International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869, Volume-2, Issue-3, March 2014

A Distributed Cloud Storage System Enhancing Security during Data Forwarding

Anju k, M.D. Anto Praveena

***Abstract*— Cloud Storage System can be considered to be a model which depicts the scenario of a networked enterprise storage.Clients can access the data from any where. Data is stored in virtualized pool of storage servers and hosted by third party. A focus on building a storage system which can ensure confidentiality while storing the data is maintained through out.Usage of proxy re-encryption schemes along with a mixing of key-policy based algorithm and lazy re-encryption scheme algorithms .Preprocessing tasks includes encrypting, encoding and forwarding. This system keeps robusting the data storage in a third party cloud.This type of system can be used in handling forensic services.**

***Index Terms*— Key attribute based algorithm, Lazy re-encryption, Proxy re-encryption,**

I. INTRODUCTION

Cloud computing is a mold that treats the resources on the Internet as a unified unit, a cloud. Users use a service without being concerned about how computation is done and storage is managed. This method used to focus on designing a cloud storage system for robustness, privacy, and functionality. A cloud storage system is considered as a large scale distributed storage system that consists of many self-governing storage servers. Data robustness is a major obligation for storage systems. There are many proposals of storing data over storage servers.

One way to present data robustness is to duplicate a message such that each storage server stores a copy of the message.It is robust because the message can be retrieved as long as one storage server survives. Another way is to encrypt a message of k symbols into a codeword of n symbols by erasure coding. To store a message, each of its encoded messages is stored in a various storage server. A storage server failure corresponds to an erasure error of the encode symbol. As long as the number of failure servers is under the acceptance threshold of the erasure code, the message can be recovered from the encode symbols stored in the available storage servers by the decoding process. This provides a trade off between the storage size and the acceptance verge of failure servers.

Manuscript received March 12, 2014.
Anju k, M.C.A, Sathyabama University
M.D. Anto Praveena, Assistant Professor, Faculty of Computing, Sathyabama University.

II. EXISTING SYSTEM

In straightforward integration method Storing data in a TA's cloud system causes serious concern on data confidentiality. To provide confidentiality for messages in storage servers, a user encrypt messages by a cryptographic method before applying an erasure code method and store messages. When use message, we retrieve the Code word symbols from storage servers, decode it. Then decrypt them by using cryptographic keys. General encryption schemes protect data confidentiality. Limits functionality of storage system, few operations are supported over encrypted data. A decentralized architecture for storage systems offers good scalability. Storage servers join or leave without control of a central authority.

Disadvantages Of Existing System

- Users perform more computation and communication traffic between the user and storage servers is high.
- User manages their cryptographic keys otherwise the security has to be broken.
- Data storing and retrieving is hard for storage servers to directly.

III. PRESENT SCENARIO WORK

A. Decentralised erasure code

The process evolves with this phase.The aim of this process is to split up the text or messages into n number of blocks.The condition to be maintained is number of storage servers must be greater than the number of splitted blocks of messages to achieve the result $n=ak^{c}$.It will be creating the code word symbols for the messages and this method is called encoding.now these n spliited messages would be sent for merging process.

B. Merging

In this process the splitted messages are rejoined into m number of blocks and re stored into large storage servers. User A encrypts his message M is decomposed into k number of blocks m1,m2, . . .,mk and which has an identifier ID. User A encrypts each block mi into a cipher text Ci and sends it to v randomly chosen storage servers. Upon receiving cipher texts from a user, each storage server linearly combines them with randomly chosen coefficients into a codeword symbol and stores it. Note that a storage server may receive fewer than k message blocks and we assume that all storage servers

know the value k in advance..Merging is used to combine messages into m number of block, which is encrypted and stored into a large number storage server. Then forwarded to user B. Data which is encrypted by using single key.This is done by using key policy based algorithm.

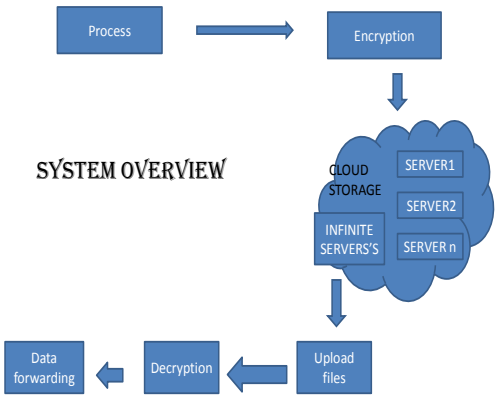

Fig 1.1 Showing the flow of the system.

C. *Encryption*

This is used to encrypt a plain text into a cipher text. Cipher text is produced along with a single key. This key is used to convert the cipher text again into a plain text.The merged data is encrypted with a single key using key policy based algorithm.

D. *Login*

Log in page make user to access an account in a cloud server. When user has an account in the cloud server for accessing data and provides other services. User can sign up the page directly else users needed to create new account using create account option.

E. *Data Forwarding*

In the data forwarding phase, user A forwards his encrypted message with an identifier ID stored in storage servers to user B such that B can decrypt the forwarded message by his secret key. To do so, A uses his secret key SKA and B's public key PKB to compute a re-encryption key RKID A->B and then sends RKID A->B to all storage servers. Every storage server uses the re encryption key to re-encrypt its codeword symbol for later retrieval needs by B. The re-encrypted codeword symbol is the grouping of cipher texts under B's public key. In order to differentiate re-encrypted codeword symbols from intact ones, we call them unique codeword symbols and re-encrypted codeword symbols, correspondingly.

F. *Data retrieval*

Data retrieval is the final module of this project. User download data and using proxy re-encryption method text decoded and partial decrypted. A proxy server can transfer a cipher text under a public key PKA to a new one under another public key PKB by using the re-encryption key RKA->B. In the data retrieval phase, user A retrieves a message from storage servers. The message is either stored by user A or forwarded to user A. User A sends a recovery request to key servers. Upon receiving the recovery request and execute a proper verification process with user A, each key server KSi needs u randomly chosen storage servers to get code symbols and does partial decryption on the received code symbols by using the key share SKA,i. Finally, user A combine the partially decrypted codeword symbols to obtain the original message M.There are two suitcases for the data recovery phase. The first case is that a user A retrieves his own message from cloud. When user A needs to retrieve the message with an identifier ID, he informs all key servers with the individuality token A key server first retrieves original code symbols from u randomly chosen storage servers and then performs partial decryption Share Dec on every retrieved original codeword symbol. The result of partial decryption is called a partially decrypted code symbol. The key server sends the moderately decrypted codeword symbols and the coefficients to user A. After user A collects replies from at smallest amount t key servers and at least k of them are originally from distinct storage servers, he executes Combine on the t partially decrypted codeword symbols to recover the blocks m1,m2, . . .,mk.

G. *Comparison*

The following Fig 1.2 costrasts the difference between the proposed and the existing system in terms of security,time and cost.

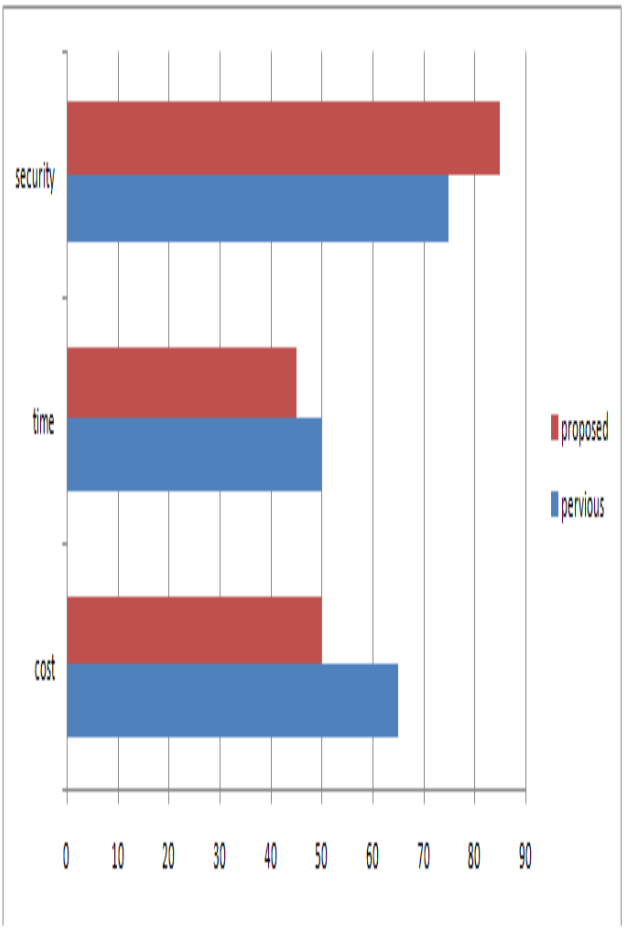

Fig 1.2 shows the comparison done between existing and the proposed work in terms of security,time,cost.

H. Output

Fig 1.3 A snap shot showing the encryption and decryption inside the cloud server using KP-AB method.

CONCLUSION

The proposed system has designed a secure storage system in a distributed environment overcoming the limitations of existing system which made use of TPA(Third party Auditor) and Centralised Server.Future works has to be carried on performance analysis to achieve more efficiency.

REFERENCES

[1] Adya, W.J. Bolosky, M. Castro, G. Cermak, R. Chaiken, J.R.Douceur, J. Howell, J.R. Lorch, M. Theimer, and R. Wattenhofer, "Farsite: Federated, Available, and Reliable Storage for an Incompletely Trusted Environment," Proc. Fifth Symp. Operating System Design and Implementation (OSDI), pp. 1-14, 2002.

[2] Ateniese.G, K. Fu, M. Green, and S. Hohenberger, "ImprovedProxy Re-Encryption Schemes with Applications to SecureDistributed Storage," ACM Trans. Information and System Security,vol. 9, no. 1, pp. 1-30, 2006.

[3] Blaze.M, G. Bleumer, and M. Strauss, "Divertible Protocols and Atomic Proxy Cryptography," Proc. Int'l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT), pp. 127-144, 2008.

[4] Brownbridge.D.R., L.F. Marshall, and B. Randell, "The Newcastle Connection or Unixes of the World Unite!," Software Practice and Experience, vol. 12, no. 12, pp. 1147-1162, 1982.

[5] Dimakis. A.G, V. Prabhakaran, and K. Ramchandran, "Ubiquitous Access to Distributed Data in Large-Scale Sensor Networks through Decentralized Erasure Codes," Proc. Fourth Int'l Symp. Information Processing in Sensor Networks (IPSN), pp. 111- 117, 2011.

[6] Dimakis.A.G., V. Prabhakaran, and K. Ramchandran, "Decentralized Erasure Codes for Distributed Networked Storage," IEEE Trans. Information Theory, vol. 52, no. 6 pp. 2809-2816, June 2006.