© 2016 IJSRSET | Volume 2 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Constraint Based Automatic Destruction of Data in Cloud Computing

S. Praveena, V. Sivaranjani , B.Preyankha, B. Hema

Information Technology Information Technology, Velammal Institute of Technology Velammal Institute of Technology, Chennai, Tamilnadu, India

ABSTRACT

Cloud computing is defined as a type of computing that relies on sharing computing resources rather than having local servers or personal devices to handle applications. Any discussion involving data must address security and privacy, especially when it comes to managing sensitive data. After the recent leaks of countless millions of user login credentials, the privacy of your cloud-based data is another consideration. In order to tackle this problem, we propose a novel secure data self-destructing scheme in cloud computing. We create three way self-distraction scheme to secure the data using AES/DES Double Encryption Algorithm to secure the data. By using this, sensitive data will be securely self-destructed after a user-specified expiration time. Secondly, user can access the data only one time from the cloud. At last, if the user enters the incorrect key three times, the data will be self-distracted. Comprehensive comparisons of the security properties indicate that this scheme proposed by us satisfies the security requirements and is superior to other existing schemes.

Keywords: Advanced Encryption Scheme, Key Policy Attribute Based Encryption with Time Specified Attributes, Cipher text, Decryption, Authentication.

I. INTRODUCTION

The requirements specification is a technical specification of requirements for the software products. It is the first step in the requirements analysis process it lists the requirements of a particular software system including functional, performance and security requirements. The requirements also provide usage scenarios from a user, an operational and an administrative perspective. The purpose of software requirements specification is to provide a detailed overview of the software project, its parameters and goals. This describes the project target audience and its user interface, hardware and software requirements. It defines how the client, team and audience see the project and its functionality.

II. METHODS AND MATERIAL

A. Existing System

We believe that sharing data among users is perhaps one of the most engaging features that motivate's cloud storage. Regarding availability of files, there are a series of cryptographic schemes which go as far as allowing a thirdparty auditor to check the availability of files on behalf of the data owner without leaking anything about the data, or without compromising the data owner's anonymity. The problem will arise when a file is shared to multiple users.

Disadvantages of Existing System:
• Privacy issues
• Large Amount of space need in Cloud.

B. Proposed System

We propose a key-policy attribute-based encryption with time-specified attributes (KPTSABE), a novel secure data self-destructing scheme in cloud computing. In the

KP-TSABE scheme, every ciphertext is labeled with a time interval while private key is associated with a time instant. The ciphertext can only be decrypted if both the time instant is in the allowed time interval and the attributes ssociated with the ciphertext satisfy the key's access structure.

Advantages of Proposed System:

- Security issue will not be there.
- Privacy issues are minimized.
- Reducing the space required to store data in cloud.

C. Literature Survey

1. Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud (2014)

With cloud data services, it is commonplace for data to be not only stored in the cloud, but also shared across multiple users. Unfortunately, the integrity of cloud data is subject to skepticism due to the existence of hardware/software failures and human errors. Several mechanisms have been designed to allow both data owners and public verifiers to efficiently audit cloud data integrity without retrieving the entire data from the cloud server.

However, public auditing on the integrity of shared data with these existing mechanisms will inevitably reveal confidential information identity privacy to public verifiers. we propose a novel privacy preserving mechanism that supports public auditing on shared data stored in the cloud. In particular, we exploit ring signatures to compute verification metadata needed to audit the correctness of shared data.

With our mechanism, the identity of the signer on each block in shared data is kept private from public verifiers, who are able to efficiently verify shared data integrity without retrieving the entire file. In addition, our mechanism is able to perform multiple auditing tasks simultaneously instead of verifying them one by one.

2. Hybrid attribute- and reencryption- based key management for secure and scalable mobile applications in clouds (2013)

Outsourcing data to the cloud are beneficial for reasons of economy, scalability, and accessibility, but significant technical challenges remain. Sensitive data stored in the cloud must be protected from being read in the clear by a cloud provider that is honest-but-curious.

Additionally, cloud-based data are increasingly being accessed by resource-constrained mobile devices for which the processing and communication cost must be minimized. Novel modifications to attribute-based encryption are proposed to allow authorized users access to cloud data based on the satisfaction of required attributes such that the higher computational load from cryptographic operations is assigned to the cloud provider and the total communication cost is lowered for the mobile user. Furthermore, data re-encryption may be optionally performed by the cloud provider to reduce the expense of user revocation in a mobile user environment while preserving the privacy of user data stored in the cloud. The proposed protocol has been realized on commercially popular mobile and cloud platforms to demonstrate real-world benchmarks that show the efficacy of the scheme. A simulation calibrated with the benchmark results shows the scalability potential of the scheme in the context of a realistic workload in a mobile cloud computing system.

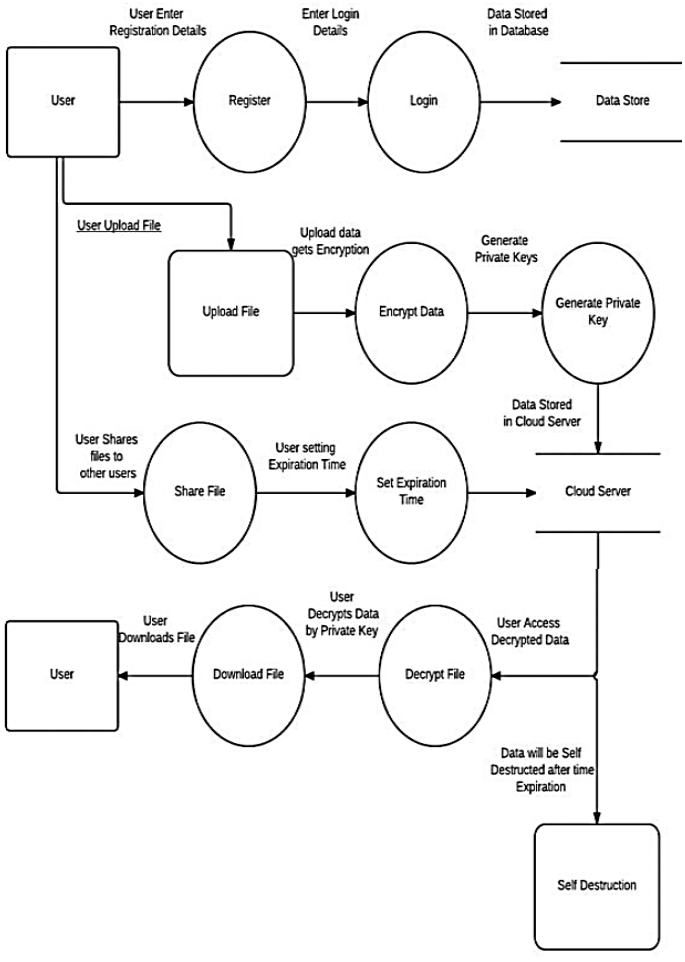
3. Achieving secure, scalable, and fine-grained data access control in cloud computing (2010)

Cloud computing is an emerging computing paradigm in which resources of the computing infrastructure are provided as services over the Internet. As promising as it is, this paradigm also brings forth many new challenges for data security and access control when users outsource sensitive data for sharing on cloud servers, which are not within the same trusted domain as data owners. To keep sensitive user data confidential against untrusted servers, existing solutions usually apply cryptographic methods by disclosing data decryption keys only to authorized users. However, in doing so, these solutions inevitably introduce a heavy computation overhead on the data owner for key distribution and data management when finegrained data access control is desired, and thus do not scale well. The problem of simultaneously achieving fine-grainedness, scalability, and data confidentiality of access control actually still remains unresolved. This paper addresses this challenging open issue by, on one hand, defining and

enforcing access policies based on data attributes, and, on the other hand, allowing the data owner to delegate most of the computation tasks involved in finegrained data access control to untrusted cloud servers without disclosing the underlying data contents. We achieve this goal by exploiting and uniquely combining techniques of attributebased encryption (ABE), proxy re-encryption, and lazy re-encryption. Our proposed scheme also has salient properties of user access privilege confidentiality and user secret key accountability. Extensive analysis shows that our proposed scheme is highly efficient and provably secures under existing security models.

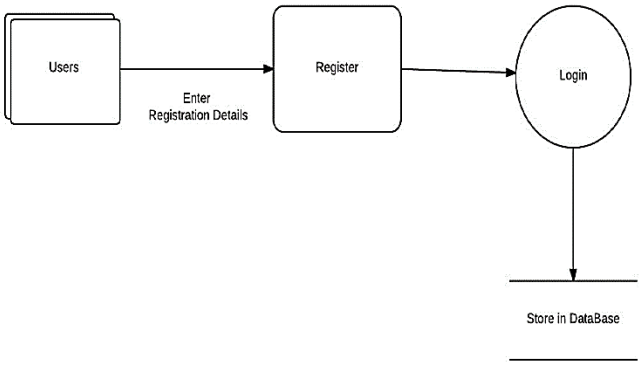
D. Architecture

Here, we introduce the architecture of the system capable of incorporation. Given below is the diagrammatic representation of the proposed system. The system consists of five modules. They are Authentication and Authorization, File Encryption and Data storing to Cloud, File Sharing, File Decryption and Download and Self Destruction of Data



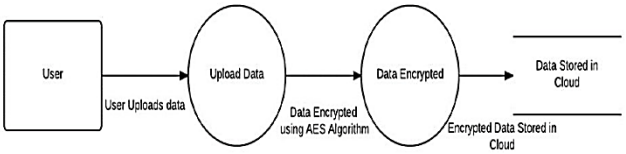
1. Authentication and Authorization

In this module the User have to register first, then only he/she has to access the data base. After registration the user can login to the site. The authorization and authentication process facilitates the system to protect itself and besides it protects the whole mechanism from unauthorized usage. The Registration involves in getting the details of the users who wants to use this application.



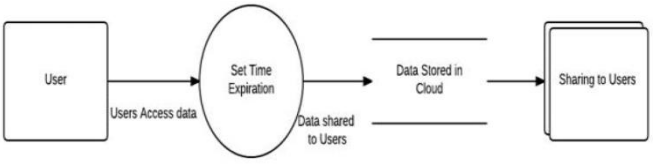
2. File Encryption and Data Storing to Cloud In this module

User Upload the files which he wants to share. At first the uploaded files are stored in the Local System. Then the user upload the file to the real Cloud Storage (In this application, we use Dropbox). While uploading to the Cloud the file got encrypted by using AES (Advanced Encryption Standard) Algorithm and generates Private key. Again the Encrypted Data is converted as Binary Data for Data security and Stored in Cloud.



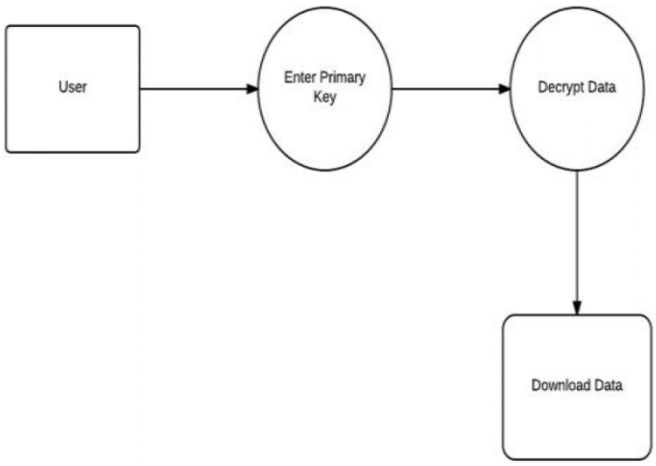
File Sharing

In this module, the uploaded files are shared to the friends or users. In this, the Data Owner set the time to expire the data in Cloud. The Private key of the Shared Data will be send through Email.



3. File Decryption and Download from Cloud

In this Module, the user can download the data by decrypting by using AES (Advanced Encryption Standard) Algorithm. The user should give corresponding Private Keys to decrypt the data. The data will be deleted if the user enters the Wrong Private Key for Three times. If the file got deleted then the intimation email will be sent to the Data owner. The Downloaded Data will be stored in Local Drive.



4. Self-Destruction of Data

The Data will be automatically deleted if the User does not download the file successfully with in the time given by the data owner. If the user downloads the data, then the Self-Destruction will be disabled. If the File got deleted by self-Destruction scheme, the intimation Email will be sent to Data Owner.

III. CONCLUSION

With the rapid development of versatile cloud services, a lot of new challenges have emerged. One of the most important problems is how to securely delete the outsourced data stored in the cloud severs. In this paper, we proposed a novel KP-TSABE scheme which is able to achieve the time-specified ciphertext in order to solve these problems by implementing flexible fine-grained access control during the authorization period and time-controllable self-destruction after expiration to the shared and outsourced data in cloud computing. We also gave a system model and a security model for the KPTSABE scheme. Furthermore, we proved that KPTSABE is secure under the standard model with the decision l-Expanded BDHI assumption. The comprehensive analysis indicates that the proposed KP-TSABE scheme is superior to other existing schemes.

IV. FUTURE WORK

Since this project is all about Sharing files to friends perform computer actions the project has been designed keeping in mind the future scopes. What we have aimed and achieved creating is not a product but a tool to a better automotive environment, a tool can be used to shape many things in the future, thus this project will give rise to many future modifications forking in all directions. Some of the near future scopes of this project are as follows.

There are few interesting problems we will continue to study for our future work. One of them is we can share a file to multi users at a time. We use AES (Advanced Encryption Scheme) to encrypt the Data. In future we may develop this application using different types of advanced algorithm for Encryption. We use Dropbox as a Cloud Server. In Future, we may develop that the user can select the Cloud Server such as Google Drive, Hostinger, Dropbox, AppBox He/She want.

V. REFERENCES

[1] B. Wang, B. Li, and H. Li, "Oruta: Privacypreserving public auditingfor shared data in the cloud," Cloud Computing, IEEE Transactions on, vol. 2, no. 1, pp. 43–56, 2014.

[2] J. Xiong, Z. Yao, J. Ma, X. Liu, Q. Li, and J. Ma, "Priam: Privacy preserving identity and access management scheme in cloud," KSII Transactions on Internet and Information Systems (TIIS), vol. 8, no. 1, pp. 282–304, 2014.

[3] J. Xiong, F. Li, J. Ma, X. Liu, Z. Yao, and P. S. Chen, "A full lifecycle privacy protection scheme for sensitive data in cloud computing," Peerto-Peer Networking and Applications. Online]. Available: http://dx.doi.org/10.1007/s12083-014-0295-x

[4] P. Jamshidi, A. Ahmad, and C. Pahl, "Cloud migration research: A systematic review," Cloud Computing, IEEE Transactions on, vol. 1, no. 2, pp. 142–157, 2013.

[5] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, "Toward efficient and privacy-preserving

computing in big data era," Network, IEEE, vol. 28, no. 4, pp. 46–50, 2014.

[6] X. Liu, J. Ma, J. Xiong, and G. Liu, "Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data," International Journal of Network Security, vol. 16, no. 4, pp. 351–357, 2014.

[7] A. Sahai and B. Waters, "Fuzzy identitybased encryption," in Advances in Cryptology–EUROCRYPT 2005, ser. LNCS, vol. 7371. Springer, 2005, pp. 457–473.

[8] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for finegrained access control of encrypted data," in Proceedings of the 13th ACM conference on Computer and Communications Security. ACM, 2006, pp. 89–98.

[9] A. F. Chan and I. F. Blake, "Scalable, serverpassive, useranonymous timed release cryptography," in Proceedings of the International Conference on Distributed Computing Systems. IEEE, 2005, pp. 504–513.

[10] K. G. Paterson and E. A. Quaglia, "Time-specific encryption," in Security and Cryptography for Networks. Springer, 2010, pp. 1–16

[11] Q. Li, J. Ma, R. Li, J. Xiong, and X. Liu, "Large universe decentralized key-policy attribute-based encryption," Security and Communication Networks, 2014. Online]. Available: http://dx.doi.org/10.1002/sec.997

[12] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in Proceedings of the 28th IEEE Symposium on Security and Privacy. IEEE, 2007, pp. 321– 334.

[13] L. Cheung and C. C. Newport, "Provably secure ciphertext policy abe," in Proceedings of the 14th ACM conference on Computer and communications security. ACM, 2007, pp. 456–465.

[14] B. Waters, "Ciphertext-policy attributebased encryption: An expressive, efficient, and provably secure realization," Public Key Cryptography–PKC 2011, pp. 53–70, 2011.

[15] A. Shamir, "How to share a secret," Communications of the ACM, vol. 22, no. 11, pp. 612–613, 1979.