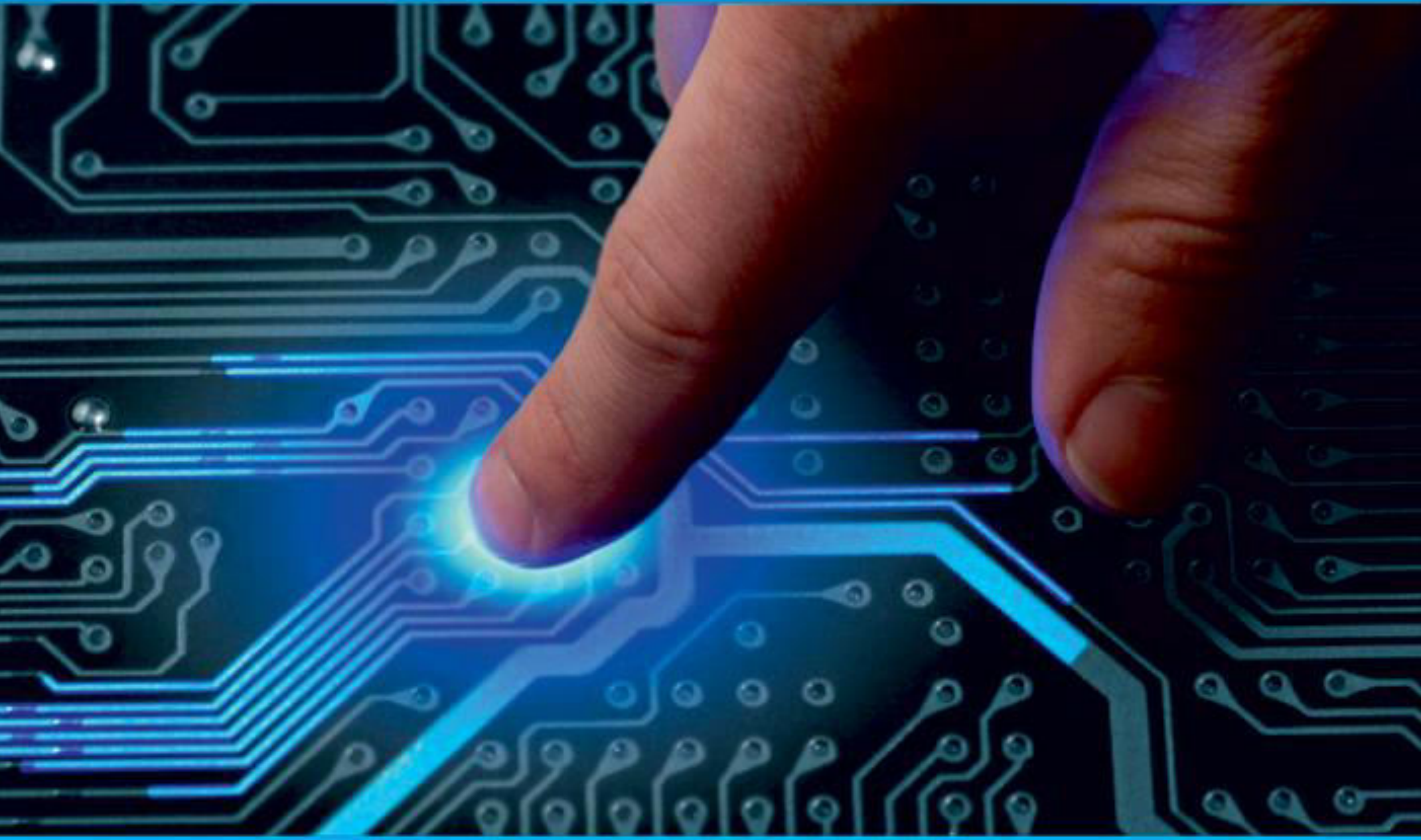
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 4, April 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

9940 572 462 6381 907 438 ijircce@gmail.com www.ijircce.com

International Journal of Innovative Research in Computer and Communication Engineering

e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 4, April 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1204066 |

Enabling Efficient, Secure and Privacy-Preserving Mobile Cloud Storage

Mrs. R.Kavishree[1], Goutham M[2], Manoj Kumar S V[2], Divakar V[2]

Assistant Professor, Department of Computer Science, Muthayammal Engineering College (Autonomous), Kakkaveri, Rasipuram, Tamil Nadu, India[1]

Department of Computer Science, Muthayammal Engineering College (Autonomous), Kakkaveri, Rasipuram, Tamil Nadu, India[2]

ABSTRACT: The system model that consists of distributed storage servers and key servers. Since storing cryptographic keys in a single device is risky, a user distributes his cryptographic key to key servers that shall perform cryptographic functions on behalf of the user. These key servers are highly protected by security mechanism. We propose a new threshold proxy re-encryption scheme and integrate it with a secure decentralized erasure code to form a secure distributed storage system. The tight integration of encoding, encryption, and forwarding makes the Storage system efficiently meets the requirements of data robustness, data confidentiality, and data forwarding. In this paper, we focus on designing a cloud storage system for robustness, confidentiality, and functionality. A cloud storage system is considered as a large scale distributed storage system that consists of many independent storage servers.

KEYWORDS-Threshold proxy re-encryption scheme, Distributed storage system, Cloud storage system, Decentralized erasure code.

I. INTRODUCTION

Cloud computing is an expression used to describe a variety of computing concepts that involve a large number of computers connected through a real time communication network such as the Internet [1]. In science, cloud computing is a synonym for distributed computing over a network, and means the ability to run a program or application on many connected computers at the same time. The phrase also more commonly refers to network based services, which appear to be provided by real server hardware, and are in fact served up by virtual hardware, simulated by software running on one or more real machines.Data robustness is a major requirement for storage systems. There have been many proposals of storing data over storage servers. One way to provide data robustness is to replicate a message such that each storage server stores a copy of the message. It is very robust because the message can be retrieved as long as one storage server survives. Another way is to encode a message of k symbols into a codeword of n symbols by erasure coding. To store a message, each of its codeword symbols is stored in a different storage server. A storage server failure corresponds to an erasure error of the codeword symbol. As long as the number of failure servers is under the tolerance threshold of the erasure code, the message can be recovered from the codeword symbols stored in the available storage servers by the decoding process. This provides a tradeoff between the storage size and the tolerance threshold of failure servers.

A decentralized erasure code is an erasure code that independently computes each codeword symbol for a message [3].Thus, the encoding process for a message can be split into n parallel tasks of generating codeword symbols. A decentralized erasure code is suitable for use in a distributed storage system. After the message symbols are sent to storage servers, each storage server independently computes a codeword symbol for the received message symbols and stores it. This finishes the encoding and storing process. The recovery process is the storage server failure corresponds to an erasure error of the codeword symbol. As long as the number of failure servers is under the tolerance threshold of the erasure code, the message can be recovered from the codeword symbols stored in the available storage servers by the decoding process. This provides a tradeoff between the storage size and the tolerance threshold of failure servers [1].

II. PROBLEM STATEMENT

In Existing System we use a straightforward Integration method. In straightforward integration method

Storing data in a third party‟s cloud system causes serious concern on data confidentiality. In order to provide strong confidentiality for message in storage servers, a user can encrypt message by a cryptographic method before applying an erasure code method to encode and store message. In drawbacks of existing system, the user can perform more computation and communication traffic between the user and storage servers is high. The user has to manage his cryptographic keys otherwise the security has to be broken. The data storing and retrieving, it is hard for storage servers to directly support other functions. The user is unable to share the data confidentiality to the destination.

III. PROPOSED SYSTEM

In our proposed system we address the problem of forwarding data to another user by storage servers directly under the command of the data owner. We consider the system model that consists of distributed storage servers and key servers. We propose a new threshold proxy re-encryption scheme and integrate it with a secure decentralized code to form a secure distributed storage system. The encryption scheme supports encoding operations over encrypted messages and forwarding operations over encrypted and encoded messages. The advantages of proposed system, it will be supports encoding, forwarding, and partial decryption operations in a distributed way. Each storage server independently performs encoding and re-encryption and each key server independently perform partial decryption.

IV. THRESHOLD PROXY RE-ENCRYPTION SCHEME

In the proxy Re-encryption key the messages are first encrypted by the owner and then stored in a storage server. When a user wants to share his messages, he sends a re-encryption key to the storage server. The storage server re-encrypts the encrypted messages for the authorized user. Thus, their system has data confidentiality and supports the data forwarding function. An encryption scheme is multiplicative homomorphic if it supports a group operation on encrypted plaintexts without decryption. The multiplicative homomorphic encryption scheme supports the encoding operation over encrypted messages [2][4].

V. SYSTEM ARCHITECTURE

As shown in Figure. 1, our system model consists of users, n storage servers SS_1; SS_2; . . . ; SS_n, and m key servers KS_1; KS_2; . ; KS_m. Storage servers provide storage services and key servers provide key management services. They work independently. Our distributed storage system consists of four phases: system setup phase, data storage, data forwarding, and data retrieval [1]. These four phases are described as follows.

In the System Setup Phase, the system manager chooses system parameters and publishes them. Each user is assigned a public-secret key pair. User A distributes his secret key A to key servers such that each key server KS_i holds a key share. The key is shared with a threshold t.

In data storage phase, User A encrypts his M message and dispatches it to storage servers. A message M is decomposed into k blocks m_1; m_2; . . .; m_k and has an identifier ID. User A encrypts each block m_i into a cipher text C_i 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 less than k message blocks and we assume that all storage servers know the value k in advance.

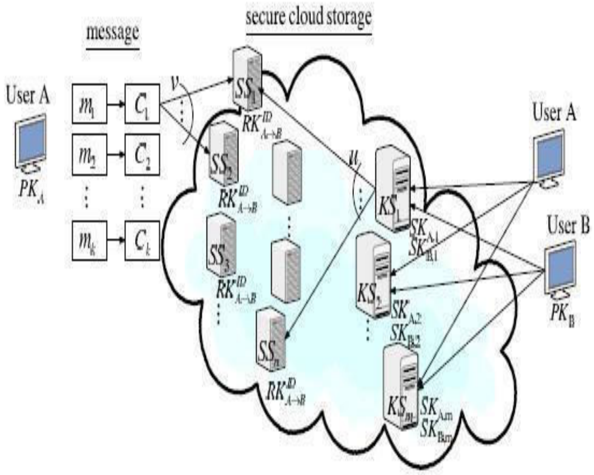


Figure 1: System architecture

In 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 and B‟s public key to compute a re-encryption key and then sends to all storage servers. Each storage server uses the re-encryption key to re-encrypt its codeword symbol for later retrieval requests by B. There-encrypted codeword symbol is the combination of cipher texts under B‟s public key.

In data retrieval phase, User A requests to retrieve a message from storage servers. The message is either stored by him or forwarded to him. User A sends a retrieval request to key servers. Upon receiving the retrieval request and executing a proper authentication process with user A, each key server KS_i requests randomly chosen storage servers to get codeword symbols and does partial decryption on the received codeword symbols by using the key share $SK_{A,i}$. Finally, user A combines the partially decrypted codeword symbols to obtain the original message M.

In our system provides following advantages,

1. It provides robustness of data, confidentiality and data forwarding using combination of encoding, encryption, and forwarding which makes an efficient system.

2. Encoding and re-encryption are performed independently by storage servers and partial decryption is performed by key servers.

3. It is flexible for the number of storage servers and its robustness.

4. While user login an additional image login will be provided to user to provide an extra security.

5. Instead of blocking a particular user an IP blocking method can be used to provide login to original user.

VI. CONCLUSION

The threshold proxy re-encryption scheme supports encoding, forwarding, and partial decryption operations in a distributed way. To decrypt a message of k blocks that are encrypted and encoded to n codeword symbols, each key server only has to partially decrypt two codeword symbols in our system. By using the threshold proxy re-encryption scheme, we present a secure cloud storage system that provides secure data storage and secure data forwarding functionality in a decentralized structure. Moreover, each storage server independently performs encoding and re-encryption and each key server independently perform partial decryption. It will reduce the storage and computation cost.

Our Key server performs the main role in our distributed storage system. This Key server performs the important role key management. But our proposed system doesn"t provide any security over this Key Server. The attacker or intruder can attack the key server to get the secret key because there is no security provided to the Secret Key. So as a future work we focus on key server for giving more secure to our storage system. To overcome this problem we are going apply the encoding over the secret key before it store in the Key server. The Key server can decode this encoded Secret Key.

REFERENCES

[1] Hsiao-Ying Lin,Wen-Guey Tzeng, "A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding",June 2012.

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

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

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

[5] H.-Y. Lin and W.-G. Tzeng (Nov 2010), "A Secure Decentralized Erasure Code for Distributed Network Storage," IEEE Trans. Parallel and Distributed Systems, vol. 21, no. 11, pp. 1586-1594.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details