European Journal of Applied Sciences 8 (4): 225-229, 2016
ISSN 2079-2077

DOI: 10.5829/idosi.ejas.2016.8.4.22969

Data Integrity Checking in Multi Cloud Storage

G. Arthi and N. Srinivasan Narayanasamy

Department-MCA, Faculty of Computing,
Sathyabama University, Chennai-600119, Tamilnadu, India

Abstract: In the recent times, Cloud Storage has become a vital growth tendency with regard to Information Technology. Nevertheless, data security has turned a crucial issue that impedes it in connection with commercial applications like data integrity, availability and confidentiality. Outsourcing of storage is one emerging trend that happens to prompt several intriguing security problems, a number of which that have already been investigated extensively during the past. Nonetheless, Provable Data Possession (PDP) is a particular subject that finds entry only in research work. In this study, we have suggested about protecting database on the cloud toward accessing information securely. We have put forward String matching algorithm and Ring Signature along with base64 algorithm. Within this analysis, a File Allocation table (FAT), that is something that can be understood easily, has been introduced. File description mechanism contains all properties pertaining to a file that one requires to know such as dates, permission and name. Cloud computing is based on outsourcing the computing exercises to some third party. This involves risk threats with regard to integrity, availability and confidentiality of information and service. It is a vital issue convincing cloud customers about their data being maintained intact especially as they normally do not use to store that information locally. For solving the said issue, we suggest ring signature algorithm toward storing information securely on cloud server while data can be accesses through verification of authentication details only. Remote information integrity testing happens to be one primitive approach for addressing the said problem. For normal cases, when client data is stored in several cloud servers by them, integrity testing and distributed storage become essential. Hence, depending on distributed computing, it is possible for us to analyze distributed remote information integrity testing model and furnish the related concrete protocol pertaining to multiple cloud storage. Our study suggests string matching algorithm with respect to information owner's access only to the data through a given particular key value for choosing the right file inside the database of cloud server. This way, attackers will not be able to attack data present in multiple cloud storage. One model is proposed in this analysis for providing a powerful security mechanism toward accountability, authorization and authentication of information sharing within the environment of cloud. Results from experiments also have validated our technique and they are presented as well.

Key words: Provable Data Possession · Multiple Cloud Storage · File Allocation Table · String Matching Algorithm · Ring Signature · Base64 Algorithm

INTRODUCTION

In the recent past, the idea of data warehousing using third party and, particularly, information outsourcing has really become very popular. Data outsourcing primarily suggests that owner of data (client) transfers his/her information to some third party service provider (server) that is presumed to supposedly for some charge reliably store the information and also make that available on demand to the data's proprietor (and sometimes, other). Attractive features of the outsourcing consists of lowers costs through savings in storing, staff and maintenance apart from transparent data upkeep and improved availability of the data. Many issues related to security in outsourcing of data have already been analyzed during past decade. Initially, work was focused about data integrity and authentication, correction of server results and completion of outputs with respect to customers' [1, 2] queries.

Corresponding Author: G. Arthi, Department-MCA, Faculty of Computing, Sathyabama University, Chennai-600119, Tamilnadu, India.

After that, studies were centered about outsourcing of encoded information (laying still low trust on server) and related difficult issues that were primarily associated with effective querying across an encoded [3, 4] domain. During recent times, anyhow, the issue of Provable Data Possession (PDP) which is sometimes also known as Proof to Information Recoverability (POR) has been identified in research processes. In a public cloud scenario, data protection in cloud happens to be the crucial challenge. Users tend to adopt outsourcing their information to the cloud for reliable, flexible and efficient storage facility. Due to issues related to security, data need to be revealed via Cloud Service Suppliers (CSP) along with other cloud users. For protecting the information from unapproved revelation, in this study, we have proposed ring signature, string matching algorithm and base64 algorithm [5]. The most customary implementations are having a grave drawback, namely, at the time of file deletion and new or fresh files being written on the media, the directory fragments incline to get scattered across the whole of the media, thus slowing down the process of writing and reading. Defragmentation may be considered a solution for this, but it is often found to be one lengthy procedure by itself and needs to be regularly performed for keeping the FAT system of file clean. Verification of authenticity of the information has come up as one vital problem in the storage of information in servers that are not trust worthy. This issue arises in network file schemes, peer-to-peer system of storage [6], database systems and web-facility object depositories [7]. Such schemes retrain repository servers from misleading or altering information by providing authentication tests while accessing information [8].

Related Work: During 2007, R.Burns, G.Ateniese, J.Herring, R.Urtmola, D.Song, Z.Peterson and L.Kissner put forward a suggestion. This study describes of the Provable Data Possession (PDP) system that permits any verifier to examine the accuracy of a given clients information that has been stored on any un trusted server by making use of sampling techniques and RSA-oriented homomorphic authenticators. The benefit of this strategy is that it is possible by the verifier to audit publicly the data integrity without recalling the whole information, which is termed public auditing. Problem with this study is that this mechanism may be suitable only on auditing integrity of any personal information [10]. In 2009, C.Wang, K.Ren, W.Lou and Q.Wang proposed a method. This technique leverages homomorphic tokens for ensuring the accuracy of the eraser codes-oriented information that is distributed on several servers. The primary contribution of the said mechanism is its ability to support dynamic information and recognition of misbehaved servers. The issue dealt with in this analysis is leakage pertaining to identity concealment to public verifiers [11, 12]. In 2010, R.Curtmola, B.Cheng, R.BURNS and G.Ateniese introduced one mechanism to audit the accuracy of information in the context of multi-server environment, wherein this information are encrypted through network coding in place of utilizing eraser codes. This system reduces communication expenses considerably during data repair phase. The issue with this particular paper is that it needs two enhanced systems. The first one happens to be BLS signatures and second is pseudo-random operation [13]. In 2008, R.D.Pietro, G.Ateniese, G.Tsudik and L.V.Mancini presented one powerful PDP mechanism on the basis of symmetric keys. The said mechanism is capable of supporting updating and deletion functions on the data, but still, inserting function is not possible in this scheme. It manipulates symmetric keys for verifying data integrity and it does not support public verification. The problem with this paper is that the scheme can only provide limited verification request to the user [14, 15]. In 2007, B.S.Kaliski and A.Juels, in their analysis, provided PORs system that can examine the accuracy of information on any server that is not trust-worthy. Original file gets added to a group of arbitrarily assessed test blocks known as sentinels. Verifier can challenge the server that is not trust-worthy by stating the locations of a group of sentinels along with requesting the unreliable server to revert the related sentinel values. POR Protocol that is based on sentinel is compliant to real-life application. The said issue focuses only on personal information in cloud. Initially, it is possible for an adversary to attempt corrupting the shared datas integrity. Secondly, cloud service supplier corrupt inadvertently (or sometimes detach) information in the storage because of human errors and hardware failures. To worsen the matters, cloud service supplier is normally financially driven, meaning that he or she may not be willing to intimate users of such data corruption. Identity of signer on every block in the data that is shared is confidential and private to the particular group. During audit process, any public checker who has been granted permission only for verifying accuracy related to integrity of the shared data may even attempt to unveil the signer's identity on every block inside the distributed data on the basis of authentication metadata [16].

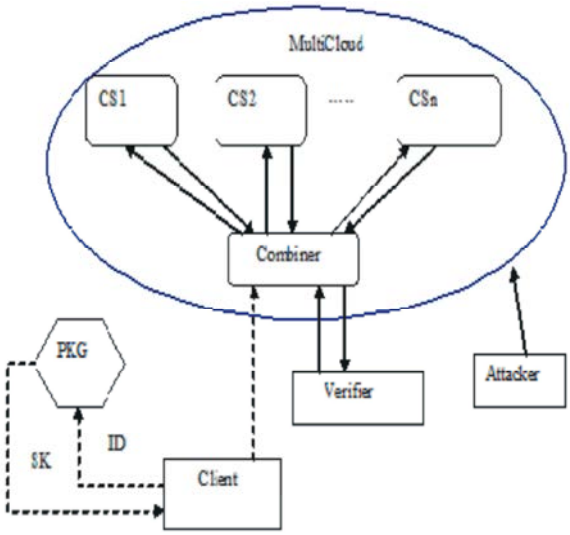

Fig. 1: Overall Architecture

Proposed System

Overview: In several-layered cloud scenario, distributed provable data possession happens to be a crucial factor for securing the remote information. The authenticator maintains information for performing integrity examining. Client who is having huge data that need to get stored with multi cloud toward computation and maintenance may either be a corporation or an individual customer. (Cloud Server) CS that is managed via cloud service supplier is having significant computation resource and storage space for maintaining client's information. Combiner that encounters request for storage distributes block-tag groups on to matching cloud servers. At the time of challenge receipt, it splits challenge and shares them with various cloud servers. At the time of getting responses via cloud servers, combiner combines them and then sends the accumulated responses to the authenticator. Harmful C or CS will not be able to create any proof for data possession In case the blocks are already altered or deleted. During each access of file by user, data blocks will dynamically be re-allotted to cloud servers. This enables achieving confidentiality within the cloud computing process.

Proposed Method and Algorithm: Any User has to undergo one initial stage process of registration at the web side. Users supply their own private data toward this procedure. The information is stored in turn, by the server, in the database. Next to the registration process, it is possible for user to upload files on to server. The files that have been uploaded are being stored on server.

Overall Architecture

Fat and Ring Signature Method: This splits the entire file of F as n blocks. Customer gets ready for storing blocks in several cloud servers. The cloud server that is employed toward storage of data will have a distinct area for storage which gets formatted with file allocation table (FAT) File System (FS). In case a user uploads information to some other cloud, during the time in which it gets split as different bits and each piece gets appended using Ring signatures prior to depositing the information within FAT FS. Information gets divided into several tiny blocks, wherein every block can be signed independently by user. The information gets encrypted as well, by using Base64 algorithm.

String Matching with Verifier: FAT FS is having Meta data and proper indexing toward various pieces of the information that gets uploaded by the given user. An attacker may be able to change information in any selected cloud server. During the time of user accessing the data, the authenticator will perform integrity examination. The authenticator can access the FS via cloud that will reciprocate with some arbitrary blend of all blocks in place of entire data being recalled during the practice of integrity examination. Integrity examination may be performed more efficiently. The process of recovery may be performed using string matching algorithm when signature authentication fails.

RESULT AND DISCUSSION

Server Architecture: Figure 2 shows server architecture. In this server, the user can select the number of server and selected server IP address need to give in server architecture.

User Registrations: Figure 3 shows user registrations. In user registrations the user need to fill user id, user name, password, contact and email etc to register in cloud.

File Upload: Figure 4 shows file upload. The user has to choose file to upload after selecting file upload into cloud.

Data Report: Figure 5 shows data report. After uploaded data in Multi-cloud the file will store in different cloud and auditor validate the file.

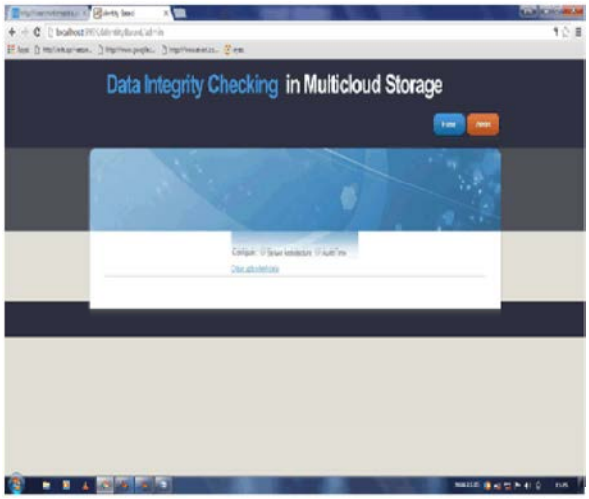

Fig. 2: Server Architecture

Fig. 3: User Registration

Fig. 4: File Upload

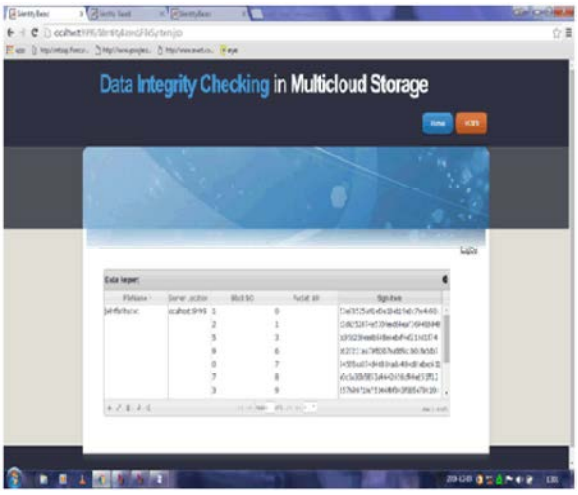

Fig. 5: Data Report

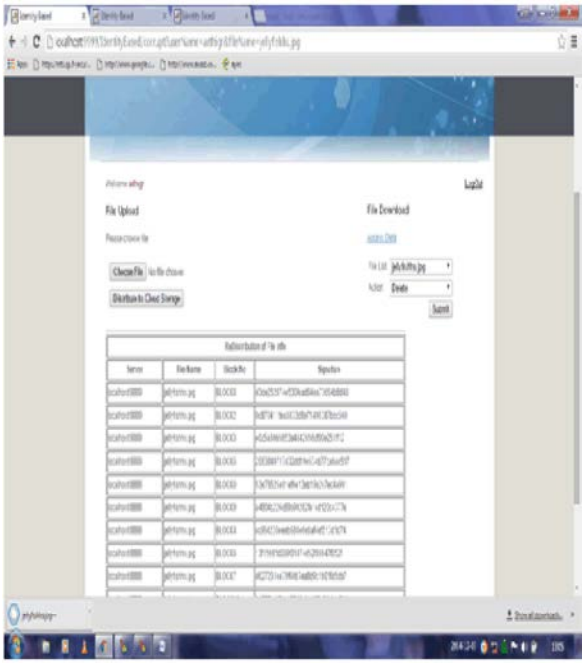

Fig. 6: File Download

File Download: Figure 6 shows file download. By giving file name the file will download from the cloud. When downloading a file from cloud different cloud data will combine and retrieve to user.

CONCLUSION

The aim of this paper is protecting information in multiple cloud storage by making use of base64 algorithm, string matching algorithm and ring signature. The problem with earlier systems is data access within cloud and possibility of alteration of information by attackers or

malevolent data. For overcoming this issue, we have suggested ring signature toward allotting data security by making use of FAT method. The techniques used in this study are ring signature along with the use of data allocation in multi cloud with some information key value that gets assumed arbitrarily. The uploaded information will be encoded by making use of base64 algorithm and uploading is done arbitrarily on multi cloud. At the end, we suggest employing the string matching algorithm for protecting security of information in cloud as it is possible to access information with matching key value only. Otherwise, data will not be available to access, view, or for altering purpose. Hence it will not be possible for any attacker to access the data. It will be protected securely. We suggest future research toward further improvement in speed and for better and dependable result.

REFERENCES

1. Devanbu, P., M. Gertz, C. Martel and S. Stubblebine, 2003. Authentic third-party data publication, in IFIP DBSec' 03, also in Journal of Computer Security, 11(3): 291-314.
2. Mykletun, E., M. Narasimha and G. Tsudik, 2004. Authentication and integrity in outsourced databases, in ISOC NDSS'04.
3. Hacigumus, H., B. Iyer, C. Li and S. Mehrotra, 2002. Executing sql over encrypted data in the database-serviceprovider model, in ACM SIGMOD'02.
4. Boneh, D., G. di Crescenzo, R. Ostrovsky and G. Persiano, 2004. Public key encryption with key-word search, in EUROCRYPT'04.
5. The FAT File System A, Walk Through, Experiment Jerry Breecher November, 2008.
6. Kallahalla, M., E. Riedel, R. Swaminathan, Q. Wang and K. Fu. Plutus: Scalable secure file sharing on untrusted storage. In Proc. of FAST, 2003.
7. Yumerefendi, A.Y. and J. Chase, 2007. Strong accountability for network storage. In Proc. of FAST.
8. Maheshwari, U., R. Vingralek and W. Shapiro, 2000. How to build a trusted database system on untrusted storage. In Proc. Of OSDI,
9. Ren, J. and L. Harn, 2008. Generalized ring signatures, IEEE Transactions on Dependable and Secure Computing, 5(3): 155-163.
10. Watson Bruce william, 1999. The performance of single-keyword and multiple-keyword pattern matching algorithms.
11. Ateniese, G., R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson and D. Song, 2007. Provable Data Possession at Untrusted Stores, Proc. 14th ACM Conf. Computer and Comm. Security, (CCS '07), 598-610.
12. Wang, C., Q. Wang, K. Ren and W. Lou, 2009. Ensuring Data Storage Security in Cloud Computing, Proc. 17th Int'l Workshop Quality of Service (IWQoS'09), 1-9.
13. Chen, B., R. Curtmola, G. Ateniese and R. Burns, 2010. Remote Data Checking for Network Coding-Based Distributed Storage Systems,Proc. ACM Workshop Cloud Computing Security Workshop (CCSW'10), 31-42.
14. Wang, B., B. Li and H. Li, 2013. Certificateless Public Auditing for Data Integrity in the Cloud, Proc. IEEE Conf. Comm. and Network Security (CNS'13), 276-284.
15. Ateniese, G., R.D. Pietro, L.V. Mancini and G. Tsudik, 2008. Scalable and Efficient Provable Data Possession, Proc. Fourth Int'l Conf. Security and Privacy in Comm, Networks (SecureComm'08).
16. Juels, A. and B.S. Kaliski, 2007. PORs: Proofs of Retrievability for Large Files, Proc. 14th ACM Conf. Computer and Comm. Security (CCS'07), 584-597.