Blockchain assisted Cloud Storage with Keyword Search

Sachin Karthick R[1]
Computer Science Engineering
S.A.Engineering College

Karthik B [2]
Computer Science Engineering
S.A.Engineering College

Lokesh Kumar B [3]
Computer Science Engineering
S.A.Engineering College

Balakrishnan C[4]
Computer Science Engineering
S.A.Engineering College

Abstract: ***Cloud storage allows users to store data in the storage servers and retrieve the stored data efficiently. Few of the data that is stored are very sensitive and should be prevented from attacks and leakage. In general, when users traditionally encrypt the data, efficient searching of data becomes difficult or impossible. Public-key encryption with keyword search (PEKS) resolves this issue. But this scheme is vulnerable to keyword guessing attacks (KGA). In this paper we introduce a secure PEKS scheme called SEPSE to prevent KGA, where users encrypt keywords with the help of dedicated key servers. These key servers enable automatic renewal of keys periodically replacing the existing key to prevent key compromise. Furtherly SEPSE can efficiently resist online KGA where each keyword request made by the user is integrated with a transaction on a blockchain network which lets the servers learn the number of keyword requests made by the user.***

Keywords: ***Cloud storage, public-key encryption with keyword search, keyword guessing attacks, key renewal, blockchain.***

I. Introduction

Evolution of online/offline cloud storage services has enabled users to store data to the server space they are provided with and lets them flexibly access and share the data with others. Applications such as the e-mail which is cloud based lets multiple users to transmit data that hold a tiny amount of keywords to some users who are in need of the data. Senders are given the ability to get off/save to the storage server the data as well as the keywords. The end users who are in need of the data can make a request to the server for the data where the data is stored by using the concept of keyword search. This can help users to cut down storage costs which is heavy and provides the receiver end the power to acquire and work on the data that is provided or sent on any desired devices at some point of time. (e.g.Smartphones)

While enjoying benefits from cloud services that is probably used for storage, users are always put into a greater security threat whilst outsourcing the data which is considered to be data confidentiality. Users perspective is that they consider their data very sensitive and private so there shouldn't be any sorts of leakage in them and they prefer encrypting the data that they outsource. This can be attained by traditional or conventional encryption algorithms, but makes efficient searching of data through cipher texted keywords is impossible.

Encrypting with the help of keyword search by using the public key is one of the algorithmic primitives of cryptography that provides a fix to this problem. In the concept of PEKS, data and the keywords both are encrypted by using the receivers public key and the cloud server is used to store the ciphers. (i.e Storage Server). At the receiving end, the receiver can generate a trapdoor with the help of the particular users secret key and the cloud servers security checks whether the cipher-text of the keyword goes with the trapdoor and if its authenticated it retrieves the data. Anyway, PEKS is subjected to some security limitations and is vulnerable to offline guessing attacks of keywords (KGA).Every keyword in a trapdoor uses public key of the receiver and finds thetext that matches the trapdoor that is ciphered and targeted. This lets the person who wants to attack the system by letting the attacker to recover the keyword hidden that violates users privacy. By studying or observing the keywords these attacks are done. The keywords that are used for these attacks are usually obtained from the spaces that are relate ably smaller in number. Familiar keywords are always used for the efficient searching of files.

To resist the system from offline guessing attacks of keywords a keyword search scheme that uses public key encryption is used. A key server is used to create the keyword and its encryption where the cloud storage server only concentrates only one the transferring of the users data. Keywords that are generated by the server uses a particular protocol that happens between the user and the server, is used to generate the key and the receiver can request for the data and the keywords from the server without putting out any kind of secretive information leaked or provided to the key server (i.e the key server does not learn anything about the keyword of the data) Attackers who will not be able to compromise the server that contains the key cannot actually obtain and create the cipher texts of the keywords while the system is offline or while performing an offline KGA.

With server aided Encryption scheme having a lot of benefits, this implementation builds trust by tricking us that it provides a better reliable server that holds the keys. Compromising of the server that holds the keys is practical when the server containing the key becomes a single point of failure. Once it is compromised by the attacker the security of the keywords is not ensured. To ensure the security of the keys, the keywords that are created can be distributed to multiple server that holds the key rather than using a single key-server and sharing a

value between them which is considered to be secretive that updates the server can be implemented.

An encryption scheme that has a greater security and efficiency called SEPSE is used to prevent both offline and online keyword guessing attacks for a cloud storage that is highly secure is proposed in the paper. In this proposed system to overcome the single point failure of the server multiple servers to hold the key is involved. A confidential information used to secure and shield these keywords from getting attacked arepassed along these servers that holds the keys. SEPSE involves in the efficient renewal of the key servers, that is each server that holds the key renews the confidential key at a particular interval of time. Moreover, to resist online guessing attacks of keywords blockchain assisted layover is used.

II. Existing System

Encryption techniques plays the vital part in the systems that outsource the data, where users can put up their queries in the form of search-keywords to a server that has the data stored and the data server provide the user with the data without learning any information anything about the search result. Since the server will have to scan the words in each file one word after the other this concept becomes inefficient. Encryption algorithms with different behaviours based on the cryptosystems that is symmetric were subsequently introduced and they are put into use when user upload their data to the cloud and search for the data when in need of it

Keyword search with public key encryption is not much of an efficient kind though has some expressive features. Considering the cloud based storage systems, the texts those of that are ciphered are secured by using the users public key who searches for the data from the data store. The keywords are selected from the series of operation that is performed by the users and they are stores as known ones. This increases the chance and advantages of Off-line guessing attacks on keywords.

Withstanding offline KGA with the help of a certified tester. Only one has the rights to run tests on comparing if the ciphered text matches with the trap-door. By this method keywords are protected from KGA performed outside. Keyword Search based on the Public key encryption is constructed by using the cryptographic primitives that is emerging to solve the problem with offline KGA. In this scheme ciphers are given rise by a signcryption scheme, and a server that is used for storage cannot generate a legit ciphered text.

Another way to resist offline KGA, PEKS with fuzzy keyword is used. That main reason to opt in with the trapdoor that uses a keyword which is fuzzy as they are fast enough to perform more than one activity. Looking up and creating ciphered texts is done by the users and a trusted node is assigned to help the user with it. To prevent the storage system from getting exploited and failing on single point problems. Different servers to generate keys at the same time are used and even renew the key by itself after a particular period of time.

III .Blockchain

Blockchain is a protocol on the internet executed by multiple users to achieve security on the system generate data unit. Each of the data unit is called *block*. These blocks together form an hash chain and is ensured that the data is secure by using an hash function that is cryptographic. Consensus algorithm plays one of the major function inthe constructions of blockchain. Examples, PoW, PoS, etc. Firstly a block header that mainly contains the following:

- **Pre Block Hash**: In the pre block hash the value of the hash is saved at the bottom and is shared with the other blocks forming a chain.
- **Time**: It is a timestamp and it indicated when the block is appended.
- **Nonce**: It is considered as the answer to the PoWpuzzle.
- **MerkleRoot**: The root value that is obtained from all the transaction in the block that is considered to be the current one via Merkle roothash.

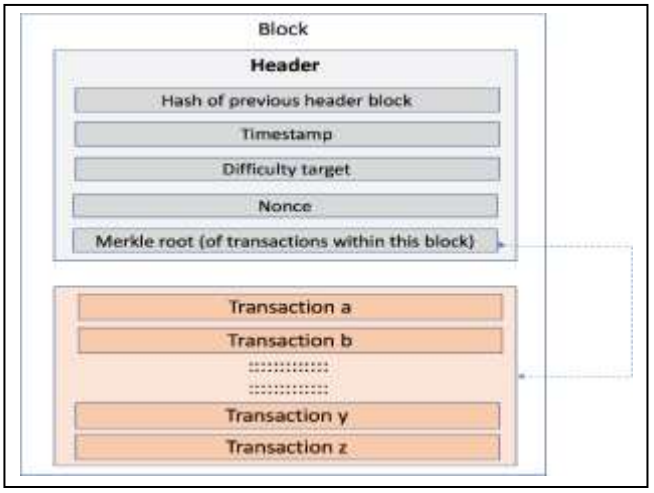

Figure 1: Structure of a Block.

Secondly, a data block for the transactions which contains all the values to be stored in the current block. Block chain can be spilt up into two categories, one is the private blockchain in which the participants must be approved by an authority or a set of authorities. Another is public blockchain, where any of the user can actively participate or leave the system without any approval of the authority. A block can be added to blockchain, only if it is accepted by n number of participants. Blockchain is one of the successful applications of public blockchain, where it serves as publicly verifiable ad distributed ledger to secure record transactions among participants.

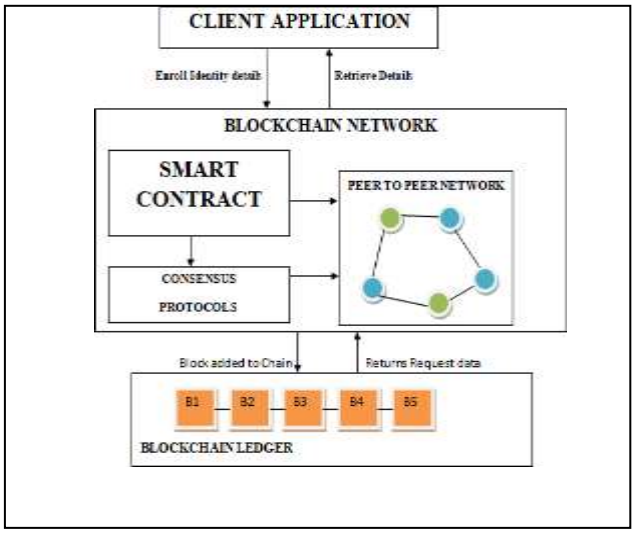

Figure 2: Blockchain Model Architecture

Each and every nodes of the peer to peer network will have a copy of the updated blockchain ledger. Consensus protocols ensure that only trusted blocks are added to blockchain using the algorithms such as Proof of Work etc. Smart Contracts are a set of codes which are deployed in a blockchain and executed during addition of blocks.

IV. Proposed System

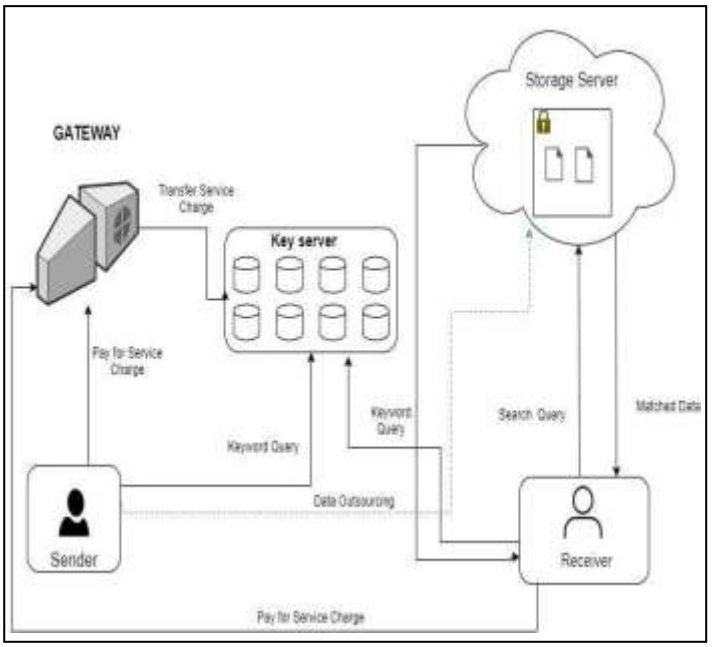

Figure 3: Proposed Architecture.

- **Senders**: Senders outsource the data to the target system by securing the data which contains a small number of keywords that is encrypted with the use of receiver's public key.
- **Receiver**: The individual who receives the data that is encrypted from the server where the data is stored and involves in the decryption locally. The receiver can later at some point of the can search for the data with the help of keywords.
- **Key Servers**: They are helpful by assisting sender and the receiver generate the keyword tobe encrypted, that will prevent the guessing attacks on keywords.
- **Storage Server**: Receives and stores the encrypted data and the keywords that is encrypted. The receiver is provided with an efficient and a secure way to search the texts that are ciphered by using keywords, and involving in sending the target ciphers to the receiving end.
- **Gateway**: Helps by assisting users to forward the request of generation of the keywords to the key that holds the key, and assist them in getting service charges from its users.

*Setup, PEKS, Trap-door, Test and Key Renew*are the major algorithm present for this implementation to be successful.

- **Setup**: The basic parameters used in the algorithm is implemented and created with the help of this algorithm.
- **PEKS**: The user generates a keyword that is derived from the server with the help of key servers and uses the receivers public key to encrypt the server-derived keywords.
- **Trapdoor**: Enables the receiver to generate the keyword that is server derived with the help of all key servers that is available, and helps creating a corresponding token for the search.
- **Test**: General test conditions are performed which allows that server that has the data stored perform operations to check whether the derived keyword matches the given search token.
- **Key Renew**: Provides the server that holds the key to refresh its confidential bits without changing the information shared among all the servers that holds the key.

The Proposed system involves in the usage of the servers that holds multiple keys to allow users in the encryption of the keywords to prevent guessing attacks on keywords, the potential to generate keywords is shared among the key servers, so that a server that is considered to be a single key, whose failure will not affect the entire security of the system.

A confidential information is shared between these servers that holds the key in a distributed and a threshold way as sharing a secret does not always need to involve a trusted node in the system. This makes the SEPSE free from single point failure.

To prevent guessing attacks on keywords that is online, a mechanism that limits the rate of the transactions and other are employed in the network. Each user will only be able to make limited request to the server.

Furthermore, each query that is given by the user to the user to the system is converted and saved in blockchain network as a transaction.

V. Implementation

This system is similar to that of the Online Cloud Storage which runs on a cloud platform allowing the users to store, edit and modify the data with the help of any device ubiquitously. This application is developed by using Python Programming Language and is deployed on a local cloud system to verify the systems stability against the offline KGA attack and an online storage system to verify the system against KGA attacks.

Personal Blockchain is created by using Ganache for Ethereum Development. and run the tests. Smart contracts and other testing environments is created by using the Truffle Framework with Metamask as the personal blockchain funds wallet.

VI. Applications and Challenges

Blockchain based Online Cloud Storage System can be implemented for enhanced security of users data that is considered to be highly senstitive.

The proposed Algorithm can further improved in terms of security. The energy consumption to mine a block of data in blockchain technology uses a lot of power which is considered to be a main challenge to be overcomed.

Furthermore performance of SEPSE can be studied and improved for much faster retrieval of data.

VII. Conclusion

SEPSE, which is considered to be an efficient PEKS scheme to provide security to the servers against off-line guessing attacks on keywords is presented, where encryption of keywords is done with the help of using multiple key servers which makes system free from the single point of failure problem. Key renewal on each key server is supported by SEPSE. By utilizing the blockchain assisted mechanism that limits the rate SEPSE can thwart Online guessing attacks on keywords where the keyword searches or requested made by the user is recorded as transactions and is limited for each user.

VIII. REFERENCES

[1] X. Liu, R. Deng, K. R. Choo, and Y. Yang, “Privacypreserving outsourced support vector machine design for secure drug discovery,” IEEE Trans. Cloud Computing, accepted 2018, to appear, doi. 10.1109/TCC.2018.2799219.

[2] A. Yang, J. Xu, J. Weng, J. Zhou, and D. S. Wong, “Lightweight and privacy-preserving delegatable proofs of storage with data dynamics in cloud storage,” IEEE Trans. Cloud Computing, accepted 2018, to appear, doi: 10.1109/TCC.2018.2851256.

[3] Y. Zhang, C. Xu, X. Lin, and X. Shen, “Blockchain-based public integrity verification for cloud storage against

procrastinating auditors,” IEEE Transactions on Cloud Computing, accepted 2019, to appear, doi: 10.1109/TCC.2019.2908400.

[4] N. Borenstein and J. Blake, “Cloud computing

standards: Where’s the beef?” IEEE Internet Computing, vol. 15, no. 3, pp. 74–78, 2011.

[5] Y. Zhang, X. Lin, and C. Xu, “Blockchain-based secure data provenance for cloud storage,” in Proc. ICICS, 2018, pp. 3–19.