ISSN:2319-6890)(online),2347-5013(print)
1 June 2016

A Technique for Resource-strained Devices By Refined Evidence of Retrievability Stratagem of Cloud Storage Services

Mrs. Ujjwala Bandawane , Mr. Sandeep Gore
G. H. Raisoni College of engineering and Management,Wagholi, pune
ujjwaladip@gmail.com, Sandeep.gore@raisoni.net

Abstract:For data integrity issue,previously, due to similar key among users decreases security.Our system implemented digital envelope which provides two keys ,asymmetric-key used for encryption of symmetric-key while the data is encrypted using symmetric-key and Multikeyword search and used TPA.Experimental outcomes between previous and Proposed security methods demonstrated.

***Keywords*—Cloud storage, Digital Envelope, Third party auditor, Integrity checking, Security.**

I. INTRODUCTION

Cloud computing is a service that allows on-demand network access, available, to a shared configurable computing resources. Cloud computing has less interference of organization or service provider communication. Sometimes many organizations requested to cloud for storage of large amount of information and that data has to be securely stored such as data contains personal information, health information and financial data. To maintain locally such large amount of information is challenging work as well as needs more maintenance cost. So, Cloud Service Provider gives Storage as a Service to decrease the overhead of large local data storage and also to decrease the cost by providing outsource data to the cloud due to the data owner store their confidential data on the cloud. Data owner has requirement of don't want their data to be corrupted as well as additionally we need more security concerns like security, integrity as well as appropriate access control.

Data security is obtained by encryption before storing it on the cloud server. Authors have tendency to propose data possession method for authentic data stored on server as well as for data integrity validating on cloud servers. PDP protocol is proposed for efficient authentication of the data integrity, such as POF (Proof of retrievability) was created. As compared with PDP, POF is a more proficient technique due to entire data file will be regenerated from parts of the data which is carefully stored on the servers. Additionally data stored on server is not able to modify by any unauthorized user i.e. access control should be specific as well as limited access of unauthorized users. Number of previous methods determined data owner as well as the storage servers are relate from the similar trust domain. So, this determination is no longer keeps if the data is outsourced to the cloud storage.

This method contains cloud server, data owners and data users. This method is also support different purposes. This system contains KDC and TPA. In that Digital envelop technique as well as also integrity checked respectively. In system, initially system has login to cloud server and requesting to KDC for key. KDC is generates master key and pair of public key and secret key is generated by using AES and ECC algorithm. Master key is encrypted by KDC utilizing ECC's public key of requested data owner and send the encrypted master key and secret key to data owner. After collecting key, data owner divides the file within blocks and encrypt them utilizing encrypted master key and send to the cloud server. At the same time hash of data blocks is generated and stored the metadata to TPA.

Client send request to third party authentication for file block security analysis and stored at cloud server. TPA stores the hash of blocks. It is tempting hash of specific file requests by client for security checking to cloud server. At the end received hash is of file block evaluated with hash store in its database. If the hash is equals, it sends the message to user, which shows that the files collected on server and it is not corrupted. There is trapdoor generated and in that important keywords are finding out by decryption of encrypted data.

The paper is organized as: section II shows the literature survey done by the researchers. Detail implementation details discussed in section III. Experimental result shows in the section IV. Finally paper is concluded with future enhancement and references used for the paper.

II. RELATED WORK

Jin Li, Xiao Tan, Xiaofeng Chen, Duncan S. Wong [1] are analyzed the problem of integrity of data storage in cloud. Authors are proposed the concept of public verifiability to mitigate the calculation overhead from client side at the same time of integrity authentication of data. To solve this issue in system OPoR a new technique is implemented which has two kinds such as server cloud storage and audit. Specifically, system only allows cloud audit server for preprocessing of the information rather than the cloud users before uploading to the cloud storage server and after that authenticated the data integrity.

H. Li, B. Wang, and B. Li [2] proposed privacy preserving techniques which permits open authenticating over shared data stored on the cloud. Specifically, they has tendency of trying ring signatures to assign the confirmed information determined which may observe the integrity of shared data. So, authors system provides security to individual users every block inside shared data from a Third Party Auditor. TPA is ready to freely check the authenticity of shared data without retaining the whole document. Experimental outcomes demonstrate the efficiency as well as ability of proposed system.

doi : 10.17950/ijer/v5s6/622

C. Wang, Q. Wang, and K. Ren [3] are determines that active data storage in distributed areas. Further, proposed challenge-response protocol may confirm the data integrity as well as search out the possible errors. Also they determined dynamic data operations on stored information. Additionally they explained the better methodology to preserve space for storing data de-duplication in cloud storage. The provable data control drawback helpful in cloud service suppliers and created a replacement separate integrity analysis method.

Jin Li, Xiao Tan, Xiaofeng Chen, Duncan S. Wong, and Fatos Xhafa [4] give an accomplice pragmatic affirmation subject for guaranteeing remote data security in cloud storage. The expected subject is set up ensured nearby reset attack within the fortified security model however supporting practical shared verifiability and dynamic data operations meanwhile foreseen a dynamic type of the past PDP method. In any case, the framework strengths from the prior beyond any doubt on the measure of inquiries and don't reinforce totally dynamic information operations.

C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou [5] propose a privacy preserving sharing review system for storing data securely in Cloud Computing. Authors are utilizes the homomorphic linear authenticator as well as in addition self-assertive covering to guarantee that the TPA would not see any results about the information content stored on the cloud server between the efficient reviewing process, which not just disposes of the largeness of cloud client from the dull and potentially luxurious looking at errand, moreover decreases the clients apprehension of their outsourced information leakage.

J. Li, X. Tan, X. Chen, D. Wong, and F. Xhafa [6] proposed a public review method in resource-constrained devices. Resource-constrained devices are a fundamental and lightweight creation. Along these lines, these devices have low calculation and also storage limit. Of course, these devices can perform high mobility which permits clients to go on and enough to use. Taking after the client may require endlessly modified information in cloud storage benefit, this operation needs to process in each update.

C. Liu, J. Chen, L. T. Yang, X. Zhang, C. Yang, R. Ranjan, and K. Ramamohanarao [7] describes that past analysis are not efficient in component information upgrade because of fixed size upgrade. Along these lines, they propose a course of action which can support variable-size block in dynamic information upgrade. DR-DPDP is a plan that gives clear movement and replication of client data more different servers. There are three contents in the framework. The client, who stores data on the CSP, challenges the CSP to guarantee the security of data and update of the stored data.

G. Ateniese, R. Smolders, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Melody [8] illustrated the Provable Data Ownership model that will give public auditability and sure responsibility of files on unreliable storage. They utilize RSA based homomorphic verifiable labels to study outsourced data. Their method is first gives blockless check and public certainty in the similar time. Regardless, Ateniese et al's. plan can't support dynamic information affirmation in the way that their plan only determines static information condition that proposes the client stores outsourced information and won't modify it.

III. IMPLEMENTATION DETAILS

A. *System Overview*

In this method cloud server, data owners and data users are participants. This method supports additional number of purposes. This system contains KDC and TPA. In that Digital envelop technique as well as also integrity checked respectively. In system, initially system has login to cloud server and requesting to KDC for key. KDC is generates master key and pair of public key and secret key is generated by using AES and ECC algorithm. Master key is encrypted by KDC utilizing ECC's public key of requested data owner and send the encrypted master key and secret key to data owner.

After collecting key, data owner divides the file within blocks and encrypt them utilizing encrypted master key and send to the cloud server. At the same time hash of data blocks is generated and stored the metadata to TPA.

Client send request to third party authentication for file block security analysis and stored at cloud server. TPA stores the hash of blocks. It is tempting hash of specific file requests by client for security checking to cloud server. At the end received hash is of file block evaluated with hash store in its database. If the hash is equals, it sends the message to user, which shows that the files collected on server and it is not corrupted.

In addition to system Multi-keyword search is performed over encrypted data stored at cloud server with the use of trapdoor ; which is generated at the time of data uploading.

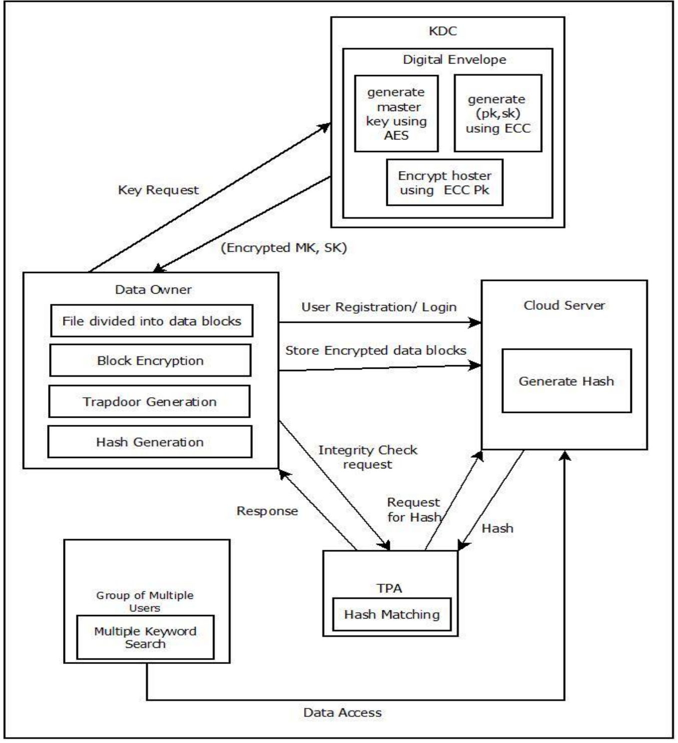

Fig. 1: System Architecture

B. Algorithm

Algorithm 1: AES Algorithm

The algorithms are used as a portion of AES and easier to indicate that they will be executed implementing cheap processors and less amount of memory. Most efficient implementation was a key variable in its selection as the AES cipher.

Step in AES Algorithm:
1. Key Expansion: - Use the Rijndael's key schedule Round keys which are derived from the cipher key.
2. If Dist_to_tree(u) > Dist_to_tree(DCM) and First-Sending(u) then
3. Initial Round: - Add_Round_Key where every byte of the state is combined with the round key utilizes bitwise XOR.
4. Rounds
Sub_Bytes: non-linear substitution step.
Shift_Rows: transposition step.
Mix_Columns: mixing operation of each column. Add_Round_Key
5. Final Round: It contain SubBytes, ShiftRows and Add_Round_Key

Algorithm 2: ECC Encryption

Elliptic curve cryptography (ECC) is a solution to solve with public key cryptography in case of the logarithmic structure of elliptic curves on finite fields. Elliptic curves are also used as a part of a many integer number factorization computations that have applications in cryptography, for example, Lenstra elliptic curve factorization.

- Key Generation

Key generation is an essential step in that we create both public key and private key pair. The Alice will encrypt the message with bob's public key and the bob will decrypt the cipher text utilizing its private key. Now, we have to choice a number as 'd' within the range of 'n'. Using the following equation:

$Q = d * P$

d be any random number within the range of (1 to n-1).
P is the point on the curve.
Q is the public key.
d is the private key / secrete key.

- Encryption

Let 'M' be the message that Alice wants to send to the bob. For this Alice has to represent this message on the curve. This has in-depth implementation details. let 'm' has the point 'M' on the curve 'E'.
Select 'k' be any Random of range [1 - (n-1)].
The output of the step is two cipher texts which are CT1 and CT2.

$CT1 = k*P$

$CT2 = M + k*Q$

CT1 and CT2 will be sent to bob.

- Pre-Processing

In this procedure by decrypting data steaming as well as stop words are removed. In cloud computing, stop words are words which are filtered out before or after processing of natural language data (text). Though stop words usually refer to the most common words in a language, there is no single universal list of stop words used by all natural language processing tools and indeed not all tools even use such a list. Some tools specifically avoid removing these stop words to support phrase search.

- Trapdoor (Key, w)

This algorithm is run by the user who has key to perform a search. It takes as input the searchable encryption key and a keyword set w, then outputs only one trapdoor T_r.

- Decryption

Bob wants the original 'm' that was send by Alice, Bob performs following steps to get original message 'm'.

$M = CT2 - d * CT1$

M is the original message that is send by Alice.

$M = CT2 - d * CT1$

'M' be represented as 'CT2 - d * CT1'

$CT2 - d * CT1 = (M + k * Q) - d *(k*P)(CT2 = M + k * Q$ and $CT1 = k * P)$

$= M + k * d * P - d * k * P$ (canceling out $k * d *P$)

$= M$ (Original Message)

Algorithm 3: Digital Envelope

Step 1: Get user request Ui for key generation.
Step 2: Run algorithm 1 (AES key generation)
Get Master key MK
Step 3: Run algorithm 2(ECC key generation)
Get key pairs (PK, SK)
Step 4: Encrypt MK using PK
Get encrypted MK as PK'
Step 5: Send PK' and SK to requested User Ui.

C. Experimental Setup

The system is built using Java framework (version jdk 1.8) on Windows platform. The Netbeans (version 8.0) is used as a development tool. The system doesn't require any specific hardware to run, any standard machine is capable of running the application. The system analysis is carried out on datasets consisting of files.

IV. RESULT AND DISCUSSION

A. Data Set

Number of files is utilized for this work. The Input Files are of various sizes varying from 1 KB to 100MB.

B. Result

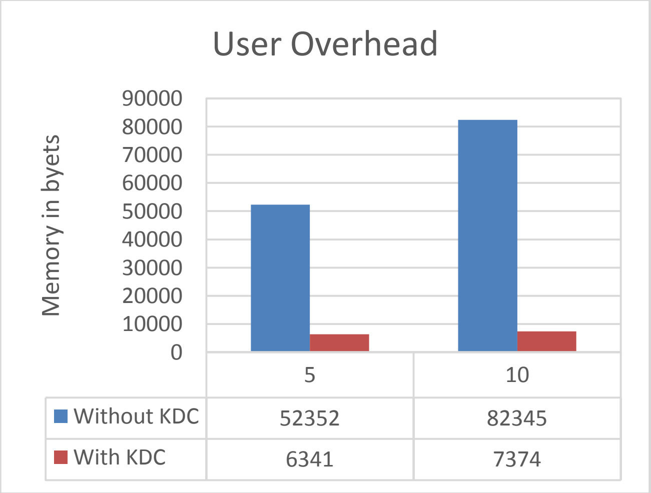

Fig. 2: User overhead graph comparison.

Figure 2 demonstrates that, the system with KDC decreases the memory overhead in bytes than the without KDC system.

Fig. 3 demonstrates the comparison between proposed system and existing system in case of security. Proposed system enhances the security due to utilization of digital envelope technique that provide key to the users. X-axis indicates the existing system and proposed system respectively and Y-axis represents the security in percentage.

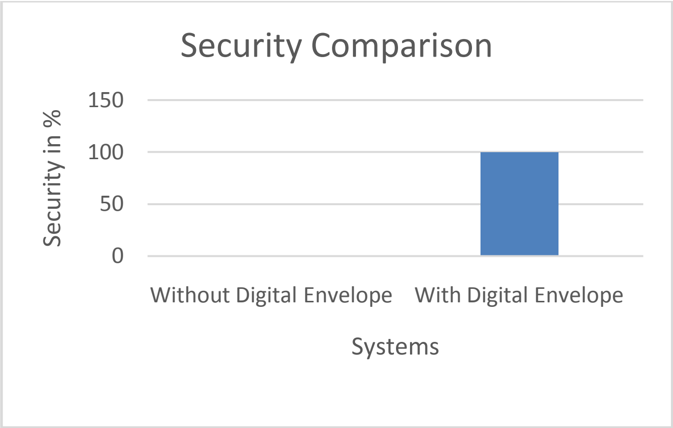

Fig. 3: Security Comparison

V. Conclusion And Future Scope

This paper presents methodology for a cloud-based storage for providing secure access control to data owners and dynamic operations . In KDC, digital-envelope encrypts data by individual's asymmetric-key to enhance the security,Multi-keyword searching on encrypted data. Experimental results demonstrates digital envelope is stronger on the for security & access control.

VI. REFERENCES

i. Jin Li, Xiao Tan, Xiaofeng Chen, Duncan S. Wong, "OPoR: Enabling Proof of Retrievability in Cloud Computing with Resource-Constrained Devices", IEEE TRANSACTIONS ON CLOUD COMPUTING, VOL. 3, NO. 2, APRIL/JUNE 2015.

ii. H. Li, B. Wang, and B. Li, "Oruta: Privacy-preserving public auditing for shared data in the cloud", IEEE Trans. Cloud Compute., vol. 2, no. 1, pp. 43-56, Jan. Mar. 2014.

iii. C. Wang, Q. Wang, and K. Ren, "Ensuring data storage security in cloud computing", in Proc. 17th Int. Workshop Quality Serv., 2009, pp. 1-9.

iv. Jin Li, Xiao Tan, Xiaofeng Chen, Duncan S. Wong, and Fatos Xhafa, "OPoR: Enabling Proof of Retrievability in Cloud Computing with Resource-Constrained Devices", IEEE Transactions on cloud computing, vol. 3, no. 2, April/June 2015.

v. C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy preserving public auditing for secure cloud storage," IEEE Transactions on Computers, vol. 62, no. 2, pp. 362-375, 2013.

vi. J. Li, X. Tan, X. Chen, D. Wong, and F. Xhafa, "OPoR: Enabling proof of retrievability in cloud computing with resource-constrained devices," accepted and to be publish in IEEE Transactions on Cloud Computing, Oct. 2014.

vii. C. Liu, J. Chen, L. T. Yang, X. Zhang, C. Yang, R. Ranjan, and K. Ramamohanarao, "Authorized public auditing of dynamic big data storage on cloud with efficient verifiable fine-grained updates," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 9, pp. 2234-2244, 2014.

viii. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores", in Proceedings of the 14thACM Conference on Computer and Communications Security, pp. 598ˆa609, Virginia, USA, 2007.

ix. G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, ˆaScalable and efficientprovable data possession,ˆa in Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks, pp. 9:1ˆa9:10, Istanbul, Turkey,2008.

x. Kevin D. Bowers, Ari Juels, Alina Oprea, Proofs of Retrievability: Theory and Implementation, CCSWˆa09, Journal of Systems and Software, v.85 n.5, p.1083-1095, May, 2012.