E-ISSN: XXXX-XXXX

International Journal of Computer Science and Management [IJCSM]

Volume:1 Issue:1 pp.26-30

PROXY RE-ENCRYPTION AND SECURE CODE-BASED CLOUD STORAGE

Dr.Geetha B.G
Professor
Computer Science and Engineering
K.S.Rangasamy College of Technology
Tiruchengode, India.
geethaksrct@gmail.com

Dr.Senthilkumar R
Associate Professor
Computer Science and Engineering
Shree Venkateshwara Hi-Tech Engineering College
Gobi, Erode, India
yoursrsk@gmail.com

Abstract: **Distributed data storage gives users access to virtualized, highly scalable resources. However, there aren't many security problems with it either. Long-term data storage via the Internet is possible using a distributed storage system, which consists of a number of storage servers. Concerns about data confidentiality are greatly raised when data is stored on an external storage device. It is possible for another user on the same system to intentionally or unintentionally access someone else's data. Re-encryption and a decentralised structure are thus used in this system to prevent such and provide data resiliency. The two systems that the storage system claimed were the storage server and the key server. To provide maximum confidentiality for messages in storage servers, data to be stored there is first encrypted using a cryptographic technique, and then messages are encoded and saved using an erasure code technique. In order for systems to have a distributed structure, all operations must be performed separately by servers. In light of this, a secure distributed storage system is created by combining a new re-encryption algorithm with a safe decentralised code.**

Keywords-Decentralized erasure code, proxy re-encryption, threshold cryptography, secure storage system.

1 INTRODUCTION

Over the past few decades, one of the primary issues with information technology has been acknowledged: data storage. The shift from server-attached storage to distributed storage has been driven by the advantages of network-based apps. A distributed data store is a type of computer network where data is duplicated and stored across multiple nodes. It is typically used to describe one of two things: a computer network where users store data on several peer network nodes, or distributed storage where users store data on several nodes. Users can typically reciprocate in peer network data stores by permitting other users to utilise their computer as a storage node. Depending on how the network is set up, other users might not be able to access some information. Usually, distributed data stores employ an error detection and correction method. When a file is partially damaged or unavailable, certain distributed data repositories recover the original file using forward error correction techniques.

Regarding control, there is no doubt about it in client-server architecture. As the focal point, the server is in charge of client service requests, backups, consistency, replication, and authentication. Different degrees of centralization can be seen in client-server architecture; two types of centralization are recognised: globally centralised and locally centralised. A highly centralised architecture with restricted scalability and failure susceptibility is the product of a globally centralised architecture, which includes a single central entity—the server. Unfortunately, scalability is limited and a Single Point of Failure results from this reliance on a globally central index server. A locally centralised design divides duties among several servers to reduce the issues that come with having a single central server. This makes the systems more scalable, more resilient to outages, and more efficient overall.

2 RELATED WORKS

2.1 CRYPTOGRAPHY OVERVIEW

The study and application of secure communication methods in the presence of third parties, or adversaries, is known as cryptography. Broadly speaking, it involves developing and assessing methods that thwart the influence of adversaries and are connected to several information security facets like data integrity, non-repudiation, authentication, and confidentiality. Electrical engineering, computer science, and mathematics are all impacted by modern cryptography [2]. Cryptography helps create secure channels for communication over otherwise unsecure ones, helps prevent data from being read, and helps determine whether data has been altered. For instance, data can be delivered in an encrypted state after being encrypted using a cryptographic algorithm, and the intended recipient can then decrypt the data [2]. It will be challenging to decode encrypted data if it is intercepted by a third party.

2.2 PROXY RE-ENCRYPTION

A proxy server can move a cypher text under private key A to a new one under public key B in a proxy re-encryption scheme. During transformation, the plaintext is unknown to the server. The information is saved on the cloud storage server after being initially encrypted using a symmetric data encryption key.

The encrypted DEK is transferred by the cloud storage server via a re-encryption method into a format that the recipient's private key can decrypt. After then, the recipient can use the DEK to decode the encrypted data after downloading it from the cloud. The public key of the recipient and the private key of the data owner are used to create a re-encryption key [6].A data owner may distribute various files to various recipient categories. As such, a recipient is unable to access data for a group to which it is not belong. Conversely, the cloud serves as a middleman stand-in. Since it cannot receive DEKs, it is unable to read the data. As a result, the system supports data transmission and maintains data secrecy.

2.2.1 Key Private Proxy Re-encryption

An important personal PRE The ciphertext and the set of public keys prevent the proxy and a group of users in collusion from determining the recipient of a communication. Only when the underlying encryption system is key-private can key private PRE be achieved. The secrecy of the key used to perform the encryption is ensured via key privacy encryption [10]. The KP-PRE scheme introduces the concept of keyprivacy for proxy re-encryption methods, in which the identities of the participating parties are not even discernible to the proxy doing the translations.Apart from preventing file contents from being viewed by the proxy, it is also beneficial to suppress as much meta-data as feasible. We might want the proxy file erver, for instance, to re-encrypt private files for certain receivers without disclosing the recipient's identity to the proxy.

2.2.2 Type Based Proxy Re-encryption

Fine grained access control and data secrecy are ensured by this encryption technique. The delegator may apply fine-grained policies with a single key pair and no additional proxy trust thanks to type-based proxy re-encryption. The delegator divides his cypher texts into several subsets according to this technique [6]. Next, each subset's decryption right is assigned to a certain delegate. The

The message type, which is used to identify the message subset, and the delegator's public key are used to generate cypher texts for the delegator. The attributes of the type-based PRE are as follows.

1. The key management problem can be simplified because the delegator only needs one pair of keys.
2. Depending on the sensitivity of the delegation, the delegator may select a specific proxy for a certain delegate [9]. One subset of communications will only be impacted by a single proxy key compromise.

2.3 DECENTRALIZED ERASURE CODE

A random linear code with a sparse generating matrix is called a decentralised erasure code. The following is the generating matrix G that an encoder creates: The encoder first marks one entry as 1 at random for each row. It then repeats this procedure k/k times with a replacement. Secondly, a value is randomly assigned by the encoder to either IF or each designated entry. The encoding process is now complete [8]. If the k-selected columns form an invertible $k \times k$ sub matrix, then the decoding process is considered successful.

As a result, the likelihood that the selected sub matrix will be invertible equals the likelihood that the decoding will succeed. A copy of Mi is sent to each of the v servers that the owner has randomly chosen with replacement. Every server carries out a linear combination of all received cypher texts after selecting a coefficient at random for each received cypher text [10]. A codeword element is the outcome of the linear combination of the coefficients selected by the server, which make up a column of the matrix. The computation can be completed individually by each server. The code is now decentralised as a result.

3 SYSTEM MODEL

3.1 Message Block Encryption

This system's user is deemed to have already undergone authentication and is recognised as a legitimate user. The file is encrypted by the system using one of the most secure algorithms—the Blowfish algorithm—after the end user first chooses which file to store across the storage server. After then, the encrypted file is made accessible for token splitting.

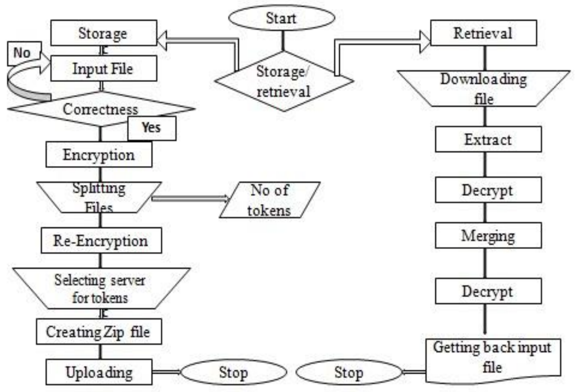

Fig-1 Overall Architecture

3.2 Re-Ciphering Function

The original file is encrypted, and then the Token-Generation process divides it into tokens. The user must provide the quantity of tokens to be generated, which will then be divided appropriately. For improved security, each split token is re-encrypted. The re-ciphering procedure is accomplished via the Blowfish algorithm. The tokens are now prepared for use in the storage sector.

3.3 Secure Data Storage

The trustworthy server is prepared to receive the re-encrypted tokens. The client computer and storage servers are linked via a secure communication protocol. For this, TCP/IP protocol is employed. The data owner determines which tokens should be sent to which storage server once a secure communication connection has been created, and data is passed appropriately.

3.4 Decentralization

This is how the original file's tokens are retrieved from the storage server. The dispersed servers search for the tokens, which are then sent to the client where they are decrypted and combined into a single file that contains the encrypted content of the original file.

3.5 Data Forwarding

The user receives the merged file when it has been encrypted. The end user's original file is the result of this last phase. For the transfer procedure to take place, the target user needs to be able to retrieve data.

4 ANALYSIS

In this section, we examine the security, accuracy, and computational and storage complexity of our cloud storage system.

Truthfulness. For correctness, there are two instances. Both user B and owner A successfully retrieve messages that have been forwarded to them. (1) shows that the encryption and decryption for A were accurate. (2) shows that the re-encryption and decryption for B were correct. An overwhelming likelihood exists for a user to get data as long as there are at least k storage servers available. Our storage system can therefore withstand n _ k server failures.
the likelihood that a retrieval will be successful. Whether or not the message was sent to or owned by the user, a successful retrieval occurs when the user is able to obtain all k blocks of the message. The random selection of storage servers during the data storage phase, the random coefficients selected by storage servers, and the random selection of important servers during the data retrieval phase are the sources of the unpredictability. The likelihood of a successful retrieval is contingent upon (n; k; u; v) and the entirety of randomness. The analysis methodology used here is comparable to that of [10] and [6]. Nevertheless, we take into account a different system model than the one in [8] and a more adaptable n ¼ akc parameter setting than those found in [10] and [6]. Our system model has critical servers, which sets it apart from the system model in [9]. To obtain the data, a single user queries k different storage servers in [7]. Conversely, every important server inside our system asks the u storage servers on its own. While using dispersed key servers improves key safety, it also complicates analysis. In [9], the ratio n=k is regarded as a fixed constant. The configuration is expanded to n ¼ ak3=2 in [6]. Our expanded parameter configuration for n ¼ akc, where c _ 1:5, permits a far larger number of storage servers than the message's block count. It provides greater adjustment freedom between robustness and the quantity of storage servers.

safety. Our cloud storage solution ensures data secrecy even in the event that an attacker compromises all storage servers, nontarget users, and up to {t _ 1\ key servers.

5. CONCLUSION

Organisations gathering data want a secure location to store it as data security concerns grow. Those that access the distributed storage servers must be able to access private data in a secure manner. A distributed data storage system that offers safe data storage is put into place by utilising the re-encryption process along with a decentralised structure. The end user's file contents are extremely difficult for hackers to decrypt thanks to the encryption techniques used. Organisations that need to keep and retrieve secret material safely, without worrying about security breaches, would find this technology to be very helpful.
Several different file types are stored on a storage system. With this system, users are permitted to store text files in the.txt,.doc, and.docx formats as well as images to be securely kept on storage servers. This method can be expanded to include video and audio. Additionally, this technology can be utilised during the cloud setup process if a private cloud is to be formed and security is a top priority. To raise the system's quality, the server's load balancing components must be attended to.

6.REFERENCES:

[1] Hu, B.; Chen, Y.; Yu, H.; Meng, L.; Duan, Z. Blockchain Enabled Data Sharing Scheme for Consumer IoT Applications. IEEE Consum. Electron. Mag.2021.

[2] Jiang, Y.; Shen, X.; Zheng, S. An Effective Data Sharing Scheme Based on Blockchain in Vehicular Social Networks. Electronics 2021, 10, 114. [.

[3] Liang, K.; Susilo, W. Searchable Attribute-Based Mechanism With Efficient Data Sharing for Secure Cloud Storage. IEEE Trans. Inf. Forensics Secur.
2017, 10, 1981–1992.

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

[5]Liang, K.; Susilo, W. Searchable Attribute-Based Mechanism With Efficient Data Sharing for Secure Cloud Storage. IEEE Trans. Inf. Forensics Secur.
2017, 10, 1981–1992.

[6] Zhou, Y.; Deng, H.; Wu, Q.; Qin, B.; Liu, J.; Ding, Y. Identity-based proxy re-encryption version 2: Making mobile access easy in cloud. Future Gener. Comput. Syst. 2016, 62, 128–139

[7] Majumdar, M.A.; Monim, M.; Shahriyer, M.M. Blockchain based Land Registry with Delegated Proof of Stake (DPoS) Consensus in Bangladesh. InProceedings of the 2020 IEEE Region 10 Symposium (TENSYMP), Dhaka, Bangladesh, 5–7 June 2020.

[8]G. Zheng, X.; Feng, W. Research on Practical Byzantine Fault Tolerant Consensus Algorithm Based on Blockchain. J. Phys. Conf. Ser. 2021, 1802, 032022

[9] Manzoor, A.; Liyanage, M.; Braeken, A.; Kanhere, S.; Ylianttila, M. Blockchain based Proxy Re-Encryption Scheme for Secure IoT Data Sharing. In Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Seoul, Korea, 14–17 May 2019.

[10] Mollah, M.B.; Azad, M.; Vasilakos, A. Secure Data Sharing and Searching at the Edge of Cloud- Assisted Internet of Things. IEEE Cloud Comput. 2017, 4, 34–42