International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682
Vol.6, No.10, October 2017 DOI:10.15693/ijaist/2017.v6i11.414-416

Cloud Data duplication using HMAC-SHA-1 for Secure Data Storage

M. Meenakshi

HOD, Department of CSE
Geethanjali College of Engineering & Technology, Kurnool,AP

Abstract:

Data duplication 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 DE duplication, 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 duplication. Different from traditional duplication systems, the differential privileges of users are further considered in duplicate check besides the data itself. We also present several new DE duplication constructions supporting authorized duplicate check in hybrid cloud Architecture. Security analysis demonstrates that our scheme is secure in terms of the definitions specified in the proposed security model. As a proof of concept, we implement a prototype of our proposed authorized duplicate check scheme and we show that our proposed authorized duplicate check scheme incurs minimal overhead compared to normal operations.

Keywords: - Data duplication, SHA-1, HMAC-SHA-1, Proof of Ownership Protocol, AES algorithm.

1. INTRODUCTION

Cloud computing enables new business models and cost effective resource usage. Instead of maintaining their own data centre, companies can concentrate on their core business and purchase resources when it will needed. Especially when combining publicly accessible clouds with a privately maintained virtual infrastructure in a hybrid cloud, the hybrid cloud technology can open up new opportunities for Businesses. Today's cloud service Providers offer both highly available storage and massively parallel computing resources at relatively low costs. As cloud computing becomes prevalent, an increasing amount Of data is being stored in the cloud and shared by users with specified privileges, which define the access rights of the stored data. One critical challenge of cloud storage services is the Management of the ever-increasing volume of data. Data DE duplication is a specialized data compression technique for Eliminating duplicate copies of repeating data in storage. DE duplication can take place at either the file level or the block level for file level DE duplication, it eliminates duplicate copies of the same file. Traditional encryption, while providing data confidentiality is incompatible with data DE duplication. Specifically, traditional encryption requires different users to encrypt their data with their own keys. Thus, identical data copies of different users will lead to different cipher texts making DE duplication impossible. Convergent encryption has been proposed to enforce data confidentiality while making DE duplication feasible. It encrypts/decrypts a data copy with a Convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users retain the keys and send the cipher text to the cloud. Since the encryption operation is deterministic and is derived from the data content, identical data copies will generate the same convergent key and hence the same cipher text. A Hybrid Cloud is a combined form of private clouds and public clouds in which some critical data resides in the enterprise's private cloud while other data is stored in and accessible from a public cloud. Hybrid clouds seek to deliver the advantages of scalability, reliability, rapid deployment and potential cost savings of public clouds with the security and increased control and management of private clouds.

2. RELATED

WORK Existing System

To make data management scalable in cloud computing, DE duplication has been well-known technique and has attracted more and more attention recently. But Traditional encryption, while providing data confidentiality, is incompatible with data DE duplication. Specifically, traditional encryption requires different users to encrypt their data with their own keys. Thus, identical data copies of different users will lead to different cipher texts, making DE

Duplication impossible.

Proposed System

Convergent encryption has been proposed to enforce data confidentiality while making DE duplication feasible. It encrypts/decrypts a data copy with a convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users retain the keys and send the Cipertext to the cloud. Since the encryption operation is deterministic and is derived from the data content, identical data copies will generate the same convergent key and hence the same Cipertext.

Algorithms:

- SHA-1
- HMAC-SHA-1
- Proof of Ownership Protocol
- AES algorithm

3. IMPLEMENTATION

USER Module:

In this module, Users are having authentication and security to access the detail which is presented in the ontology system. Before accessing or searching the details user should have the account in that otherwise they should register first.

• File Tag (File) - It computes SHA-1 hash of the File as File Tag.

• Token Req(Tag, UserID) - It requests the Private Server for File Token generation with the File Tag and User ID.

• DupCheck Req(Token) - It requests the Storage Server for Duplicate Check of the File by sending the file token received from private server.

• Share TokenReq(Tag, {Priv.}) - It requests the Private Server to generate the Share File Token with the File Tag and Target Sharing Privilege Set.

• File Encrypt(File) - It encrypts the File with Convergent Encryption using 256-bit AES algorithm in cipher block chaining (CBC) mode, where the convergent key is from SHA-256 Hashing of the file.

• File Upload Req(FileID, File, Token) – It uploads the File Data to the Storage Server if the file is Unique and updates the File Token stored.

Private Server:

Our implementation of the Private Server includes corresponding request handlers for the token generation and maintains a key storage with Hash Map.

• TokenGen(Tag, UserID) - It loads the associated privilege keys of the user and generate the token with HMAC-SHA-1 algorithm.

• ShareTokenGen(Tag, {Priv.}) - It generatesthe share token with the corresponding privilegekeys of the sharingprivilege set with HMAC-SHA-1algorithm.

Public Server (Storage Server):

Our implementation of the Storage Server provides deduplication and data storage with following handlers and maintains a map between existing files and associated token with Hash Map.

• DupCheck(Token) - It searches the File to Token Map for Duplicate and

• FileStore(FileID, File, Token) - It stores the File on Disk and updates the Mapping.

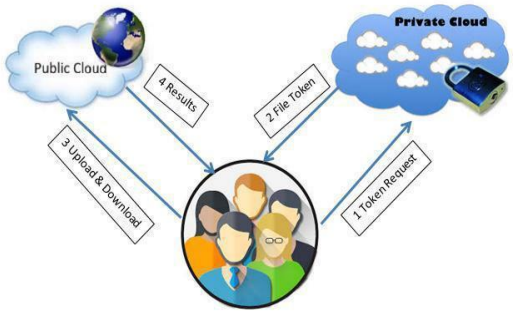

Fig:-1 Architecture for authorized deduplication.

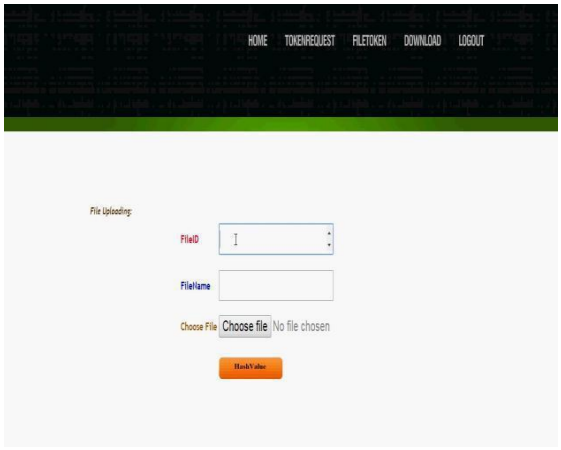

Fig:-2 Authentication and Authorization

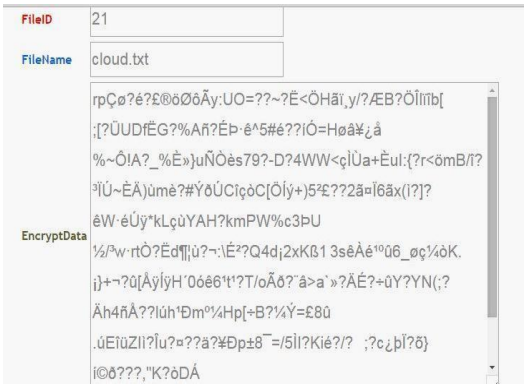

Fig: 3 Data Upload

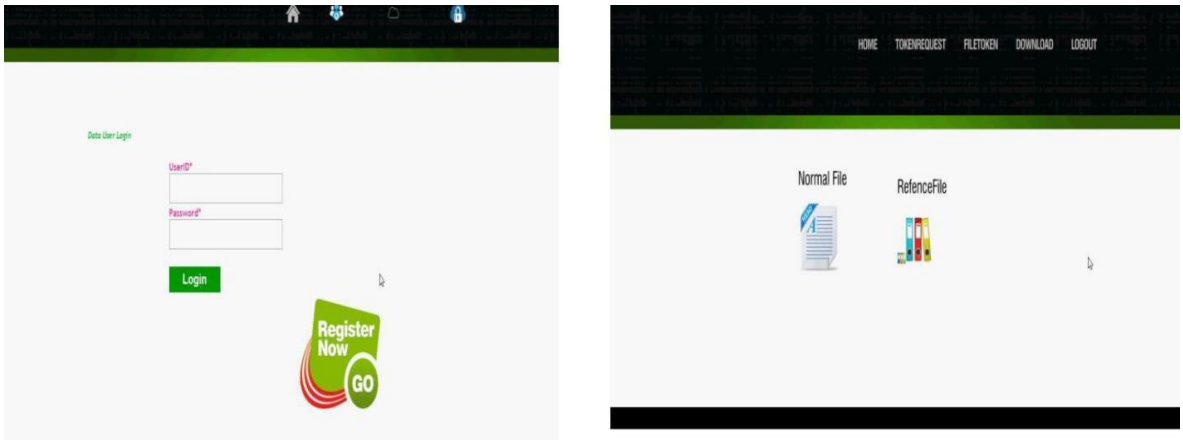

Fig: 4 Normal & Reference File

5. CONCLUSION

In this Project, the notion of authorized data deduplication was proposed to protect the data security by including differential privileges of users in the duplicate check. In thisproject we perform several new deduplication constructions supporting authorized duplicate check in hybrid cloud architecture, in which the duplicate-check tokens of files aregenerated by the private cloud server with private keys. As aproof of concept in this project we implement a prototype of our proposed authorized duplicate check scheme and conducttestbed experiments on our prototype. From this project we show that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer. Futures work: It excludes the security problems that may arise in the practical deployment of the present model. Also,it increases the national security. It saves the memory bydeduplicating the data and thus provides us with sufficientmemory. It provides authorization to the private firms andprotects the confidentiality of the important

REFERENCES

[1] C. Ng and P. Lee. Revdedup: A reverse deduplicationstorage system optimized for reads to latest backups. InProc. of APSYS,Apr 2013.

[2] J. Li, X. Chen, M. Li, J. Li, P. Lee, andW. Lou. Securededuplication with efficient and reliable convergent keymanagement. In IEEE Transactions on Parallel andDistributed Systems, 2013.

[3] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl.Asecure datadeduplication scheme for cloud storage.InTechnical Report, 2013.

[4] J. Xu, E.-C.Chang, and J. Zhou. Weak leakage-resilientclient-side deduplication of encrypted data in cloudstorage. In ASIACCS,pages 195–206, 2013.

[5] J. Yuan and S. Yu. Secure and constant cost publiccloud storage auditing with deduplication. IACRCryptology ePrint Archive, 2013:149, 2013.

[6] K. Zhang, X. Zhou, Y. Chen, X.Wang, and Y. Ruan.Sedic: privacy aware data intensive computing on hybridclouds. In Proceedings of the 18th ACM conference onComputerandcommunications security,CCS'11, pages515–526, New York, NY, USA, 2011. ACM.

[7] M. Bellare, S. Keelveedhi, and T. Ristenpart.Dupless:Serveraided encryption for deduplicated storage. InUSENIX Security Symposium, 2013.

[8] M. Bellare, S. Keelveedhi, and T. Ristenpart.Messagelockedencryption and secure deduplication. InEUROCRYPT, pages 296–312, 2013.

[9] M. W. Storer, K. Greenan, D. D. E. Long, and E. L.Miller. Secure data deduplication. In Proc. of StorageSS,2008. [10]P. Anderson and L. Zhang. Fast and secure laptopbackups with encrypted de-duplication. In Proc. OfUSENIX LISA, 2010.

[11]R. D. Pietro and A. Sorniotti.Boosting efficiency andsecurity in proof of ownership for deduplication. In H.Y. Youm and Y. Won, editors, ACM Symposium on