INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

CONFIRMABLE MAP-BASED PROVABLE MULTICITY DYNAMIC DATA POSSESSION IN CLOUD COMPUTING SYSTEMS

Dr.V.Goutham[1], A. Roja Ramani[2], B. Priyanka[3]

[1]Head, Department of Computer Science and Engineering, Teegala Krishna Reddy Engineering College
Hyderabad, T.S-500 097, India. E-Mail: v.goutham@gmail.com
[2]Asst. Professor, Department of Computer Science and Engineering, Teegala Krishna Reddy Engineering College Hyderabad, T.S-500 097, India. E-Mail: rojadabbulu@gmail.com
[3]Department of Computer Science and Engineering, Teegala Krishna Reddy Engineering College
Hyderabad, T.S-500 097, India. E-Mail: Priyankareddy333@gmail.com

Abstract: - A service provider that offers customers storage or software services available via a private (private cloud) or public network (cloud).Usually, it means the storage and software is available for access via the Internet. A third-party entity that manages and distributes remote, cloud-based data backup services and solutions to customers from a central data centre. Cloud backup providers enable customers to remotely access services using a secure client login application to back up files from the customer's computers or data centre to the online storage server using an encrypted connection. A cloud backup solution enables enterprises or individuals to store their data and computer files on the Internet using a storage service provider, rather than storing the data locally on a physical disk, such as a hard drive or tape backup. In storage terminology, a remote backup refers to an online managed backup service for backing up data to a remote, cloud-based server ("cloud backup").To update or restore a cloud backup, customers need to use the service provider's specific client application or Web browser interface. Files and data can be automatically saved to the cloud backup service on a regular, scheduled basis, or the information can be automatically backed up anytime changes are made.

KEYWORDS: Cloud service providers, Outsourcing data storage, MB-PMDDP, Map version

Cloud Service Providers

Storage-as-a-Service offered by cloud service providers (CSPs) emerged as a solution to mitigate the burden of large local data storage and reduce the maintenance cost by means of outsourcing data storage. CSP often provides better disaster recovery by replicating the data on multiple servers across multiple data centres achieving a higher level of availability [9]. Thus, many authorized users are allowed to access the remotely stored data from different geographic locations making it more convenient for them. Data owner physically releases sensitive data to a remote CSP, there are some concerns regarding confidentiality, integrity, and access control of the data[8]. The confidentiality feature can be guaranteed by the owner via encrypting the data before outsourcing to remote servers. The proposed model provides trusted computing environment by addressing important issues related to outsourcing the storage of data, namely confidentiality, integrity, access control and mutual trust between the data owner and the CSP.

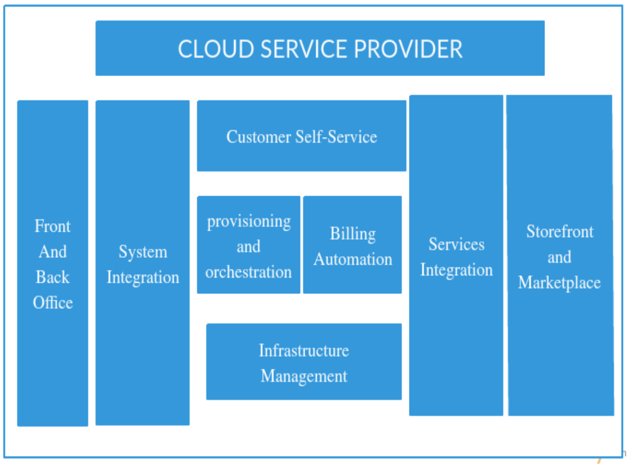

Fig: 1 Cloud Service Provider

MB-PMDDP

MB-PMDDP scheme produce unique differentiable copies of the data file is the core to design a provable multi-copy data possession scheme. Identical copies enable the CSP to simply deceive the owner by storing only one copy and pretending that it stores multiple copies [1]. Using a simple yet efficient way, the proposed scheme generates distinct copies utilizing the diffusion property of any secure encryption scheme. The diffusion property ensures that the output bits of the cipher text depend on the input bits of the plaintext in a very complex way, *i.e.*, there will be an unpredictable complete change in the cipher text, if there is a single bit change in the plaintext [11]. The interaction between the authorized users and the CSP is considered through this methodology of generating distinct copies, where the former can decrypt/access a file copy received from the CSP. In the proposed scheme, the authorized users need only to keep a single secret key to decrypt the file copy. In this work, we propose a MB-PMDDP scheme allowing the data owner to update and scale the blocks of file copies outsourced to cloud servers which may be un trusted [10]. Validating such copies of dynamic data requires the knowledge of the block versions to ensure that the data blocks in all copies are consistent with the most recent modifications issued by the owner. Furthermore, the verifier should be aware of the block indices to guarantee that the CSP has inserted or added the new blocks at the requested positions in all copies. To this end, the proposed scheme is based on using a small data structure (metadata), which we call a map-version table.

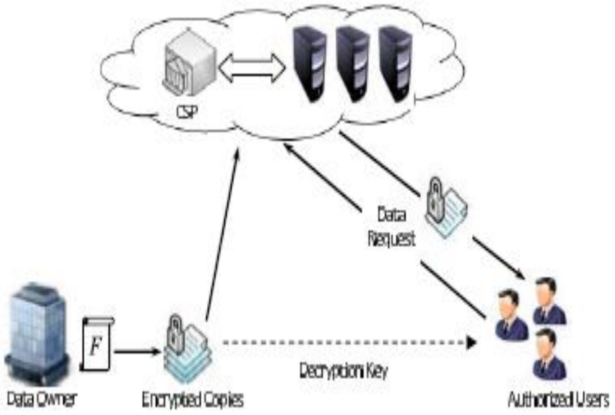

Fig: 2 Cloud Computing Data Storage System Model

Outsourcing Data Storage

The owner encrypts the data before sending to cloud servers. To access the data, the authorized user sends a data-access request to the CSP, and receives the data file in an encrypted form that can be decrypted using a secret key generated by the authorized user [2]. It is assumed that the interaction between the owner and the authorized users to authenticate their identities has already been completed, and it is not considered in this work. The TTP is an independent entity, and thus has no incentive to collude with any party. However, any possible leakage of data towards the TTP must be prevented to keep the outsourced data private [4]. The TTP and the CSP are always online, while the owner is intermittently online. The authorized users are able to access the data file from the CSP even when the owner is offline.

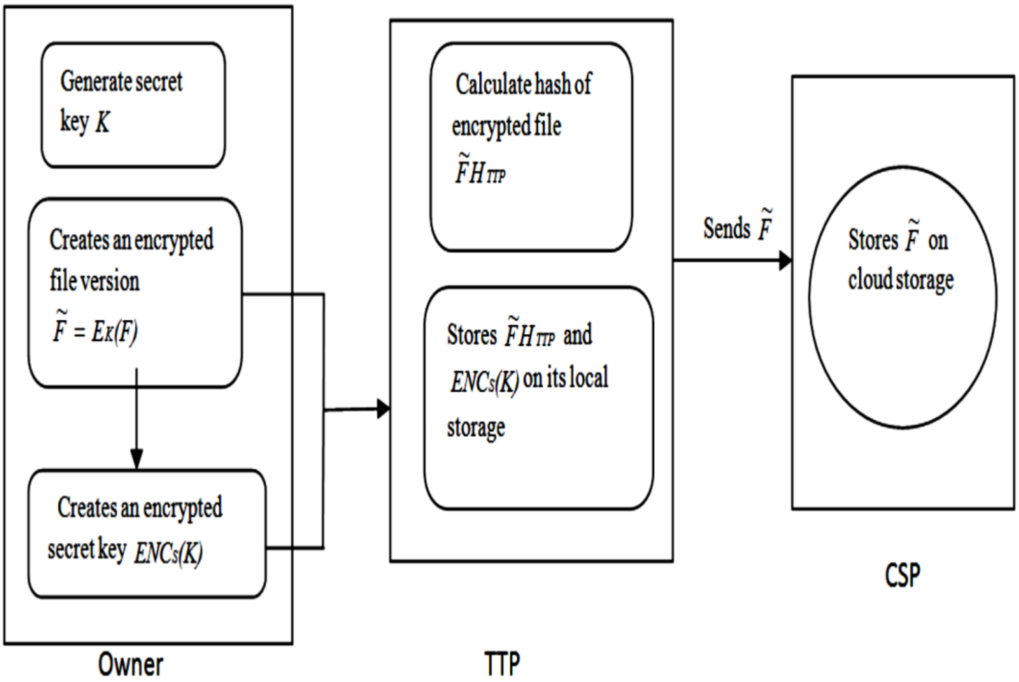
Generate secret key K
Creates an encrypted file version $\tilde{F} = E_K(F)$
Creates an encrypted secret key $ENC_S(K)$
Owner
Calculate hash of encrypted file $\tilde{F}H_{TTP}$
Stores $\tilde{F}H_{TTP}$ and $ENC_S(K)$ on its local storage
TTP
Sends $\tilde{F}$
Stores $\tilde{F}$ on cloud storage
CSP

Fig: 3 Setup and file preparation for data outsourcing

Map-Version

The map-version table (MVT) is a small dynamic data structure accumulates on the verifier side to authenticate the reliability and uniformity of all file copies outsourced to the CSP. The MVT consists of three columns: serial number (SN), block number (BN), and block version (BV). The *SN* is an indexing to the file blocks. It point out the physical position of a block in a data file [6]. The BN is a counter used to make a logical numbering/indexing to the file blocks. Therefore, the relation between BN and SN can be observed as a mapping between the logical number BN and the physical position SN. The BV specifies the current version of file blocks. When a data file is originally created the BV of each block is 1[3]. If a specific block is being updated, its BV is incremented by 1. comment *2:* It is significant to note that the verifier remain only one table for infinite number of file copies, *i.e.*, the storage condition on the verifier side does not depend on the number of file copies on cloud servers. For *n* copies of a data file of size | *G*|, the storage condition on the CSP side is O(n |G|), while the verifier's overhead is O(m) for all file copies (m is the number of file blocks) [7]. Comment 3: The MVT is applied as a linked list to make simpler the insertion deletion of table entries. For actual achievement, the SN is not needed to be stored in the table; SN is considered to be the entry/table index, i.e., each table entry contains just two integers BN and BV (8 bytes) [5]. As a result, the total table size is 8*m* bytes for all file copies. We additionally note that even if the table size is linear to the file size, in practice the previous would be smaller by several orders of magnitude.

SN	BN	BV
1	1	1
2	2	1
3	3	1
4	4	1
5	5	1
6	6	1
7	7	1
8	8	1

(a) Initially

SN	BN	BV
1	1	1
2	2	1
3	3	1
4	4	2
5	5	1
6	6	1
7	7	1
8	8	1

(b)Modifying Block at position 4

Fig: 4 Changes in MVT due to different dynamic operations on copies of a file

REFERENCES

[1] G. Ateniese et al., “Provable data possession at untrusted stores,” in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), New York, NY, USA, 2007, pp. 598–609.

[2] K. Zeng, “Publicly verifiable remote data integrity,” in Proc. 10th Int. Conf. Inf. Commun. Secur. (ICICS), 2008, pp. 419–434.

[3] Y. Deswarte, J.-J. Quisquater, and A. Saïdane, “Remote integrity checking,” in Proc. 6th Working Conf. Integr. Internal Control Inf. Syst. (IICIS), 2003, pp. 1–11.

[4] D. L. G. Filho and P. S. L. M. Barreto, “Demonstrating data possession and uncheatable data transfer,” IACR (International Association for Cryptologic Research) ePrint Archive, Tech. Rep. 2006/150, 2006.

[5] F. Sebé, J. Domingo-Ferrer, A. Martinez-Balleste, Y. Deswarte, and J.-J. Quisquater,“Efficient remote data possession checking in critical information infrastructures,” IEEE Trans. Knowl. Data Eng., vol. 20, no. 8, pp. 1034–1038, Aug. 2008.

[6] P. Golle, S. Jarecki, and I. Mironov, “Cryptographic primitives enforcing communication and storage complexity,” in Proc. 6th Int. Conf. Financial Cryptograph. (FC), Berlin, Germany, 2003, pp. 120–135.

[7] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan, “Auditing to keep online storage services honest,” in Proc. 11th USENIX Workshop Hot Topics Oper. Syst. (HOTOS), Berkeley, CA, USA, 2007, pp. 1–6.

[8] M. A. Shah, R. Swaminathan, and M. Baker, “Privacy-preserving audit and extraction of digital contents,” IACR Cryptology ePrint Archive, Tech. Rep. 2008/186, 2008.

[9] E. Mykletun, M. Narasimha, and G. Tsudik, “Authentication and integrity in outsourced databases,” ACM Trans. Storage, vol. 2, no. 2, pp. 107–138, 2006.

[10] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, “Scalable and efficient provable data possession,” in Proc. 4th Int. Conf. Secur. Privacy Commun. Netw. (SecureComm), New York, NY, USA, 2008, Art. ID 9.

[11] C. Wang, Q. Wang, K. Ren, and W. Lou. (2009). “Ensuring data storage security in cloud computing,” IACR Cryptology ePrint Archive, Tech. Rep. 2009/081. [Online]. Available: http://eprint.iacr.org/

AUTHOR DETAILS

1. **Dr. V. GOUTHAM** is a Professor and Head of the Department of Computer Science and Engineering at Teegala Krishna Reddy Engineering College affiliated to J.N.T.U Hyderabad.

2. **Mrs. A.ROJA RAMANI** is working as a Assistant Professor in the Department of Computer Science and Engineering at Teegala Krishna Reddy Engineering College affiliated to J.N.T.U Hyderabad.

3. **Ms. B. PRIYANKA** Department of Computer Science and Engineering at Teegala Krishna Reddy Engineering College affiliated to J.N.T.U Hyderabad.