ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

Secured Data Sharing Principles with Strong Key Analysis over Cloud Crypto System

B.MURALI,
Research Scholar
Department of Computer Science,
J.J. College of Arts and Science (Autonomous),
Pudukkottai, Tamil Nadu, India

Dr.S.ADAEKALAVAN,
Assistant Professor,
Department of Computer Science,
J.J. College of Arts and Science (Autonomous),
Pudukkottai, Tamil Nadu, India.

ABSTRACT: The main objective of this system is to efficiently share the resources from server to destiny ends using advanced encryption scheme called Key Aggregation encryption and decryption, proves this scheme is better than all the existing techniques. Data Sharing is the main task over cloud computing, but the query always present in every one's mind is how to securely, efficiently, and flexibly the data is shared with others. Strong Key Cryptosystem is introduced to produce a constant size cipher texts such that efficient allocation of decryption rights for any set of cipher texts are possible. Advanced Key sharing system based on hint text methodology is formed to share the data safely. Once the data sharing is completed then the key exposure differs from its actual form. So the user cannot guess the key based cryptosystem and this process provides efficient solution than the existing ones.

KEYWORDS: Cloud Computing, Cloud Storage Auditing, Data Integrity, Key Exposure.

I. INTRODUCTION

Now-a-days, cloud storage is becoming one of the most attractive choices for individuals and enterprises to store their large scale of data. It can avoid committing large capital of users for purchasing and managing hardware and software. Although the benefits of cloud storage are tremendous, security concerns become significant challenges for cloud storage.

One major concern on cloud storage security is about the integrity of the data stored in cloud. Because clients lose the control of their data stored in cloud and data loss might happen in cloud storage, it is natural for clients to doubt whether their data are correctly stored in cloud or not. Cloud storage auditing, as one effective security technique, is proposed to ensure the integrity of the data stored in cloud. Many cloud storage auditing schemes have been proposed up to now.

These schemes consider several different aspects of cloud storage auditing such as the data dynamic update, the privacy protection of user's data, the data sharing among multiple clients and the multi-copies of cloud data. Key-exposure resilience, as another important aspect, has been proposed recently. Indeed, the secret key might be exposed due to the weak security sense and/or the low security settings of the client. Once a malicious cloud gets the client's secret key for cloud storage auditing, it can hide the data loss incidents by forging the authenticators of fake data. As the same reason, it even can discard the client's rarely accessed data for saving the storage space without being found out by cloud storage auditing.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

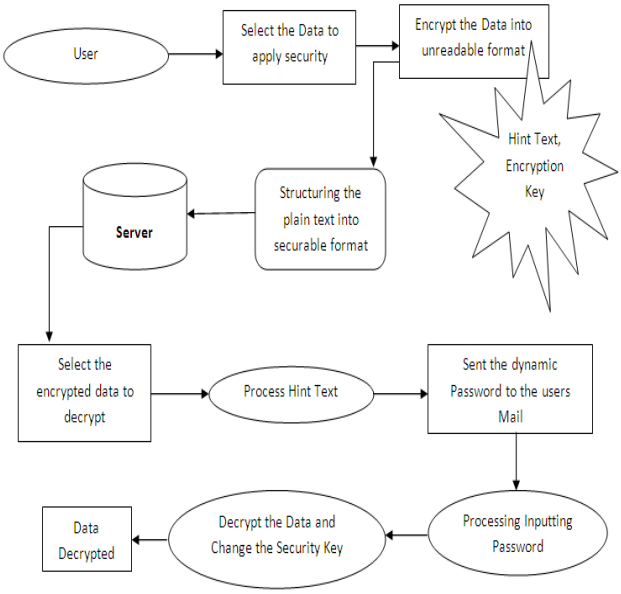

Fig.1 Proposed System Archietctural Design

In a key update technique based on binary tree structure is used to protect the security of authenticators generated in time periods earlier than the key exposure. As a result, the cloud storage auditing scheme, to some extent, can deal with the key exposure problem. However, in some cases, the key exposure problem is not fully solved in the scheme due to the following reason. When the key exposure happens, it often cannot be found out at once. The key exposure might be difficult to be found out because the attacker might stop intrusion at once when it gets the client's secret key. So it is common that there is a long time span crossing multiple time periods between key exposure and its detection. The key exposure might be detected only when the user finds the valid authenticators are not generated by himself. At that time, the user has to revoke the old pair of public key and secret key, and regenerate a new pair. We give an example to show this problem in Fig. 1. Suppose the hacker has appropriated the client's secret key during session te but the key exposure has not been detected at that time. The attacker can update the exposed secret key, as same as the client does, to generate the secret keys for time periods te; :::; td until key exposure is found out during time period td.

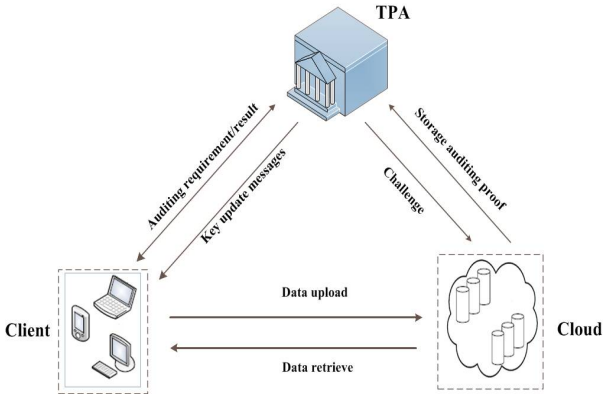

Fig.2 The proposed system model

This means that the malicious cloud trading with this hacker can modify even delete the client's data uploaded during time periods te;....;td without worrying about being found out. It can generate the authenticators for fake data to

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

pass the cloud storage auditing using the updated secret keys. It is a natural problem of how to protect the security of the cloud storage auditing during the time periods not only earlier than but also later than the key exposure.

II. EXISTING APPROACHES – A SUMMARY

Many cloud storage auditing schemes have been proposed up to now. These schemes consider several different aspects of cloud storage auditing such as the data dynamic update the privacy protection of user's data the data sharing among multiple clients and the multicopies of cloud data. Key-exposure resilience, as another important aspect, has been proposed recently . Indeed, the secret key might be exposed due to the weak security sense and/or the low security settings of the client. Once a malicious cloud gets the client's secret key for cloud storage auditing, it can hide the data loss incidents by forging the authenticators of fake data. The existing system has several disadvantages, some of them are listed below: (a) Each file in cloud requires a unique authentication key to decrypt, (b) Hard to support multiple file requests at same time, (c) Confusion arises while decryption and also the key will expired means again we need to request for the same file and (d) Poor in performance and time consumption is more because of unidirectional file requesting scheme.

III. PROPOSED SYSTEM

We investigate how to preserve the security of cloud storage auditing scheme in any time period other than the key-exposure time period when the key exposure happens. We propose a paradigm named strong key-exposure resilient auditing as a practical solution for this problem in this system.

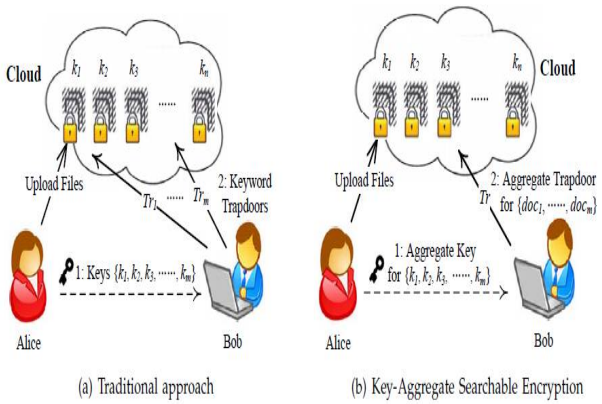

Fig.3 Secured Data Sharing over Cloud

We design a concrete strong key-exposure resilient auditing scheme for secure cloud storage. A novel and efficient key update technique is used in the designed scheme. In our detailed construction, the Third Party Auditor (TPA) generates an update message from his secret key in each time period, and then sends it to the client.

We formalize the definition and the security model of this new paradigm. In the security model, we consider the most powerful adversary who can query the secret keys of the client in all except one unexposed time period. The proposed system has several advantages, some of them are listed below: (a) More than one file in cloud can be extracted with single authentication key, (b) Support multiple file requests at same time without any delay, (c) Highly fault tolerant and past system's key confusions are removed and (d) Best in performance and less time consumption because of using single key to multiple files at a same time.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

System Implementation

The proposed system is implemented with the help of following modules, all of them are described in detail below:

A. User Authorization and Authentication

The User Authorization and Authentication module allows the user to register their identity and login into the system without any interruptions. Authentication is one of most popular and important factor to enter into the required portals and applications. This authorization module allows the user (Data Owner/Data User) to authenticate themselves into the system with proper identities such as Name, Mobile Number, E-Mail-Id, Address and so on.

Once the authentication process is done, the users have specific rights to proceed into the application and access all the features present into it.In order to find a new route, the source node sends route request message to other neighbor nodes which are presented in the network till destination is reached or to find the active route. Then route reply is send back to the source node once the route is identified. The nodes on active route communicate with each other by passing hello messages periodically to its immediate neighbor. If a node does not receive a reply then it deletes the node from its list and sends route error to all the members in the route.

B. Cloud Data Auditing

Usually in cloud storage scheme, all the users are preserving their data into remote system with different types of security precautions, but all the user's data need to audit by some third-parties, they are called as Third-Party Providers (TPA). The TPAs can verify the data by download the content uploaded by data owners and cross check it with cloud based strategies such as owner identity, file type and so on. But the lacking occurs here, the person who audit the data is of course a third-party, so that the owner cannot says the system is 100% secure and private, because of third-party auditing. In the proposed approach, a new mechanism is introduced, called Owner Data Auditing scheme, which allows the data owners to audit and process the data for further manipulations.

The TPAs can only monitor the user details such as Data Owner Identity maintenance and Data User Identity maintenance and they do not have an option to update any content over the owner's data. So, the proposed approach data auditing scheme is safer and secure compare to existing cloud schemes.

C. File Encryption and Integrity Verification

The main objective of this system is to preserve the security in owner data, in which the file is encrypted and then the file is stored to the remote storage server. So if any unauthorized user request for the file, the storage will initially declined their request or else if any user get the file, the files is encrypted and it is not in a readable format.

The file integrity is maintained with high level of data security by means of Advanced Encryption Standard (AES) algorithm, which transforms the plain text of data into unreadable format of data, so unauthorized users cannot modify the content present into it and the owner can feel free to preserve the data into remote server with strong key nature.

D. Hint Text Processing

Simply, says hint text is the process of generating file identity, If the owner uploads hundreds of data into cloud server, the owner needs to identify the file based on some important hints associated with it (Ex. If the owner uploads file regarding Bank Applicatioin, the hint is Bank). This process fully involves in the context of generating efficient hint texts against the given data. Once the user inputting the data this system asks the user to provide the hint text for manipulating the data against encryption, after that the hint text and the sampling data will be forwarded to the user’s mail for clarification.

E. Dynamic Decryption

In cryptography, encryption is the process of transforming information (referred to as plaintext) using an algorithm (called cipher) to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key. The result of the process is encrypted information (in cryptography, referred to as cipher text).

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

In many contexts, the word encryption also implicitly refers to the reverse process, decryption (e.g. “software for encryption” can typically also perform decryption), to make the encrypted information readable again (i.e. to make it unencrypted).

IV. LITERATURE SURVEY

In the year of 2015, the authors "R. Curtmola, J. Garay" proposed a paper titled “Searchable Symmetric Encryption: Improved Definitions And Efficient Constructions", in that they described such as: Searchable symmetric encryption (SSE) allows a party to outsource the storage of its data to another party (a server) in a private manner, while maintaining the ability to selectively search over it. This problem has been the focus of active research in recent years. In this paper we show two solutions to SSE that simultaneously enjoy the following properties: Both solutions are more efficient than all previous constant-round schemes. In particular, the work performed by the server per returned document is constant as opposed to linear in the size of the data. Both solutions enjoy stronger security guarantees than previous constant-round schemes. In fact, we point out subtle but serious problems with previous notions of security for SSE, and show how to design constructions which avoid these pitfalls.

Further, our second solution also achieves what we call adaptive SSE security, where queries to the server can be chosen adaptively (by the adversary) during the execution of the search; this notion is both important in practice and has not been previously considered. Surprisingly, despite being more secure and more efficient, our SSE schemes are remarkably simple. We consider the simplicity of both solutions as an important step towards the deployment of SSE technologies. As an additional contribution, we also consider multi-user SSE. All prior work on SSE studied the setting where only the owner of the data is capable of submitting search queries. We consider the natural extension where an arbitrary group of parties other than the owner can submit search queries. We formally define SSE in the multi-user setting, and present an efficient construction that achieves better performance than simply using access control mechanisms.

In the year of 2016, the authors "P. Van and S. Sedghi" proposed a paper titled “Computationally Efficient Searchable Symmetric Encryption", in that they described such as: Searchable encryption is a technique that allows a client to store documents on a server in encrypted form. Stored documents can be retrieved selectively while revealing as little information as possible to the server. In the symmetric searchable encryption domain, the storage and the retrieval are performed by the same client. Most conventional searchable encryption schemes suffer from two disadvantages.

First, searching the stored documents takes time linear in the size of the database, and/or uses heavy arithmetic operations. Secondly, the existing schemes do not consider adaptive attackers; a search-query will reveal information even about documents stored in the future. If they do consider this, it is at a significant cost to the performance of updates. In this paper we propose a novel symmetric searchable encryption scheme that offers searching at constant time in the number of unique keywords stored on the server. We present two variants of the basic scheme which differ in the efficiency of search and storage. We show how each scheme could be used in a personal health record system.

In the year of 2015, the authors "X. Song and D.Wagner" proposed a paper titled “Practical Techniques For Searches On Encrypted Data", in that they described such as: It is desirable to store data on data storage servers such as mail servers and file servers in encrypted form to reduce security and privacy risks. But this usually implies that one has to sacrifice functionality for security. For example, if a client wishes to retrieve only documents containing certain words, it was not previously known how to let the data storage server perform the search and answer the query, without loss of data confidentiality. We describe our cryptographic schemes for the problem of searching on encrypted data and provide proofs of security for the resulting crypto systems.

Our techniques have a number of crucial advantages. They are provably secure: they provide provable secrecy for encryption, in the sense that the untrusted server cannot learn anything about the plaintext when only given the ciphertext; they provide query isolation for searches, meaning that the untrusted server cannot learn anything more about the plaintext than the search result; they provide controlled searching, so that the untrusted server cannot search for an arbitrary word without the user's authorization; they also support hidden queries, so that the user may ask the untrusted server to search for a secret word without revealing the word to the server. The algorithms presented are simple, fast (for a document of length n, the encryption and search algorithms only need O(n) stream cipher and block cipher operations), and introduce almost no space and communication overhead, and hence are practical to use today.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 3, March 2018

V. CONCLUSION

In this system, we further study on how to deal with the key exposure problem in cloud storage auditing. We propose a new paradigm called strong key-exposure resilient auditing scheme for secure cloud storage. In this paradigm, the security of the cloud storage auditing not only earlier than but also later than the key exposure can be preserved. We formalize the definition and the security model of this new kind of cloud storage auditing and design a concrete scheme. The security proof and the experimental results demonstrate that the proposed scheme is secure and efficient.

REFERENCES

[1] F. Sebe, J. Domingo-Ferrer, A. Martinez-balleste, Y. Deswarte, and J. Quisquater, "Efficient Remote Data Integrity checking in Critical Information Infrastructures," IEEE Transactions on Knowledge and Data Engineering, vol. 20, no. 8, pp. 1-6, 2008.
[2] C. Wang, K. Ren, W. Lou, and J. Li, "Toward Publicly Auditable Secure Cloud Data Storage Services," IEEE Network, 2010.
[3] Y. Zhu, H. Wang, Z. Hu, G. J. Ahn, H. Hu, and S. S. Yau, "Efficient Provable Data Possession for Hybrid Clouds," Proc. 17th ACM Conference on Computer and Communications Security, 2010.
[4] K. Yang and X. Jia, "Data Storage Auditing Service in Cloud Computing: Challenges, Methods and opportunities," World Wide Web, 2012.
[5] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," IEEE Trans. Parallel and Distributed Systems, vol. 22, no. 5, pp. 847-859, May 2011.
[6] Y. Zhu, H.G. Ahn, H. Hu, S.S. Yau, H.J. An, and C.J. Hu, "Dynamic Audit Services for Outsourced Storages in Clouds," IEEE Trans. on Services Computing, vol. 6, no. 2, pp. 409-428, 2013.
[7] C. Wang, S. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, Vol. 62, No. 2, pp. 362-375, 2013.
[8] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," IEEE Trans. Parallel and Distributed Systems, Vol. 24, No. 9, pp. 1717-1726, 2013.
[9] J. Yuan and S. Yu, "Public Integrity Auditing for Dynamic Data Sharing With Multiuser Modification," IEEE Transactions on Information Forensics and Security, 2015.
[10] B. Wang, B. Li, and H. Li. "Public auditing for shared data with efficient user revocation in the cloud," INFOCOM 2013 Proceedings IEEE, pp. 2904-2912, 2013.
[11] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MR-PPDP: Multiple-Replica Provable Data Possession," Proc. 28th IEEE International Conference on Distributed Computing Systems, pp. 411-420, 2008.
[12] A. Barsoum, and M. Hasan, "Provable Multireplica Dynamic Data Possession in Cloud Computing Systems," IEEE Transactions on Information Forensics and Security. vol. 10, no. 3, pp. 485-497, Mar. 2015.
[13] J. Yu, K. Ren, C. Wang, and V. Varadharajan, "Enabling Cloud Storage Auditing with Key-Exposure Resistance," IEEE Transactions on Information Forensics and Security. vol. 10, no. 6, pp. 1167-1179, Jun. 2015.
[14] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security, pp. 598-609, 2007.
[15] A. Juels, and B. Kaliski, "PORs: Proofs of Retrievability for Large Files," Proc. 14th ACM Conf. Computer and Comm. Security, pp. 584-597, 2007.
[16] H. Shacham and B. Waters, "Compact Proofs of Retrievability," Advances in Cryptology-Asiacrypt'08, pp. 90-107, 2008.
[17] Y. Dodis, S.P. Vadhan, and D. Wichs, "Proofs of Retrievability via Hardness Amplification," Proc. Theory of Cryptography Conf. Theory of Cryptography, pp. 109-127, 2009.
[18] G. Ateniese, R.D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession," Proc. 4th International Conference on Security and Privacy in Communication Networks, 2008.