Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 13, Number 7 (2017), pp. 3455-3466

http://www.ripublication.com

Re-Encryption Based Data Forwarding in an Erasure-Coded Cloud Storage System

N. Abirami

Student, Department of Computer Science and Engineering
Sathyabama University, Chennai, Tamil Nadu, India.

S. Murugan

Professor, Department of Computer Science and Engineering
Sathyabama University, Chennai, Tamil Nadu, India.

Abstract

Currently, the business processes of information technology have adapted to use cloud computing services considering scalability, cost efficiency, low marketing-time and availability features. Cloud computing has emerged as a new technology by offering computational resources such as software, virtual memory, storage space as its services with the main advantage that the services are accessible over internet. The services offered by cloud are web-based, which enables the customer to avail the services from anywhere in the world. So, cloud simply changes the way the user accesses his application or stores is information. When information is moved from in-house data center to cloud storage the customer is greatly concerned about the security issues. Apart from security it is also concerned about whether it would be able to retrieve the information while needed, hence availability and robustness of data also needs to be addresses. In this paper, a secured data forwarding functionality using re-encryption in an erasure code-based cloud storage environment is proposed. The system uses encryption for secured storage of data and uses re-encryption method for securely forwarding the data to another user. The key importance of the proposed system is that it does not store the plaintext and its corresponding cipher text directly in the cloud storage system, rather it stores only the partitioned blocks of the cipher text. When the file is forwarded, only the re-encrypted cipher text is stored in the inbox of the user. So, both storing and forwarding operations are performed under the control of the owner. The experimental implementation of the system shows successful retrieval of stored and forwarded messages with less storage and computational costs.

Keywords: Cloud Storage;data forwarding;re-encryption;security

I. **INTRODUCTION**

Cloud computing is a computational model that offers on-demand services which are accessible over the internet with less interaction between the customer and the cloud service provider. The numbers of customers have grown at immense rate during the past few years due to the opportunities the cloud has created in the market. Cloud allows the organizations to focus on their core business rather than building infrastructure with less cost spent on infrastructure management.

Cloud computing offers the services which include both hardware and software based services through a network of systems interconnected. The delivery models of cloud can be classified as:

a) *Public Cloud* is one which offers services to many organizations and individuals. The public cloud offers free-trial services and payment is mostly based on the services used or on monthly subscription basis. The public cloud is owned by organizations such as Amazon Cloud services, Microsoft Azure, Google Cloud Service and so on.

b) *Community cloud* is shared by a group of users belonging to a community based on some common requirements. Community cloud services may be accessed by many organizations.

c) *Private Cloud* is one which offers service to single private organization. The servers, networks, storage devices, software are configured based on proprietary architecture. The services offered are customized based on the factors such as speed of service, security parameters, network bandwidth, storage space etc.

d) *Hybrid Cloud* [1] is a combination of multiple public and private clouds.The architecture of hybrid cloud handles the complexityofapplication distribution across the public and private clouds.

The services offered by cloud can be classified as:

a) Software as a Service (SaaS):

The software services are accessed through web. Here, the users need not perform any manual up gradation of software since it is taken care of by cloud provider automatically. Cloud provider provides some Application Programming Interface (API) to build software components and their integration. SaaS was initiated by SalesForce Customer Relationship Management (CRM) product. The users access these services on a pay-per-use basis.

b) Platform as a Service (PaaS):

PaaS provides development, deployment and testing environment for application developers. This platform for application deployment is provided as services by cloud. The user can focus on application development and the resource management is provided by cloud. Some additional PaaS services includes API containing functions for management of platform based services and persistent storage.

c) Infrastructure as a Service (IaaS):

IaaS[2] services provides virtual servers, storage space, operating system of user's choice and network communication. The customers can access infrastructure requirements as a service from cloud provider rather than investing cost and time by buying large scale servers, installing and maintaining them. IaaS is commonly used by start-up organizations and organizations that do not want to spend much cost on resources.

Despite the services provided by cloud, security is of great concern, when the customers migrate the data from their own data centers to remote cloud servers. Some of the security concerns from the user point of view:

- The user does not have control on the exact location in which the information is stored.
- The inability of the user, to excise control over the security aspects of the systems in which the application is deployed.
- Customers are at the high risk of data loss and may not bet have complete control on their data.
- The downtime period of cloud servers may affect the availability of data.

II. RELATED WORK

Attribute-Based Encryption [3] proposes an attribute-based encryption scheme where the data to be stored is encrypted and the cipher text is tagged with some attributes and the private-key contains information which provides details about to which cipher text the user is allowed to decrypt. The user is allowed to decrypt the cipher text based on the access information in the private-key.

The disadvantage of the system is that there is no functionality in which the attributes associated with the cipher texts are hidden, which possess a great security risk.

Secured data deduplication scheme [4] applies encryption techniques to less popular data and applies deduplication techniques to more popular data. The transition from encryption to deduplication is performed at storage server in case of popular data. This scheme provides difference protection levels according to the popularity of the content stored by the users. The classification of data as popular or non-popular is based on indexing service. However, the security of the unpopular data is slightly compromised and there is no dynamic set up of threshold value when the file is encrypted.

In Privacy-preserving [5] the authors have emphasized the importance of data integrity through third-party public auditing to check the data being transferred along with the cloud storage. They have used homomorphic linear authenticator and random masking to ensure that the content of data is not known to the authenticator. The performance is reduced in the case of batch auditing.

In Conditional Proxy Re-encryption [6] scheme, the decryption of the cipher text is forwarded to the recipient based on the condition set by the owner of the file. The security of the system is proven using –quotient Diffie-Hellman assumption model. In order to forward the message to a recipient the owner encrypts the message using the owner's public key and gives the re-encryption key to the proxy for decryption along with the condition key. Decryption is possible for those cipher text based on the availability of the condition key to the cipher text.

Identity-based conditional proxy re-encryption (IBCPRE) [7] method, a conditional tag is attached to the cipher text and the re-encryption key. The decryption of the cipher text is successful only if both the condition test applied to the re-encryption and the cipher text is passed successfully. In IBCPRE scheme, the public key of a user is a string which represents is identity such as SSN or employee id. One of the main advantages of IBCPRE is that the owner need not know in advance about the recipients to whom the cipher text needs to be shared. The message delegation can be done after the owner has encrypted the original message and uploaded it in the cloud server. Another advantage is that it supports one-to-many encryption by generating multiple re-encryption key and sending it to the server.

In Time and ID-Based proxy re-encryption [8] scheme the cipher text of the owner is generated by combining the owner's public key and time intervals which includes date and time. The classification of the owner's data is done internally using the owner ID followed by the requestor's time interval.

III. PRELIMINARIES

i. Erasure Coding

One way in which erasure coding can be implemented is by using Reed-Solomon codes [9]. Reed-Solomon (RS) is an efficient error correcting code)[10] which successfully retrieves the original message in case of one or more storage server failures.

The series of operations performed by RS encoder is illustrated below:

i. The encoder reads the 'k' data symbols where each symbol is of size 's' bits.
ii. Compute the parity bits 't'.
iii. Append 't' parity bits to 'k' data symbols and produces the output of 'n' data symbols where n=k+t.

The series of operations performed by the decoder is illustrated as follows:

i. Receives the 'n' codeword symbols as input.
ii. Compute the polynomial structures n(x) for each 'n' symbols.
iii. Determine the number of errors; if errors are found find the error locations 'xi' by using Berlekamp-Massey algorithm [11].
iv. Form the error locator polynomial L(x).
v. Find the polynomial roots using Chien Search algorithm [12].
vi. Find the error values 'y_i' and correct them using Forney's algorithm [13].

vii. The recovered code word symbols are produced as output.

ii. Proxy re-encryption

Proxy re-encryption is one of the methodologies adapted to implement access control in cloud computing applications. In order to understand proxy re-encryption we consider the scenario in which an owner 'A' stores his file in the cloud storage and later on decides to send the file to another recipient 'B'. Here, the owner 'A' represents the person who owns the file and the recipient 'B' represents the person who request for a particular file uploaded in the cloud server. The owner generates a single key-pair and sends it to the proxy server. The proxy server performs re-encryption for all the cipher texts uploaded by the owner 'A' and delegates to recipient.

But in many scenarios the owner may decide to implement control over his data by allowing only designated users to view only a subset of data. In this scenario the owner may generate multiple key-pairs for every subset of data and forward the data through proxy. But these methodologies are not possible in practical applications. The alternative approach is to re-encrypt a subset of cipher text and forward it to the specific user.

Properties of Proxy re-encryption scheme:

Delegation property:

Delegation [14] is a function in which re-encryption key is generated which is a combination of message recipient private key and the owner's public key.

Bi-directional scheme:

The re-encryption message is used to delegate messages from owner to recipient and vice-versa.

Uni-directional scheme:

In this scheme, the re-encryption of message occurs from owner to recipient and not the reverse.

Transitivity:

In this scheme, the re-encryption is applied to the cipher text multiple times.

IV. SYSTEM MODEL

A. Symbolic Notations and Definitions

For the understanding of the description of our model we have summarized the symbols used and their definitions in Table 1.

TABLE 1. SYMBOLIC NOTATIONS USED IN THE DESCRIPTION OF THE MODEL

SYMBOL	DEFINITION
n	Number of blocks
f	File to be stored
s	Size of the file
CS	Cloud storage
P	Proxy system
O	Owner of the file
U	User who receives the file
e_k	Encryption key used for file storage
C_k	Cipher text obtained using e_k
m	Plaintext
e_{k1}	Encryption key used for file forwarding
c_{k1}	Cipher text obtained using e_{k1}
p_k	Public key
s_k	Secret key
c_{tn}	Multiple blocks of cipher text
rc_p	Re-encrypted cipher text

B. Secured cloud storage and secured forwarding application (SCSFA)

We choose a public cloud service provider to build our SCSFA. The public cloud infrastructure comprises of network resources, hardware, software, servers and operating system, as a whole they provide a virtual environment for the user to access the services. Users access the services through insecure public internet system. The users may choose a file from his desktop and upload it in the storage space and download it later when required and also share the file to another user. Storing the data in single cloud system is at the risk of storage server failure. The main objective of SCSFA is storing the files in the public cloud in a secured manner. Our security mechanism is incorporated in such a way that the data is secured while it is stored in the public storage and also secured while it is being forwarded to another user. In order to avoid failure issues multiple copies of data can be stored in number of storage systems which is called replication, but it causes increased storage cost. Erasure code [15] is adapted as an alternate solution to replication due to its higher reliability and less storage cost. Erasure coding is a data reliability mechanism where input data is

partitioned into multiple blocks and each block is stored in a cloud storage server. By this way the plain-text is not directly uploaded in the cloud storage but as multiple blocks. One common method to store a file securely in a distributed file system is through encryption. In SCSFA we encrypt the plain-text as an initial step and then partition the cipher-text into multiple blocks and store them in the cloud storage system. So we ensure efficient storage security by partitioning the cipher-text rather than the plain-text.

For forwarding functionality, we provide an option for the owner to select the file which he wishes to share and the user to whom the file is intended to be forwarded. We generate a re-encryption key which is composed of encryption key and public key. When the owner selects the file to be shared, encryption key is used for first-level encryption and public key is used for re-encrypting the cipher text. When the owner selects the user, a private key is generated by SCSFA and is sent to the respective user which is used for retrieving the file forwarded to him.

C. Threat Model

We analyze the security and privacy issues in the cloud environment. The proxy referred in our system is the public cloud system, which is a semi-trusted entity. So, we strongly ensure that the data stored in cloud is not accessed by insiders, namely the cloud administrator who has full permission to access the data or by other unauthorized person. We consider the following security requirements while the data is stored and forwarded to another user:

ii. The cloud system is a semi-trusted entity so it should not get access to any information from the plaintexts stored.

iii. The cloud system should not get access to the keys generated so that it cannot decrypt the cipher text.

iv. The user should be able to decrypt the cipher text only after the re-encryption operation is performed.

V. SYSTEM OPERATION

a) Owner and user authentication with Cloud

The owner and the user register with the public cloud service provider and obtain a user name and password credentials for subsequent login. On successful registration they are allocated storage space in the public cloud.

b) Owner and user authentication with SCSFA

The owner and the user undergo registration process on their first attempt to access the SCSFA. On successful registration the owner is allowed to perform upload, download and forward operations, whereas the user is provided with an inbox which contains the files forwarded to him.

c) Storing the file in Cloud

Encryption

The owner chooses a file f, to be uploaded in cloud. The file is first encrypted using Enc(f,e_k) algorithm, as a result cipher text c_k is obtained.

- *Enc (f,e_k)*

Input – File f chosen by the owner o.

Output – Cipher text c_k

Partition

The file is partitioned into n blocks, where each block is of same size. Each block of the file is stored in their respective directories created in the CS. Suppose if n=4, the file is partitioned into 4 blocks and c1, c2, c3, c4 are created.

- *Split*(c_t,n)

Input: 'ct' is the cipher text encrypted using 'e_k' and 'n' is the number of chunks

Output: c_{tn}-multiple parts of cipher-text are generated. If n=4 then the cipher texts generated are c_{t1},c_{t2},c_{t3} and c_{t4}

d) Re-encryption

Re-encryption is performed when the owner decides to share his file to another user. Public, secret key pairs (p_k,s_k) are generated using Keygen() algorithm .The SCSFA provides the list of files available in the CS to the owner. The owner chooses the file and the user who is intended to receive the file. The file chosen by the owner is encrypted using Enc1(f,e_{k1}) producing the output c_{k1}, which is re-encrypted using Reenc(p_k,c_{k1}) producing output rc_p .This rc_p is stored in the inbox directory of the user.

- *Keygen()*

Output: p_k,s_k where p_k and s_k are public and secret key pairs for user u.

- *Enc1(f,e_k)*

Input – File f chosen by the owner o,e_k is the encryption key used to encrypt the file f.

Output – Cipher text c_k

- *Reenc($p_{k,,}c_k$)*

Input – p_k is the pubic key and c_k is the cipher text

Output – rc_p is the re-encrypted cipher text.

e) Retrieving the file

Merging

File retrieval is performed either by the owner or the user. Once the owner selects the file which he wishes to retrieve from the CS, he provides the dec_k. The file stored in the cloud as multiple chunks is merged and downloaded in the local desktop of the user.

- *Merge(c_{tn},n)*

Input – c_{tn} is the multiple blocks of cipher text, n is the number of chunks

Output – c_k is obtained as a result of merging c_{tn}

Decryption

When the user wishes to retrieves the file forwarded to him, he accesses his inbox in the SCSFA and selects the file which he wishes to retrieve. Then he provides the secret s_k key received from SCSFA, the file is decrypted and downloaded to the desktop of the user.

- *Dec(r_{cp},s_k)*

Input – r_{cp} is the re-encrypted cipher text, s_k is the secret key.

Output – plaintext file fs

VI. IMPLEMENTATION AND PERFORMANCE EVALUATION

The experimental setup comprises of desktop computer with 32-bit 2.50 GHz Intel Core i5 processor and 16 GB RAM, running Windows 7 ultimate operating system. The public cloud service provider used in the system is CloudMe[16]. The algorithms described were implemented in Java language using SunJCE provider[17] with PKCS5 padding[18].

The performance analysis has been evaluated based on storage cost, computation cost and communication cost.

Storage Cost:

Let *s1* and *s2* be size of bits belonging to the groups *G1* and *G2*.As discussed already CS stores n blocks of cipher text. So, the total cost computed is (1+2s1+s2+ns3) bits which is much lesser than the original file size *s*. Thus the system has less storage cost.

Computation Cost:

The computational cost of the operations involved in the model such as pairing, modular exponentiation in *G1*, modular exponentiation in *G2*, modular multiplication in *G1*, modular multiplication in *G2*and arithmetic operation is also computed. The

parameters used in the computational operations are listed in Table 2 and the computational cost is tabulated in Table 3.

TABLE 2. THE PARAMETERS USED FOR COMPUTATIONAL OPERATIONS

Operation	Parameter
Pairing Operation	PRG
Modular exponentiation in G1	ME_1
Modular exponentiation in G2	ME_2
Modular multiplication in G1	MM_1
Modular multiplication in G2	MM_2
Arithmetic operation in GF(p)	F_p

TABLE 3. COMPUTATIONAL COST OF ALGORITHMS

Operation	Computation cost
Enc	$1PRG+1ME_1+1MM_2$
Split	$n^2ME_2+(n-1)MM_2+O(n^3)F_p$
Keygen	$1ME_1$
Enc1	$1PRG+1ME_1+1MM_2$
Dec	$1ME_1$
Merge	$nME1+O(t^2)F_p+(n-1)MM_1$
Reenc	$1PRG+1MM_1$

For the analysis of computational cost, we consider the cost of *n* blocks of messages for storing and retrieving. The cost of arithmetic operation F_p is much lower than M_1 and M_2.The exponent operations E_1 and E_2 are performed in terms of multiplication and squaring operations since the computation time for the exponent operation is 1.5[log *p*] times greater than the computation operation of M_1 and M_2 with respect to groups G_1 and G_2.

While performing the storing operation, the owner runs Enc(.) algorithm, where generating c_k cipher text requires a PRG operation, a ME_1 operation and a MM_2 operation. So, the computational cost is ($PRG+ME_1+MM_2$). In the Spilt(.) algorithm

for generating n blocks of cipher text c_{tn}, the computational cost requires (n^2ME_2+(n-1)MM_2+O(n^3)F_p) .

While performing the re-encryption operation, the owner runs the Reenc(.) algorithm, where the cipher text r_{cp} is generated ,which requires a PRG and a MM_1 operation. The keygen() algorithm requires one ME_1 operation.

In the case of retrieval operation the owner runs the Merge(.) algorithm, which involves retrieving n blocks of message from cloud storage and requires O(t^2)F_p, $n$$ME_1$ and *(n-1)* MM_1.The Dec(.) algorithm involves decrypting the cipher text r_{cp} and requires one ME_1 operation.

VII. CONCLUSION

In this paper, the problem of secured data storage and secured data forwarding in public cloud environment is investigated. In the proposed system it is not stored the cipher text directly in the cloud storage; rather the cipher text is split into multiple blocks and then store it. For forwarding the data to another user re-encryption is applied and send the re-encrypted cipher text to the inbox of the user. The proposed implementation shows the successful retrieval of data, less storage and computational cost.

REFERENCES

[1] Rajkumar Buyya, James Broberg, Andrzej Goscinski - Cloud Computing Principles and Paradigms pg:no-26.

[2] Borko Furht,Armando Escalante;Handbookof Cloud Computing ISBN 978-1-4419-6523-3 e-ISBN 978-1-4419-6524-0 , Springer New York Dordrecht Heidelberg London;pgno-7,DOI 10.1007/978-1-4419-6524-0

[3] Vipul Goyal,Omkant Pandey,Amit Sahai and Brent Waters,"Attribute-based encryption for fine-grained access control of encrypted data," CCS '06 Proceedings of the 13thACM conference on Computer and communications security,pp 89-98,Nov. 2006, DOI>10.1145/1180405.1180418

[4] Jan Stanek, Alessandro Sorniotti, Elli Androulaki, and Lukas Kencl,"A Secure Data Deduplication Scheme for Cloud Storage",International Conference on Financial Cryptography and Data SecurityFC 2014: Financial Cryptography and Data Security pp 99-118,DOI: 10.1007/978-3-662-45472-5_8

[5] Cong Wang, Sherman S.-M. Chow, Qian Wang, Kui Ren, and Wenjing Lou,"Privacy-Preserving Public Auditing For Data Storage Security In Cloud Computing", 2010 Proceedings IEEE INFOCOM,pp 362-375 DOI: 10.1109/INFCOM.2010.5462173

[6] Jian Weng,Robert H. Deng,Xuhua Ding,Cheng-Kang Chu and Junzuo Lai,"Conditional proxy re-encryption secure against chosen-ciphertext

attack",ASIACCS '09 Proceedings of the 4th International Symposium on Information, Computer, and Communications Security,pp 322-332,Mar.2009,doi>10.1145/1533057.1533100

[7] Jun Shao,Guiyi Wei,Yun Ling,Mande Xie,"Identity-Based Conditional Proxy Re-Encryption",2011 IEEE International Conference on Communications (ICC),July 2011,DOI: 10.1109/icc.2011.5962419

[8] Kambombo Mtonga,Anand Paul and Seungmin Rho,"Time-and-ID-Based Proxy Reencryption Scheme",Journal of Applied Mathematics,May 2014,Volume 2014 ,http://dx.doi.org/10.1155/2014/329198

[9] C.K.P.Clarke," ReedSolomon Error Correction" BBC Research and Development-Whitepaper WHP 031 pgno:2

[10] W. Cary Huffman and Vera Pless,"Fundamentals of Error-Correcting Codes",Published in the United States of America by Canbridge University Press,New York,pp 168,2003, ISBN 1139439502

[11] G. H. Norton ,"The Berlekamp-Massey Algorithm via Minimal Polynomials",August 20, 2010 https://arxiv.org/pdf/1001.1597.

[12] El Habti El Idrissi Anas1 , El Gouri Rachid,Hlou Laamari1 ,Ibn Tofail Kenitra, ,"FPGA Implementation of A New Chien Search Block for ReedSolomon Codes RS (255, 239) Used In Digital Video Broadcasting DVB-T", Int. Journal of Engineering Research and Applications www.ijera.com ISSN : 2248-9622, Vol. 4, Issue 8(Version 1), August 2014, pp.82-86

[13] http://mobiledevdesign.com/news/error-control-coding-digital-communications-systems

[14] Sherman S. M., ChowJian Weng,Yanjiang Yang and Robert H. Deng,"Efficient Unidirectional Proxy Re-Encryption",Published in "International Conference on Cryptology in Africa,AFRICACRYPT 2010: Progress in Cryptology – AFRICACRYPT 2010" pp 316-332Volume 6055,DOI: 10.1007/978-3-642-12678-9_19

[15] Cheng Huang, Huseyin Simitci, Yikang Xu, Aaron Ogus, Brad Calder, Parikshit Gopalan, Jin Li, aand Sergey Yekhanin," Erasure Coding in Windows Azure Storage" Microsoft Corporation

[16] https://www.cloudme.com/

[17] http://docs.oracle.com/javase/7/docs/technotes/guides/security/SunProviders.html

[18] Chritopher Steel, Ramesh Nagappa ,Core Security Patterns:Best Practices ans Strategies for J2EE,™ Web Services and Identity Management