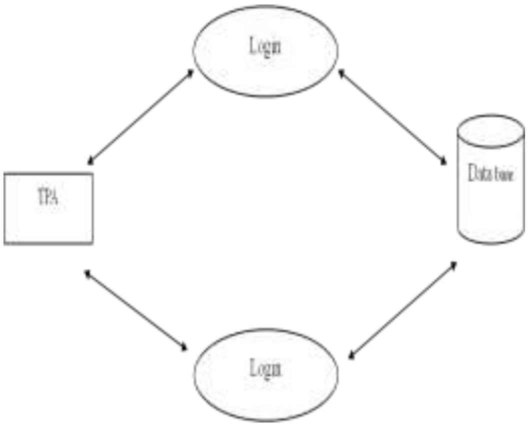


Figure. 3.2.3. TPA Module.

TPA (Third Party Authorization) Module:

In this module, third party user views the details of uploaded files in the cloud. But he could not able to read the content of the file and could not download the file.

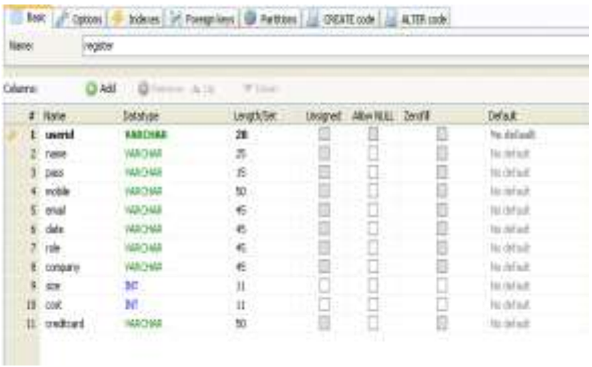
Download:

This module allows the admin to download the file using his secret key to decrypt the downloaded data and verify. This ensures the files to be protected from unauthorized user.

4. Results and Discussion

In this paper, a secure data sharing scheme, which can achieve secure key distribution and data sharing for dynamic group was provided. A secure way for key distribution without any secure communication channels was provided. The users can securely obtain their private keys from group manager without any Certificate Authorities due to the verification for the public key of the user. This scheme can achieve fine-grained access control, with the help of the group user list, any user in the group can use the source in the cloud and revoked users cannot access the cloud again after they are revoked. A secure data sharing scheme which can be protected from collusion attack was proposed. The revoked users can not be able to get the original data files once they are revoked even if they conspire with the untrusted cloud. This scheme can achieve secure user revocation with the help of polynomial function. This scheme is able to support dynamic groups efficiently, when a new user joins in the group or a user is revoked from the group, the private keys of the other users do not need to be recomputed and updated.

The computation cost is irrelevant to the number of revoked users in RBAC scheme. The reason is that no matter how many users are revoked, the operations for members to decrypt the data files almost remain the same. The cost is irrelevant to the number of the revoked users. The reason is that the computation cost of the cloud for file upload in this scheme consists of two verifications for signature, which is irrelevant to the number of the revoked users. The reason for the small computation cost of the cloud in the phase of file upload in RBAC scheme is that the verifications between communication entities are not concerned in this scheme. In this scheme, the users can securely obtain their private keys from group manager Certificate Authorities and secure communication channels. Also, this scheme is able to support dynamic groups efficiently, when a new user joins in the group or a user is revoked from the group, the private keys of the other users do not need to be recomputed and updated.

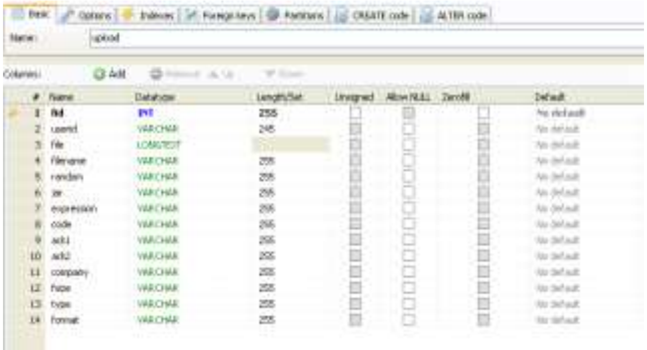


Name: register

Columns:

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	userid	VARCHAR	20	□	□	□	No default
2	name	VARCHAR	25	□	□	□	No default
3	pass	VARCHAR	15	□	□	□	No default
4	mobile	VARCHAR	50	□	□	□	No default
5	email	VARCHAR	45	□	□	□	No default
6	date	VARCHAR	45	□	□	□	No default
7	role	VARCHAR	45	□	□	□	No default
8	company	VARCHAR	45	□	□	□	No default
9	size	INT	11	□	□	□	No default
10	cost	INT	11	□	□	□	No default
11	creditcard	VARCHAR	50	□	□	□	No default

Figure 4.1 Registration details of user

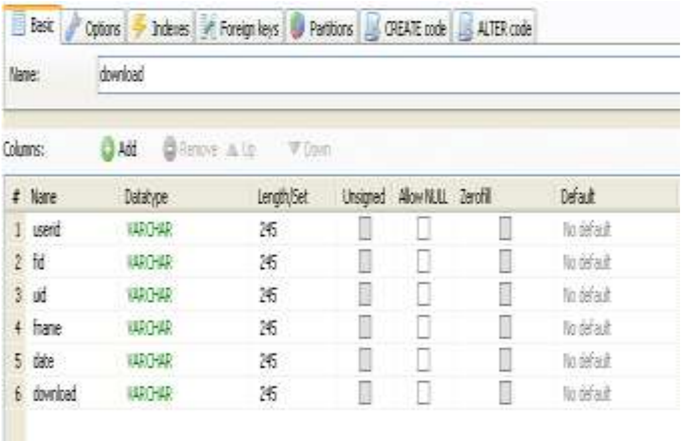


Name: upload

Columns:

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	fid	INT	255	□	□	□	No default
2	userid	VARCHAR	245	□	□	□	No default
3	file	LONGTEXT		□	□	□	No default
4	filename	VARCHAR	255	□	□	□	No default
5	random	VARCHAR	255	□	□	□	No default
6	sk	VARCHAR	255	□	□	□	No default
7	expression	VARCHAR	255	□	□	□	No default
8	code	VARCHAR	255	□	□	□	No default
9	ack1	VARCHAR	255	□	□	□	No default
10	ack2	VARCHAR	255	□	□	□	No default
11	company	VARCHAR	255	□	□	□	No default
12	fsize	VARCHAR	255	□	□	□	No default
13	type	VARCHAR	255	□	□	□	No default
14	format	VARCHAR	255	□	□	□	No default

Figure 4.2 Upload details of user



Name: download

Columns:

#	Name	Datatype	Length/Set	Unsigned	Allow NULL	Zerofill	Default
1	userid	VARCHAR	245	□	□	□	No default
2	fid	VARCHAR	245	□	□	□	No default
3	uid	VARCHAR	245	□	□	□	No default
4	fname	VARCHAR	245	□	□	□	No default
5	date	VARCHAR	245	□	□	□	No default
6	download	VARCHAR	245	□	□	□	No default

Figure 4.3 Download details of user

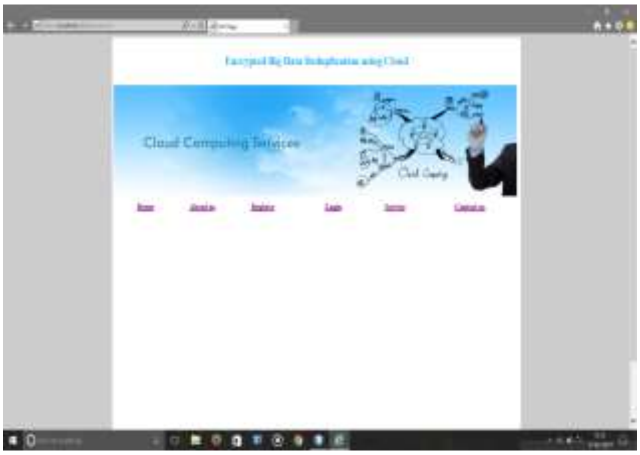


Figure 4.4 Home Page of Encrypted Big Data Deduplication using cloud

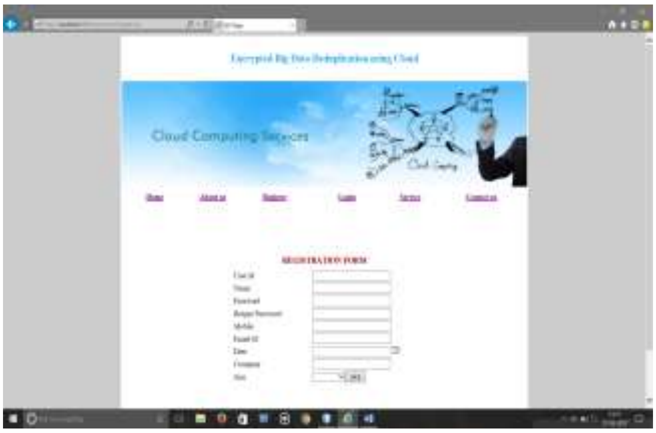


Figure 4.5 Registration Form

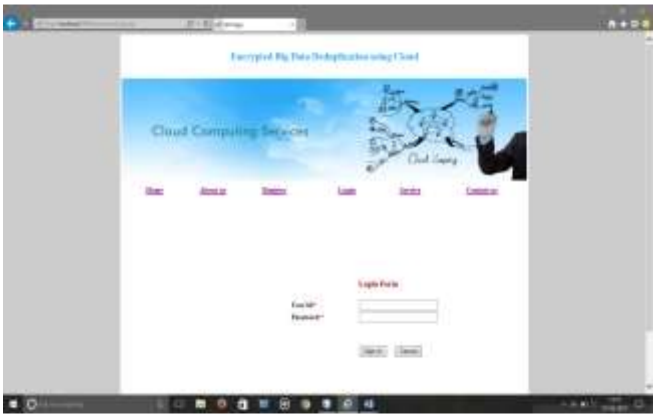


Figure 4.6 Login Form

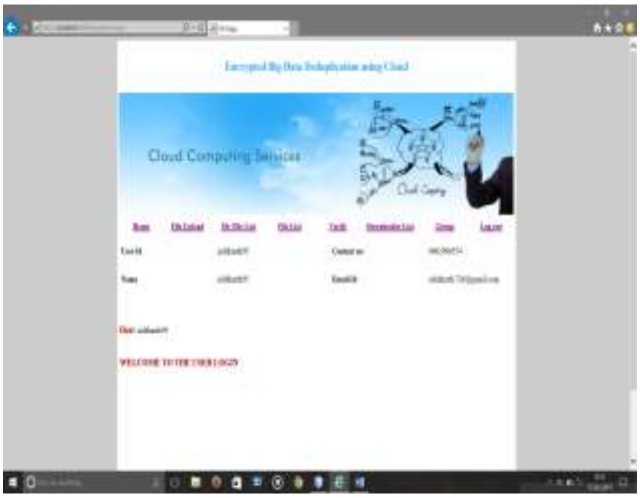


Figure 4.7 Upload and Download Details of a user

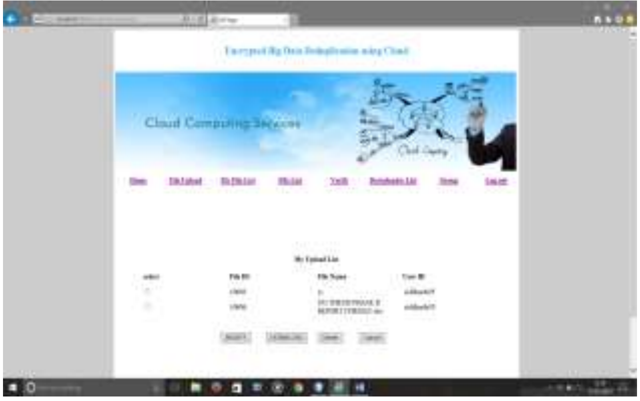


Figure 4.8 File Download

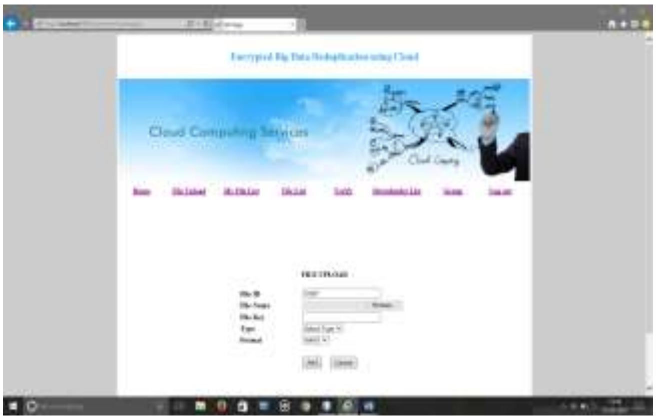


Figure 4.9 File Upload

5. Conclusion

Managing encrypted data with deduplication is important and significant in practice for achieving a successful cloud storage service, especially for big data storage. In this paper, a practical scheme to manage the encrypted big data in cloud with deduplication based on ownership challenge was proposed. This scheme can flexibly support data update and sharing with deduplication even when the data holders are offline. Encrypted data can be securely accessed because only authorized data holders can obtain the symmetric keys used for data decryption. Extensive performance analysis and test showed that this scheme is secure and efficient under the described security model and very suitable for big data deduplication. The advanced construction is motivated by the fact that customers always want to encrypt their data before updating and allows for integrity auditing and secure deduplication directly on encrypted data.

References

[1] "Improved proxy re-encryption schemes with applications to secure distributed storage," ACM Trans. Inform. Syst. Secur., vol. 9, no. 1, pp. 1–30, 2006.

[2] Bellare, et al,. "DupLESS: Server aided encryption for deduplicated storage,"in Proc. 22nd USENIX Conf.Secur. 2013, pp.179–194.

[3] Douceur, et al., "Reclaiming space from duplicate files in a server less distributed file system" in Proc. IEEE Int. Conf. Distrib. Comput. Syst., 2002, pp. 617–624.

[4] Fan, et al,. "Hybrid data deduplication in cloud environment," in Proc. Int. Conf. Inf. Secur. Intell. Control, 2012, pp. 174–177.

[5] Fuetal, "Accelerating restore and garbage collection in deduplication-based backup's systems via exploiting historical information," in Proc. USENIX Annu. Tech. Conf., 2014, pp. 181–192.

[6] Gao, "Game theoretic analysis on acceptance of a cloud data access control scheme based on reputation," M.S. thesis, Xidian University, State Key Lab of ISN, School of Telecommunications Engineering, Xi'an, China, 2015.

[7] Halevi, et al,."Proofs of ownership in remote storage systems," in Proc. 18th ACM Conf. Comput. Commun. Secur., 2011, pp. 491–500.

[8] Kaaniche et al.,"A secure client side deduplication scheme in cloud storage environments" in Proc. 6th Int. Conf. New Technol. Mobility Secure, 2014, pp. 1–7.

[9] Lillibridge, et al,. "Improving restore speed for backup systems that use inline chunk-based deduplication," in Proc. USENIX Conf. File Storage Technol., 2013, pp. 183–198.

[10] Li et al.,"A hybrid cloud approach for secure authorized deduplication," IEEE Trans. Parallel Distributed System vol.26, no.5, pp. 1206–1216, May 2015.

[11] Meye, et al,. "A secure two phase data deduplication scheme," in Proc. HPCC/CSS/ICESS, 2014, pp. 802–809.

[12] O'Neill et al., "Low-cost digital signature architecture suitable for radio frequency identification tags," IET Comput. Digital Techn., vol.4, no.1, pp.14–26, 2010.

[13] Paulo et al., "A survey and classification of storage deduplication systems, ACM Comput. Surveys, vol.47, no.1, pp.130,2014.

[14] Pietro et al,. "Boosting efficiency and security in proof of ownership for deduplication," in Proc. 7th ACM Symp. Inf. Comput. Commun. Secur., 2012, pp. 81–82.

[15] Sun, et al,. "DeDu: Building a deduplication storage system over cloud computing," in Proc. IEEE Int. Conf. Comput. Supported Cooperative Work Des., 2011, pp. 348–355.

[16] Wallace, et al., "Characteristics of backup workloads in production systems,"inProc.USENIX Conf. FileStorage Technol., 012, pp.1–16.

[17] Wei, et al., "Security and privacy for storage and computation in cloud computing," Inf. Sci., vol. 258, pp. 371–386, 2014.

[18] Yan, et al., "Controlling cloud data access based on reputation," Mobile Netw. Appl., vol. 20, no. 6, 2015, pp. 828–839.

Author Profile

N..B. Mahesh Kumar received his Bachelor of Technology degree in Information Technology from Anna University, Chennai, India in 2005. He received the Master of Engineering degree in Computer Science and Engineering from Anna University of Technology, Coimbatore, India in 2009. And Doctor of Philosophy in Computer Science and Engineering from Anna University, Chennai, India in 2016. She has 9 years teaching experience in academic field. A life member of Indian Scociety of Technical Education.