IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

LITURATURE SURVEY ON IASS: AN IMPROVED AUTHENTICATION USING SECRET SHARING BASED ON HADOOP

Gholap Pravin Sanjay*, Gagare Ganesh Sahebrao, Kanaskar Shekhar Shantaram, Gunjal Jayesh Suresh

* Department Of Computer, Samarth Group Of Institution's College Of Engineering, Belhe, Pune, India..

ABSTRACT

In today's modern world with high tech technology everyone prefer to store their Personal data in the Cloud which may has account numbers, passwords and other important information that could be used and misused by a miscreant, a opponent, or a court of law. These data are retrieved, copied, and archived by Cloud Service Providers (CSPs), often without users' permission and control. Self-destructing data plays a vital role in protecting the user data's privacy. All the data stored at servers and their copies become destructed after a user-specified time, and also this data became unreadable for any user intervention. The decryption key is destructed after the user-specified time that is ttl(time-to-live) field. In proposed scheme, we present self-destructing data system that meets this challenge through a novel integration of secure cryptography techniques with active storage techniques based on '*hadoop'*. We implement a proof-of-concept SeDas prototype. By functionality and security properties evaluate the SeDas prototype, the results conclude that SeDas is practical to use and achieve all the privacy-preserving aims described. Compared to the system without self-destructing data mechanism, performance of uploading and downloading with the proposed SeDas acceptably decreases, while latency for uploading and downloading operations with self-destructing data mechanism increases.

KEYWORDS: Active storage, Cloud computing, data privacy, self-destructing data, Cloud Service Providers.

INTRODUCTION

With evolution of Cloud computing and popularization of mobile Internet, cloud services are becoming much necessary for peoples part of life. Peoples are more or less desired to submit or post some personal private information on the Cloud by the Internet. When people does this, they subjectively hope service providers will provide security policy to secure their data from hacking, so others people will not infect their privacy.

A secret sharing scheme starts with a secret and then assume from it certain shares which are distributed to a group of users (i.e., participants). The secret may be variously resolve (i.e., recovered) only by certain predetermined subgroups of users which constitute the access structure. The important category of access structure is the (w,N)-threshold access structure in which, given N shareholders, an authorized group consist of any w or more participants and any group of at most w−1 participants is an illegal group.

As nation are attracted more and more to the Internet system and Cloud system, security of this privacy takes much more risk. On the one hand, when database is being processed, reconstruct and stored by the current system or network, systems or network must cached, copy or archive it. These copies are essential for system computer and the network. However, people don't have knowledge about these copies and cannot control them, so these copies may leak their confidentiality. On the other hand, their privacy also can be leaked via Cloud Service Providers (CSPs') , hackers' intrusion or some fair actions. These problems present dangerous challenges to protect people's privacy. Secret sharing has broad applications in this real world and can be used for situations in which access to important resources to be protected. There is old story which have motivated the secret sharing principle [8]: a group of pirates accidentally detected a map that would lead them to an enclave full of treasure. Who was going to be entrusted to keep the map? Safe result is: the map should be divided into pieces such that all pieces are needed to reconstruct the map and missing any piece would make the map which is not readable. Thus, every pirate was given one such piece. Another important function or operation of secret sharing is e-voting where the vote of every single will be absolutely and correctly

counted in the voting result but there is no way for their people (including candidates and authorities) to know whom the individual voted for [12].

Today database and networking era, sharing data secretely is also a fundamental issue in network security and can be used in key administration and multi-party secure computation [7]. Since the concept of secret sharing, along with an efficient structure to accomplish it, was proposed by Shamir in 1979 [20] Blakley also did the similar work at same time [5], there have been many papers approaching Shamir's scheme and designing new secret sharing schemes [2], [7], [9], [10], [12], [15]. Secret sharing schemes can be classified into various categories according to different criteria's. In terms of multiple numbers of secrets to be divided, two classes can be identified: single secret and multiple secrets.

LITERATURE SURVEY

Lingfang Zeng[16],suggested that the secrete data stored in the Cloud has personal important information that could be used and misused by any miscreant, a competitor, or a court of law. These data are cached, copied, saved by Cloud Service Providers (CSPs), often without users' permission and control. Self-destructing data aiming is to protect the users data's privacy. In the previous earlier system there are multiple disadvantages are available. In this Hacker can attack the confidential data and gain all the information from the database. This is big disadvantages of this system. client want to security of the data which is secure from other's. In this hacking process the sensitive data can be modified by anyone, or if anyone can do changes in this client database then it is very harmful for client.

Shaofeng Zou[17], developed a novel information theoretic approach is proposed to solve the problem of secret sharing, in which a distributor distributes one or more secrets to participants in such a manner that for each secret only qualified sets of users can reconstruct this secret by combining their shares together while nonqualified sets of users does not obtain any information about the secret even they pool their shares together. While existing secret sharing systems assume that communications between the distributor and participants are noiseless, this paper takes a more practical assumption that the distributor delivers shares to the participants via a noisy broadcast channel. Thus, in contrast to the available solutions that are mainly based on number theoretic tools, an information theoretic approach is introduced, which exploits the channel randomness during delivery of shares as additional resources to fulfill secret sharing requirements. Secret sharing problems can be reformulated as same secure communication problems via wiretap channels, and can hence be resolved by employing the powerful information notional security techniques.

Lingfang Zeng[3], proposed improved Washington's Vanish system for self-destructing data under cloud computing, and it is open to "hopping attack" and "sniffer attack". In this paper working of Safe Vanish, to prevent hopping attacks by way of Increasing the length of the key shares to rise the attack cost did some more enhancement on the Shamir Secret Sharing algorithm implemented in the Original Vanish system. They present an improved approach to prevent sniffing attacks by using the public key cryptography system to protect from sniffing operations. In addition, they evaluate analytically the functionality of the proposed Safe Vanish system.

Wang et al. [6] proposed a public auditing scheme consists algorithms (KeyGen, SigGen, VerifyProof GenProof,). KeyGen is a key generation algorithm that should be run by the user to set the scheme. To user to generate verification of metadata SigGen is used , which may consist of MAC or other related information that will be used for checking. GenProof is run on the cloud server to give a proof of data storage while VerifyProof is run by the TPA to audit the proof defi the cloud server. public auditing system can be constructed from the above auditing scheme this is done in two phases, Setup and Audit:

- Setup: The user starts the public and secret parameters of the system by execution of KeyGen, and preprocesses the data file F by using SigGen to generate the verification metadata. The user stores the data file F on the cloud server, deletes local copy, and publishes the verification metadata to TPA for audit. As part of pre-processing, the user alter the data F by expanding it or including additional metadata is to be stored at server.
- Audit: TPA issues an audit challenge or message to the cloud server to make sureness that the cloud server has regained the data file F at the time of the audit. The cloud server will finds a reply mssage from a function of the stored data file by executing GenProof. Using the verification metadata, the TPA does varification of the response via VerifyProof.

Yu Zhang[21] Introduced that paper we present a reconfigurable calculating solution that can provide high-performance, flexible processing capabilities for the storage nodes. The dynamic reconfiguration upturns the functional density; however, the configuration self results in extra overhead, which may make the overall performance be downgraded. In the future works, we will implement multiple Processing Elements in the reconfigurable accelerator, and design an efficient method for dispatching the Processing Elements to hide the reconfiguration latency to improve the performance.

FU Xiao[13] Realized emails were being watched by the government. For the advantage that Big Data technologies such as large distributed storage and user behaviour analysis and so on emails became one of the highly popular Big Data that has been targeted at as a large source of intelligence by some organizations keeps eye on public accounts every hour every day. research work was just opposite to what the NSA has did: To design and implement a system which can store emails securely, and terminate them clearly when they expired. In another word, a self-destructing emails system. But in this system there is no parallel processing for multiuser access.

Yu Zhang[21] Introduced that paper we extant a reconfigurable computing solution that can provide flexible, high processing capabilities for the storage nodes. The active reconfiguration increases the functional density; however, the configuration self results in extra upstairs, which may make the complete performance bee normously downgraded. In the future works, we will implement multiple Processing Elements in the reconfigurable accelerator, and design an efficient method for dispatching the Processing Elements to hide the reconfiguration latency to improve the performance.

FU Xiao[13] Realized emails were being observed by the government. For the benefits that Big Data technologies such as user behaviour analysis and so on bring, emails became one of the most popular widely used Big Data that has been targeted at as a huge source of intelligence by some organizations snooping on public accounts every hour every day. Their research work was just reverse to what the NSA has done: To design and implement a system that can store emails protectively, and finish them clearly when they expired. In another word, a self-destructing emails system. But in this system there is no parallel processing for multiuser access.

Tina Miriam John[14], suggested the increasing performance and decrease in cost of processors and memory are causing system intelligence to move from the CPU to peripherals such as removable storage devices. Storage system designers are using this trend toward excessive computation capability to perform much complex processing and optimizations directly inside the storage devices. Such kind of optimizations are performed at low levels of the storage protocol. Other factor to consider is the current trends in storage mechanics, density and electronics, which are eliminating the traffic encountered while moving data off the media and putting pressure on interconnected systems and host processors to move data more efficiently. Previous working of active storage has taken more advantage of the power of extra processing on individual disk drives to run application-level code. The experimental setup they used to test out this application was a 16 node Linux cluster. Several test runs were conducted, with changing number of targets and Varying data set sizes.

M. Plastoi [18],worked on wireless sensor networks technology is a rapidly growing domain, credited in the area of civil and military applications. In the same time with technological progress, new and risky information security threats have involved. In this paper we considered that a node self-destruction procedure must be performed as a last stage in the sensor node lifecycle in order to assure the confidentiality regarding information like: network topology, measurement data collected by sensors, encryption/authentication algorithms and key-exchange working etc. that can be opend otherwise through reverse engineering methods. Our methodology relies on an efficient power observing scheme, constructed on combined in-network and predictive data, which discover the low battery nodes and starts a self-destruction procedure for that nodes.

Deghaili, R. [19], worked on distributed systems, it is needed to establish trust before the entities interact together. This trust establishment process involves making every entity ask for some identifications from the other entity, which indicates some privacy loss for both parties. We propose a model to reach the right privacy-trust balance in distributed environments. Each entity aims to join a group in order to protect its privacy. Interface between entities is then changed by interaction between groups on behalf of the members. Data shared between groups is saved from dissemination by a self-destruction process. Simulations performed on the proposed model using the platform of Aglets show that entities requesting a service need to give up private information details when their past involvements are not good, or when the server entity is of a paranoid nature. The privacy loss in all cases is formatted and controlled.

In the existing system there are multiple disadvantages are available. In this Hacker can attack the confidential data and gain all the information from the database. This is big disadvantages of this system. Because client want to security of the data which is confidential from other's. In this hacking process the sensitive data can be modified by anyone, or if anyone can do changes in this client database then it is very harmful for client.

PROPOSED SYSTEM

Secret sharing in cryptography , refers to a method for distributing a *secret* amongst a group of participants, *shares* of the secret allocated to each . In the proposed system we used Hadoop for provide much more security to user database. As people rely more and more on the Internet and Cloud technology, security of their privacy takings additional risks. when data is being processed, transformed and stored by the current computer system . systems or network must cache, copy or stored it. These copies are essential for systems and the network. The secret can only be reconstructed when the all shares are combined together; individual shares are of no use on their own

Need:

- Gives tight control and removes single point vulnerability
- Individual key share holder cannot change/access the data.

The SeDas system describes two new modules, a self-destruct method object that is associated with each secret key part and continuation time parameter for each secret key part. In this case, SeDas can meet the requests of self-destructing data with manageable survival time while users can use this system as a general object storage system. Our contributions are abstract as follows Distribution algorithm which is used as the core algorithm to implement client (users) distributing all keys in the object storage system. Use these methods to implement a safety destruct with equal divided key.

Store and manage: Based on active storage framework, we use an object-based storage to store and manage the equally divided key on systems. We implemented a SeDas prototype.

Goal is to divide data *D* (e.g., the safe combination) into n pieces D1,D2….Dn such that:

- Knowledge of any k or more D pieces makes D easily computable.
- Knowledge of any k -1 pieces leaves D completely undetermined

This concept is called (k, n) threshold scheme. If k=n then all participants are required together to reconstruct the secret. Uploading file process: When a user uploads a file to a storage system and stores his key in this SeDas system, he should identify the file, the key and *TTL (time-to-live)*arguments for the uploading procedure. In these codes, we assume key has been read from the file. The ENCRYPT procedure uses a common encrypt algorithm or user-defined encrypt algorithm. On the uploading data to storage server, key shares generated by *Shamir Secret Sharing* algorithm will be used to construct active storage object (ASO) in storage node in the SeDas system. Downloading file process: Any user who has needed permission can download data stored in the data storage system. The data should be in decrypted form before use.

The whole logic is implemented in user's application.

In the proposed system there are multiple advantages are available. In this system we used the Hadoop technology for provide larger security of user secret or confidential data like, hacking of secret keys or passwords. The main advantage is to secure data. In this system the data is more secure than existing system. Because the data is access by only authorized users. In this proposed system if the hacker can try to get client confidential data, then this data made useless by destroying the private key. This type of securities provided by proposed system.

In this system the user key or password is distributed over multiple servers using Shamir's algorithm. By using this algorithm, the secret password or key is distributed over multiple server machines and using this approach the hackers can't hack the user password or secret key.

MATHEMATICAL BACKGROUND

$f(x)=a_0+a_1x+a_2x_2+\ldots+a_{k-1}x^{k-1}$

$K_d=\{a_0,a_1,a_2,\ldots a_n\}$

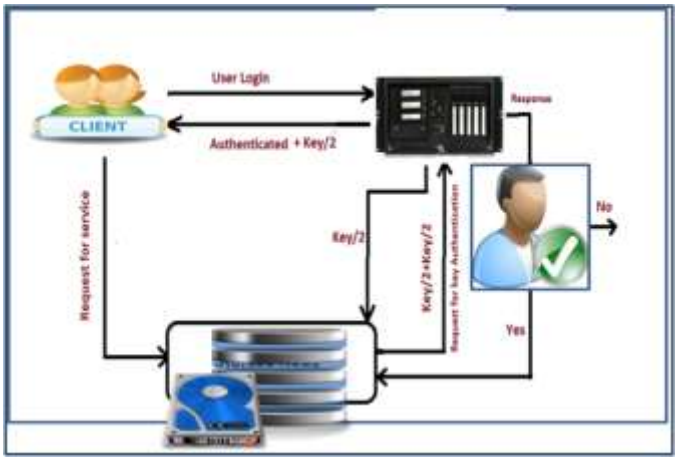
key is distributed among the node and on the knowledge of k or K_{di} the key is reconstructed using Lagrange's polynomial.

$l_0=x-x_1/x_0-x_1$
$l_1=x-x_0/x_1-x_0$
$l_2=x-x_0/x_2-x_0$

The key can be reformed using following equation

$f(x)=\sum_{j=0}^{k} yj.lj(x)$

SYSTEM ARCHITECTURE



Above Fig. shows the architecture of system. There are three parties

i) **Metadata server** (MDS): MDS is used for user management, server management, session management and file metadata management.
ii) **Application node**: It is a client to use storage services of the system.
iii) **Storage node**: Each storage node is an OSD (Object Storage Device).

EXPECTED RESULT

As we are using hadoop in this project so there will be increase in performance as well as flexibility of the SeDas project. Also there will be high number of client handling as we using hadoop which is helpful for parallel processing.

CONCLUSION

Hadoop has been very efficient solution for companies dealing with the data in petabytes. It has solve many problems in industry related to hug data management and distributed system and it produce much more security to the database.

Data outsourcing using Shamir's secret sharing methods are compared using the Hadoop technology. The Secret Sharing methods are computationally inexpensive when compared with the normal encryption techniques. The security provided by these methods is not bounded by the computational capabilities of the existing hardware.

By taking the above points into consideration we determine that, due to the distributed nature of the hadoop system, information distribution of data is the more optimal approach for data outsourcing.

REFERENCES

[1] Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.
[2] Li Bai and Xukai Zou. A proactive secret sharing scheme in matrix projection method. *International Journal of Security and Networks*, 4(2):15–23, 2009.

[3] L. Zeng, Z. Shi, S. Xu, and D. Feng, "Safevanish: An improved data self-destruction for protecting data privacy," in *Proc. Second Int. Conf. Cloud Computing Technology and Science (CloudCom)*, Indianapolis, IN, USA, Dec. 2010, pp. 521–528.
[4] R. Perlman, "File system design with assured delete," in *Proc. Third IEEE Int. Security Storage Workshop (SISW)*, 2005.
[5] G. R. Blakley. Safeguarding cryptographic keys. *American Federation of Information Processing Societies Proceedings*, 48:313–317, 1979.
[6] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for storage security in cloud computing," in *Proc. IEEE INFOCOM*,2010.
[7] M. Franklin and M. Yung. Communication complexity of secure computation. *STOC*, pages 699–710, 1992.
[8] R. Gennaro. Theory and practice of verifiable secret sharing. *Ph.Dthesis, MIT*, 1995.
[9] H. Ghodosi, J. Pieprzyk, and R. Safavi-Naini. Secret sharing in multilevel and compartmented groups. *Lecture Notes in Computer Science*, 1438:367–378, 1998.
[10] J. He and E. Dawson. Multistage secret sharing based on one-way function. *Electronics Letters*, 30:1591–1592, 1994.
[11] Y. Tang, P. P. C. Lee, J. C. S. Lui, and R. Perlman, "FADE: Secure overlay cloud storage with file assured deletion," in *Proc. SecureComm*, 2010.
[12] S. Iftene. General secret sharing based on the chinese remainder theorem with applications in e-voting. *Electronic Notes in Theoretical Computer Science*, 186:67–84, 2007.
[13] RFU Xiao, WANG Zhi-jian, WU Hao, YANG Jia-qi, WANG Zi-zhao, "How to send a Self-destructing Email, a method of self-destructing email system," in *Proc.* IEEE DOI 10.1109/BigData.Congress.2014, pp. 304–309.
[14] Tina Miriam John, Anuradharthi Thiruvenkata Ramani, John A. Chandy, "Active Storage using Object-Based Devices". Second International Workshop on High Performance I/O Systems and Data Intensive Computing (HiperIO'08) IEEE, 2008, pp. 472-478.
[15] K. M. Martin, J. Pieprzyk, R. Safavi-Naini, and H. Wang. Changing thresholds in the absence of secure channels. *Lecture Notes in Computer Science*, 1587:177–191, 1999.
[16] Lingfang Zeng , Shibin Chen , Qingsong Wei , and Dan Feng," SeDas: A Self-Destructing Data System Based on Active Storage Framework", IEEE TRANSACTIONS ON MAGNETICS, VOL. 49, NO. 6, JUNE 2013,pp:2548-2554.
[17] Shaofeng Zou, *Student Member, IEEE*, Yingbin Liang, *Member, IEEE*, Lifeng Lai, *Member, IEEE*, and Shlomo Shamai (Shitz), *Fellow, IEEE,"* An Information Theoretic Approach to Secret Sharing*",* IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 61, NO. 6, JUNE 2015, pp:3121-3136.
[18] M.Plastoi, Curiac,D.-I, "Energy-Driven methodology for node self-destruction in wireless sensor networks", Applied computational intelligence and informatics, 2009.SACI '09.5th international symposium', pp:319-322.
[19] Deghaili,R.,Chehab,A.,Kayssi,A., "Trust-privacy tradeoffs in distributed systems", Innovations in information technology[IIT],2008.IIT2008,pp:39-43.
[20] Yu Zhang, Dan Feng," An Active Storage System for High Performance Computing**",** Advanced Information Networking and Applications, 2008. AINA 2008. 22nd International Conference. pp: 644 – 651.
[21] Ron Steinfelda, Josef Pieprzyka, and Huaxiong Wang. Lattice-based threshold changeability for standard shamir secret-sharing schemes. *IEEE Transactions on Information Theory*, 53:2542–2559, 2007.
[22] D. R. Stinson, editor. *Cryptography: Theory and Practice*. CRC Press, Inc., Boca Raton, Florida, USA, 1995.
[23] Tamir Tassa. Hierarchical threshold secret sharing. *Journal of Cryptology*, 20(2):237–264, November 200
[24] Tamir Tassa, Nira Dyn, and U Ci. Multipartite secret sharing by bivariate interpolation. In *33rd International Colloquium on Automata, Languages and Programming, ICALP 2006, Lecture Notes in Comput. Sci. 4052*, pages 288–299. Springer-Verlag, 2006.