IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 5, May 2018

Block-Level Message-Locked Encryption for Secure Large File De-duplication

Soham Chaudhari[1], Kshitij Sawale[2], Sasmit Shinde[3], Anish Hakhu[4]

Student, Computer Department, Modern COE, Pune, Maharashtra[1,2,3,4]

Abstract: Cloud computing is the long dreamed vision of computing as a utility. Besides all the benefits of the cloud computing security of the stored data need to be considered while storing sensitive data on cloud. Cloud users cannot rely only on cloud service provider for security of their sensitive data stored on cloud. To achieve optimal usage of storage resources, many cloud storage providers perform de-duplication, which exploits data redundancy and avoids storing duplicated data from multiple users. System proposes a new approach to achieve more efficient deduplication for (encrypted) large files. Our approach, named Block-Level Message-Locked Encryption (BL-MLE), can achieve file-level and block-level deduplication, block key management, and proof of ownership simultaneously using a small set of metadata. The BL-MLE method can be simply completed to hold confirmation of storage, that makes it multi-purpose for secure cloud storage.

Keywords: Third Party Authenticator, AES Algorithm, RSA Algorithm, SHA 512 Algorithm, deduplication, Block-Level Message-Locked Encryption.

I. INTRODUCTION

Uploading large files would consume extensive bandwidth; source-based deduplication seems to be a better choice for large file outsourcing. Unlike target-based deduplication which requires users to upload their files regardless of the potential data redundancy among those files, source-based deduplication could save the bandwidth significantly by eliminating the retransmission of duplicated data. Deduplication system, the user firstly sends a file identifier to the server for file redundancy checking. If the file to-be-stored is duplicated in the server, the user should convince the server that he/she indeed owns the file. Otherwise, the user uploads the identifiers/tag of all the file blocks to the server for block-level deduplication checking. Finally, the user uploads data blocks which are not stored in the server. Proof-of-Ownership (PoW) is necessary for source-based deduplication. PoW is an interactive protocol between a prover (file owner) and a verifier (data server). By executing the protocol, the prover convinces the verifier that he/she is an owner of a file stored by the verifier. PoW protocol in which presents three schemes that differs in terms of security and performance. In the previous encryption data privacy, is opposing the deduplication happens file level and block level. The replicated copies of the same file is eradicated by file level deduplication .For the block level duplication which eliminates duplicates blocks of data that occur in non-identical files.

II. RELATED WORK

Data deduplication is the technique which used to reduce redundancy in the storage data, there are two types of strategies are used for deduplication purpose one is file level deduplication, block level data deduplication.[3] In file level deduplication, single instance storage is used to perform deduplication task. In block level data deduplication, data files are divided into blocks and these blocks are compared to either these blocks are contains same value or not. That way the task of data deduplication is performed.
In this paper, we present DARE, a deduplication-aware, Low-overhead resemblance detection and elimination scheme for data reduction in backup/archiving storage systems. DARE uses a novel approach, DupAdj, which exploits the duplicate-adjacency information for efficient resemblance detection.[1]

This paper [2] presents a method to copy the encrypted data accumulated on cloud based on ownership challenge and proxy re-encryption. It integrates cloud data deduplication with access control. It gives the performance based on extensive analysis and computer simulations.
A three-tier cross-domain architecture,[4] with an proficient and privacy-preserving big data deduplication in cloud storage referred to as EPCDD achieves both privacy-preserving and data availability, and resists brute force attacks. In addition, the accountability can taken into consideration to offer better privacy assurances than existing schemes.[4]
The paper [5] the protocol that prevents unauthorized access by using a secure proof of ownership protocol. The protocol uses authorize deduplicate check for hybrid cloud architecture.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 5, May 2018

III. PROPOSED SYSTEM

The architecture diagram of the proposed system is as shown in below figure.

1. Proof of Ownership

Data Owner uploads document, metadata, checksum on cloud after encryption using keys from Data Owner and Cloud Service Provider. Also, a copy of metadata and checksum is sent to Auditor.

2. Data Access Via Permission model

Registered users send access request and receive encrypted file if authorized. User calculates checksum to compare with original and reports to Data Owner if checksum mismatch occurs.

3. Prevention De-duplication

Avoid De-duplication
a. File Level
b. Block Level
Maintains the checksum of file data and block of file data and compare at the time of file upload to avoid De-duplication.

4. Proof of Storage by Third Party Authenticator (TPA)

Auditor Receives metadata after upload. Performs periodic or on-Demand integrity checks by sending challenges to Cloud Service Provider. On response from Cloud Service Provider, Auditor confirms response and reports status to Data Owner.

IV. PROPOSED ALGORITHM

A. AES Algorithm

AES is depending on a intended standard called as a substitution-permutation network, and is quick in both software and hardware.[8] Unlike its predecessor DES, AES does not use a Feistel network. AES is a variant of Rijndael which has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. The multiple of 32 bits, both with a minimum of 128 and a maximum of 256 bits.

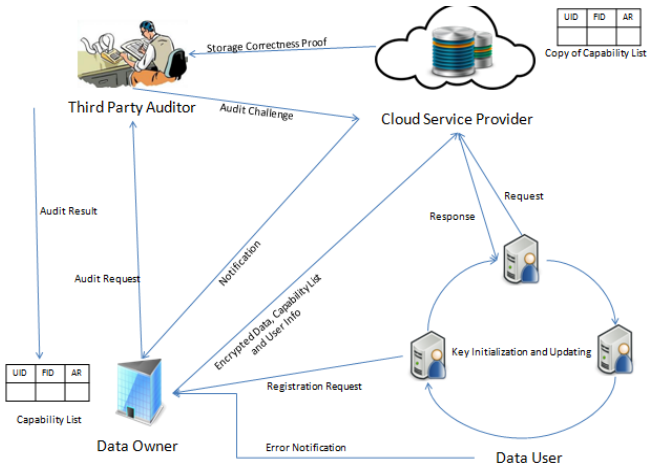


Fig.1 Architecture Diagram Of Proposed System

AES operates on a 4×4 column-major order matrix of bytes, termed the state, although some versions of Rijndael have a larger block size and have additional columns in the state. The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the cipher text. The number of cycles of replication is given:

- 10 cycles of replication for 128-bit keys.
- 12 cycles of replication for 192-bit keys.
- 14 cycles of replication for 256-bit keys.

Each round consists of several processing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform cipher text back into the original plaintext using the same encryption.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 5, May 2018

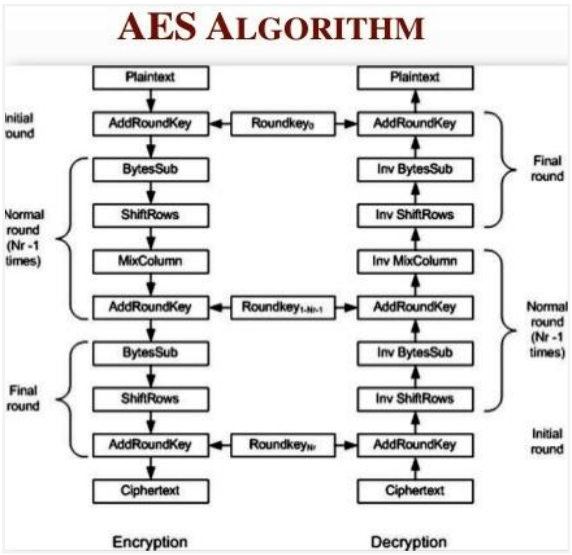


Fig.2 AES Algorithm Flowchart

B. RSA Algorithm

Key Generation

The keys for the RSA algorithm are generated the following way:

1. Choose two distinct prime numbers p and q.
- For security purposes, the integer's p and q should be chosen at random, and should be similar in magnitude but 'differ in length by a few digits to make factoring harder. Prime integers can be efficiently found using a primarily test.
2. Compute n = pq.
- n is used as the modulus for both the public and private keys. Its length, usually expressed in bits, is the key length.
3. Compute $\varphi(n) = \varphi(p)\varphi(q) = (p - 1)(q - 1) = n - (p + q - 1)$, where φ is Euler's totient function. This value is kept private.
4. Choose an integer e such that $1 < e < \varphi(n)$ and $\gcd(e, \varphi(n)) = 1$; i.e., e and $\varphi(n)$ are coprime.
5. Determine d as $d \equiv e^{-1} \pmod{\varphi(n)}$; i.e., d is the modular multiplicative inverse of e (modulo $\varphi(n)$)
- This is more clearly stated as: solve for d given $d \cdot e \equiv 1 \pmod{\varphi(n)}$
- e having a short bit-length and small Hamming weight results in more efficient encryption – most commonly $216 + 1 = 65,537$. However, much smaller values of e (such as 3) have been shown to be less secure in some settings.
- e is released as the public key exponent.
- d is kept as the private key exponent.

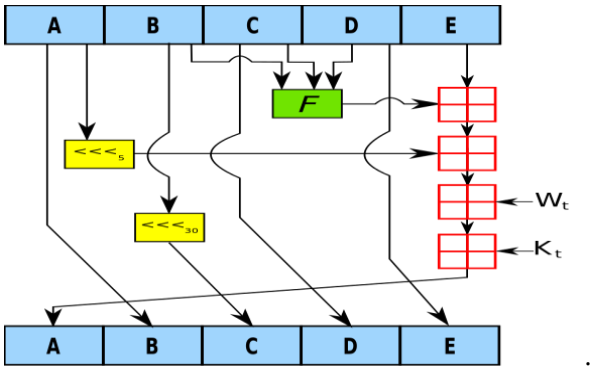
The public key consists of the modulus n and the public (or encryption) exponent e. The private key consists of the modulus n and the private (or decryption) exponent d, which must be kept secret. p, q, and $\varphi(n)$ must also be kept secret because they can be used to calculate d.

C. SHA Algorithm

In cryptography, SHA-1 (Secure Hash Algorithm 1) is a cryptographic hash function designed by the United States National Security Agency and is a U.S. Federal Information Processing Standard published by the United States NIST. SHA-1 produces a 160-bit (20-byte) hash value known as a message digest. A SHA-1 hash value is typically rendered as a hexadecimal number, 40 digits long. Image Description: One iteration within the SHA-1 compression function: A, B, C, D and E are 32-bit words of the state; F is a nonlinear function that varies; n denotes a left bit rotation by n places; n varies for each operation; Wt is the expanded message word of round t; Kt is the round constant of round t; denotes addition modulo 232.

The de-duplication, is one in which data redundancy is checked and avoids storing duplicated data from multiple users.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 5, May 2018



V. RESULTS

For using the system user have to login First.

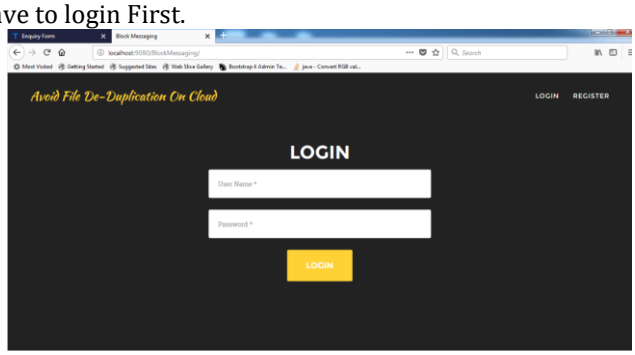


Fig.3 LogIn

Data Owner uploads document, metadata, the checksum on a cloud after encryption using keys.

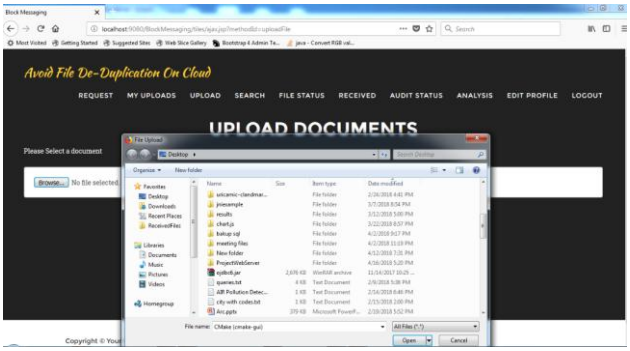


Fig.4 Upload Document

After login user profile is generated with all the details.

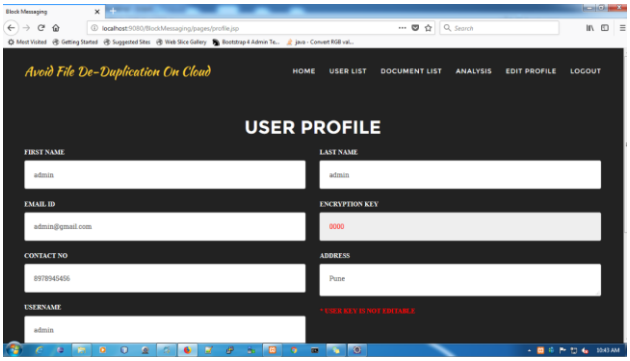


Fig.4 User Profile

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 5, May 2018

Auditor Receives metadata after upload. On response from Cloud Service Provider, Auditor confirms response and reports status to Data Owner.

Fig.5 Audit Report

The graph analysis for the unique and duplicate file is done as shown in figure 6.

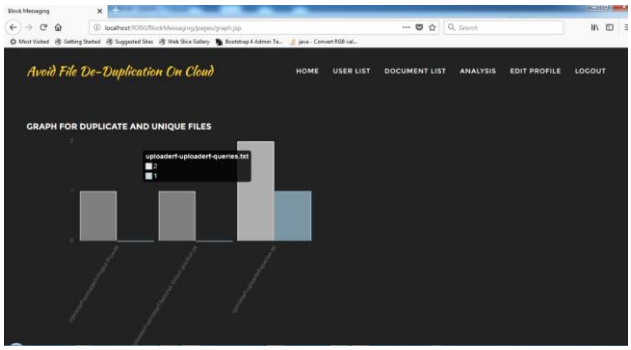


Fig.6 Graph for duplicate and unique files

VI. CONCLUSION

We are developed a system i.e. block-level deduplication which can provide more space savings than file-level deduplication does in large file storage. It is worth noting that for block level deduplication, the block size can be either fixed or variable. This system exploits data redundancy and avoids storing duplicated data from multiple users System focus on the block-level deduplication with fixed block size.

REFERENCES

[1] Wen Xia, Member,Hong Jiang "DARE: A Deduplication-Aware Resemblance Detection and Elimination Scheme for Data Reduction with Low Overheads", ,IEEE TRANSACTIONS ON COMPUTERS, VOL. 65, NO. 6, JUNE 2016.
[2] Zheng Yan, Wenxiu Ding,Xixun Yu,"Deduplication on Encrypted Big Data in Cloud",IEEE TRANSACTIONS ON BIG DATA, VOL. 2, NO. 2, APRIL-JUNE 2016.
[3] Rongmao Chen,Yi Mu,"BL-MLE: Block-Level Message-Locked Encryption for Secure Large File Deduplication",IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY.
[4] "Xue Yang,Rongxing Lu,"Achieving Efficient and Privacy-Preserving Cross-Domain Big Data Deduplication in Cloud",IEEE Transactions on Big Data.
[5] " Mr.Vinod B Jadhav ,Prof.Vinod S Wadne "Secured Authorized De-duplication Based Hybrid Cloud Approach" International Journal of Advanced Research in Computer Science and Software Engineering – 2014.
[6] Aparna Ajit Patil, Asst. Prof. Dhanashree Kulkarni "Block Level Data Duplication on Hybrid Cloud Storage System" International Journal of Advanced Research in Computer Science and Software Engineering – 2015.