IJETIE
International Journal of Emerging Technology and Innovative Engineering
Volume I, Issue 3, March 2015
ISSN: 2394 - 6598
www.ijetie.org

Concealing Data Files in the Cloud
(Threshold Proxy Re-Encryption Scheme)

T. Mariya John
II Year M.E (CSE)
Mount Zion College of Engineering and Technology, Pudukkottai
email:tommarjohn@yahoo.co.in, Cell:9842519243

ABSTRACT

Network online storage of data is the present trend and next generation architecture of Computer Technology. Computing resources can be shared on demand through Internet network; it is convenient for all users due to cloud computing. The collection of storage servers and key servers enables the cloud computing process. Third party cloud storage cause data confidentiality, General encryption scheme limits its functionality only few operations are supported over encrypted data; the original codeword symbol is re-encrypted in a re-encryption scheme and integrated with decentralized erasure codeword designing distributed system. Without central authority, storage system distributed has to face many challenges due to multiple functions. Proxy re-encryption scheme supports encoded operations over forwarded operations and encrypted message. Threshold proxy re-encryption scheme integrates by decentralized erasure code such that a secure distributed storage system is formulated. This fully integrates encrypts, encodes, and forwards.

General Terms

Source code, Concealing, Data, Storage Server, Secure,

Keywords

Cloud storage system, encryption, decryption, Decentralized erasure code, proxy re-encryption, threshold , and key servers.

.

1. INTRODUCTION

Cloud computing is linked with every individual's need like email, online shopping, banking, marketing, studies, research, advertisement, examination result etc., the cloud offers great convenience for the users because they need not worry about the complexities of direct hardware and software management. The real fact is cloud computing posses various challenges and security threats. The data stored in cloud may need frequent updating, reordering, modifying or need verification by user for correctness. The protocols of storage correctness assurance are important to achieve robust and secure cloud storage.

Threshold proxy re-encryption scheme supports secure data forwarding storing and retrieving it is robust since they last until storage server survives. Decentralised erasure code in distributed storage server tolerate the failure server threshold of the erasure code storage servers data can be recovered by decoding the codeword symbols in storage servers.

2. EXISTING SYSTEM DISADVANTAGE

Third party cloud systems are expertise capabilities who are mediators to assess and expose risk of cloud storage service on behalf of users upon request these cause serious concern on data confidentiality in many situations.

General encryption scheme in this method the user has to manage his cryptographic key, the user has to do computation and communication traffic between user and storage server is high. Encryption scheme limits functionality, only few operations are supported over encrypted data.

3. PROPOSED SYSTEM ADVANTAGE

In proposed system there are distributed storage servers and key servers. The data's are divided into blocks and represented as symbols. Fig. 1 shows the system architecture.

(1) In the threshold proxy re-encryption scheme storage servers independently encode and forward data's and key servers independently perform partial decryption.

(2) The size of storage file is the same as that of the encoded result but a encrypted symbol. confidentiality, flexibility and robustness is maintained in proposed system.

COPYRIGHT TO IJETIE

IJETIE
International Journal of Emerging Technology and Innovative Engineering
Volume I, Issue 3, March 2015
ISSN: 2394 - 6598
www.ijetie.org

4. SYSTEM ARCHITECTURE

Fig.1

5. PROXY RE-ENCRYPTION SCHEME

The proxy server IP is given in the storage server setup. This server can transfer a cipher text with a private key to a new text with another public key.

The data is first encrypted with data encryption key and stored in the cloud storage server, the cloud storage server uses a re-encryption algorithm to transfer the encrypted file into the format that can be decrypted by the recipients private key.

The recipient can download the encrypted data from cloud and use file for decryption. A re-encryption key is generated from the owner's private key and a recipient public key.

The data owner may share different files with different recipient groups. A recipient cannot read data to which it does not belong to. Here the proxy acts as intermediate, the system supports data forwarding downloading and confidentiality is maintained. The data flow diagram is shown below in fig. 2

6. DATA FLOW DIAGRAM

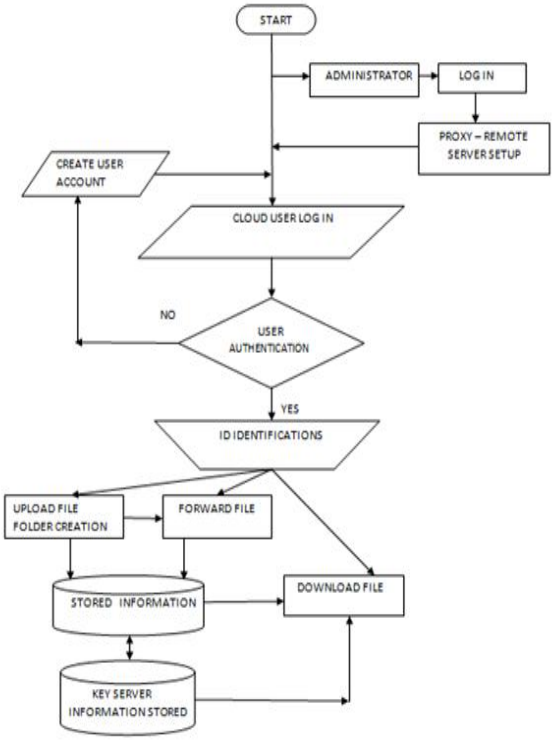

Fig.2

7. SYSTEM MODEL

The general model of this type of secure system is as in Fig.3 below. The theoretical design is turned out into a working system by using application servers Tomcat, Glassfish, Server side scripts are HTML, Java, JSP and Netbeans IDE as the platform with internet.

COPYRIGHT TO IJETIE

IJETIE International Journal of Emerging Technology and Innovative Engineering
Volume I, Issue 3, March 2015
ISSN: 2394 - 6598
www.ijetie.org

7.1. Key Generation Algorithm
a. Key Generation (μ)
1.1. Start
1.2. Obtain parameters (g,h,p).
1.3. G1 can be generated by g and h.
1.4. G2 can be generated by prime number p.
1.5. Set μ=(g,h,ẽ,G1,G2,p,f)
1.6. f:Zp*{0,1} Zp*Is a one-way hash function
1.7. User A selects three parameters a1; a2; a3 Zp*
1.8. PKA=($ggaa1,haa1$),ska=(a1,a2,a3).
1.9. Stop.

7.2. Share Key Generation Algorithm
b) Sharekeygen (SKA,t,m)
2.1. Start
2.2. Select m key servers
2.3. Share secret key SKA to key server ksi
2.4. SKA,i=(fA,1(i),fA,2(i)) where 1≤i ≤m.
2.5. Stop.

7.3. Encryption Algorithm
c) Encryption (PKA,T,m1,m2,…..mk)
3.1. Start
3.2. Divide message in to m1,m2,…..mk
3.3. Calculate cipher text c1,c2,…..ck by using
$C_i = (0, \alpha_i, \beta, \gamma_i) = (0, g^{r_i}, \tau, m_i \tilde{e}(g^{a_1}, \tau^{r_i}))$
3.4. Stop.

7.4. Key Recovery Algorithm
d)Key Recover(SKA,i1 , SKA,i2 , SKA,i3 , SKA,i4……, SKA,in)
4.1. Start
4.2 .User searches for first component a1
4.3. if (a1 found)
4.3.1. No need of key recovery.
4.4. if (a1 not found)
4.4.1. Recovered by using

$$a_1 = \sum_{s \in T} \left(f_{A,1}(s) \prod_{s' \in T/\{s\}} \frac{-s'}{s - s'} \right) mod\ p$$

4.5. Stop

7.5. Re Key Generation Algorithm
e) ReKeyGen (PKA, SKA, ID, PKB)
5.1. Start
5.2. Encrypt the data by using SKA.
5.3. Select a random number e from Zp*.
5.4.Compute rekey for proxy reencryption by using
$RK_{A \to B}^{ID} = \left((h^{b_2})^{a_1 (f(a_3, ID) + e)}, h^{a_1 e} \right)$
5.5. Stop.

7.6. Proxy Re-Encryption Algorithm
f) Proxy ReEncryption ()
6.1. Start
6.2. Calculate RK, C'.
6.3. ReEncrypted symbol can be computed as
$C'' = (1, \alpha, h^{b_2 a_1}(f(a_3, ID)+e), \gamma . \tilde{e}(\alpha, h^{a_1 e}))$

$= (1, g^{r'}, h^{b_2 a_1}(f(a_3, ID)+e), W\tilde{e}(g,h)^{a_1 r' ((f(a_3, ID)+e)})$
6.4. Stop

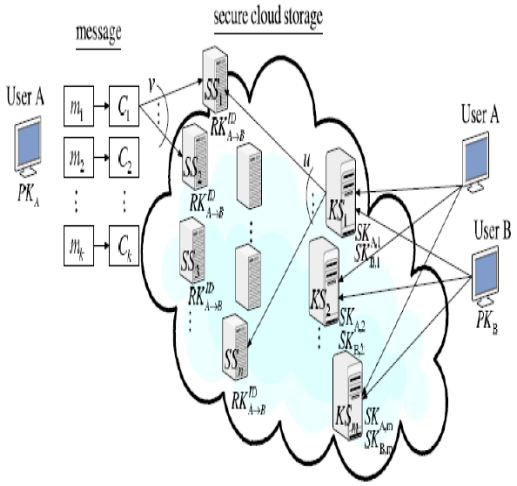

Fig.3 General system model of secure cloud storage

8. LIST OF MODULES
1. Storage Module
2. Encryption Module
3. Forwarding Module
4. Retrieving Module

8.1 Storage Module : Administrator logs in giving his username and password, Server Setup menu opens, he sets the IP address for the Authorised users. By checking Available Storage Server button available IP of servers can be reviewed.

8.2 Encryption Module : The User has to register to have a account for cloud system, username, email, password, date of birth gender and location are needed. In upload process new folder is created for that process user is asked a question, answer must be given by user and also he has to remember the answer for further use. Now server cloud gives encrypted form of the uploaded file.

8.4 Forwarding Module : To forward storage details of uploaded files are seen first by clicking storage details option filename, question answer,

COPYRIGHT TO IJETIE

folder name, forward value (True or False) email forward are displayed. If forward value is True user can not forward, If forward value is False user can forward by selecting filename, email address of the forwarder and enter the code to the forwarder. Now receiver can check his account properly.

8.5 Retrieving Module : For downloading receiver gives username, filename, server connects in that menu user name, filename, question answer and code are required then download option encrypted key appears, using key file can be appropriately viewed.

9. CONCLUSION

To ensure the security of data I proposed an effective distributed storage system where proxy is used to setup server with each key server independently performing decryption and each storage server did encoding and re-encryption only authorized users will be able to access data's in cloud server in this method.

10. ACKNOWLEDGMENTS

This work was done with the support of our College Management and Staff, Especially my Guide **Mrs. K. Shanmugapriya**, Associate Professor, Department of Computer Science and Engineering, Mount Zion College of Engineering and Technology, Pudukkottai. My thanks to all the experts who have contributed towards development of this topic.

11. REFERENCES

[1] Q. Tang, "Type-Based Proxy Re-Encryption and Its Construction," Proc. Ninth Int'l Conf. Cryptology in India: Progress in Cryptology (INDOCRYPT), pp. 130-144, 2008.

[2] G. Ateniese, K. Benson, and S. Hohenberger, "Key-Private Proxy Re-Encryption," Proc. Topics in Cryptology (CT-RSA), pp. 279-294, 2009.

[3] J. Shao and Z. Cao, "CCA-Secure Proxy Re-Encryption without Pairings," Proc. 12th Int'l Conf. Practice and Theory in Public Key Cryptography (PKC), pp. 357-376, 2009.

[4] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession," Proc. Fourth Int'l Conf. Security and Privacy in Comm. Netowrks (SecureComm), pp. 1-10, 2008.

[5] H. Shacham and B. Waters, "Compact Proofs of Retrievability," Proc. 14th Int'l Conf. Theory and Application of Cryptology and Information Security (ASIACRYPT), pp. 90-107, 2008.

[6] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," Proc. IEEE 29th Int'l Conf. Computer Comm. (INFOCOM), pp. 525-533, 2010.

[7] A. Shamir, "How to Share a Secret," ACM Comm., vol. 22, pp. 612- 613, 1979.

[8] W. Dong, F. Douglis, K. Li, H. Patterson, S. Reddy, and P. Shilane, "Tradeoffs in Scalable Data Routing for Deduplication Clusters," Proc. Ninth USENIX Conf. File and Storage Technologies (FAST), p. 2, 2011.

[9] Hsiao-Ying Lin, and Wen-Guey Tzeng, "A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding" ieee transactions on parallel and distributed systems, vol. 23, no. 6, june 2012.

COPYRIGHT TO IJETIE