Hybrid Technique towards Self-Distraction Scheme to Secure Data for Sensitive Data Sharing in Cloud Computing

Vishali. D[1], Indra Gandhi. R[2]
[1, 2] G.K.M. College of Engineering and 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, use can access the data only one time from the cloud. At last, if the user enter 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- Delegate data, guaranteed deletion, confidentiality-protect, fine-grained access control, dynamic computing.

I. INTRODUCTION

Cloud computing provides unlimited "virtualized" resources to users as services across the whole Internet, which hides the details of platform and implementation. Cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. Nowadays, in cloud an increasing amount of data is being stored in the cloud and shared by users with specified privileges, which define the access right of the stored data. The challenge of cloud storage services is the management of the ever-increasing volume of data. Deduplication is the recent technique which makes data management scalable in cloud computing. Data deduplication is a specialized data compression technique for eliminating duplicate copies of repeating data in storage. The technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. Instead of keeping multiple data copies with the same content, deduplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy. Deduplication can take place at either the file level or the block level. For file-level deduplication, it eliminates duplicate copies of the same file. Deduplicationcan also take place at the block level, which eliminates duplicate blocks of data that occur in non-identical files.

Although data deduplication brings a lot of benefits, security and privacy concerns arise as users' sensitive data are susceptible to both insider and outsider attacks. Traditional encryption, while providing data confidentiality, is incompatible with data deduplication. Specifically, traditional encryption requires different users to encrypt their data with their own keys. Thus, identical data copies of different users will lead to different cipher texts, making deduplication impossible. Convergent encryption has been proposed to enforce data confidentiality while making deduplication feasible. It encrypts/decrypts a data copy with a convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users retain the keys and send the ciphertext to the cloud.

Since the encryption operation is deterministic and is derived from the data content, identical data copies will generate the same convergent key and hence the same cipher text. To prevent unauthorized access, a secure proof of ownership (POW) protocol is also needed to provide the proof that the user indeed owns the same file when a duplicate is found. After the proof, subsequent users with the same file will be provided a pointer from the server without needing to upload the same file. A user can download the encrypted file with the pointer from the server, which can only be decrypted by the corresponding data owners with their convergent keys. Thus, convergent encryption allows the cloud to perform deduplication on the cipher texts and the proof of ownership prevents the unauthorized user to access the file.

It seems to be contradicted if we want to realize both deduplication and differential authorization duplicate check at the same time.

II. PROPOSED SYSTEM

ARCHITECTURE

A key-policy attribute-based encryption with time-specified attributes (KP-TSABE), a novel secure data self-destructing scheme in cloud computing. In the KP-TSABE scheme, every cipher text is labeled with a time interval while private key is associated with a time instant. The cipher text can only be decrypted if both the time instant is in the allowed time interval and the attributes associated with the cipher text satisfy the key's access structure.

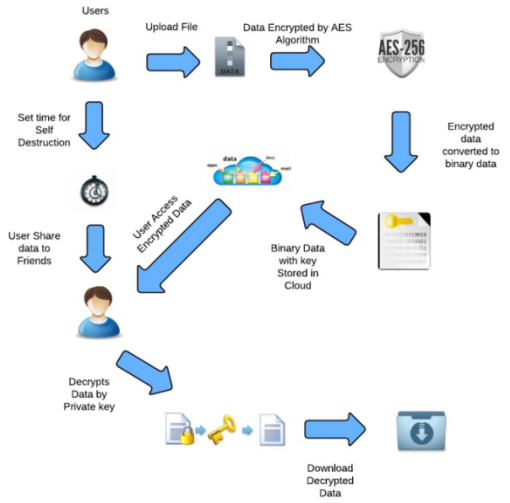

Fig 1. System Architecture for Secure Data Self-Destructing Scheme

Advantages of Proposed System

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

III. MODULE DESCRIPTION

Detail module description for authentication, security listed below and along with brief description

1. Authentication and Authorization
2. File Encryption and Data storing to Cloud.
3. File Sharing
4. File Decryption and Download
5. Self-Destruction of Data

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.

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.

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 enter 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.

Self-Destruction of Data

The Data will be automatically deleted if the User does not downloaded the file successfully with in the time given by the data owner. If the user download 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. SAMPLE SCREEN SHOTS FOR REFERENCES DURING IMPLEMENTATION

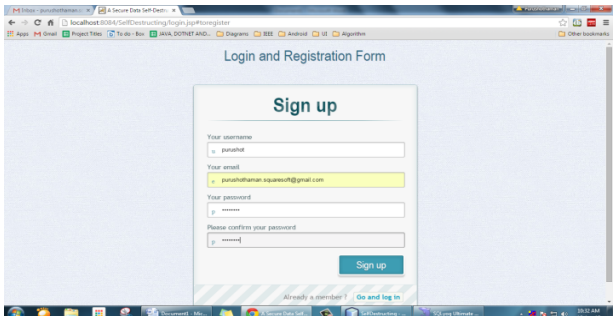

Fig 1. Registration Form for Secure Data

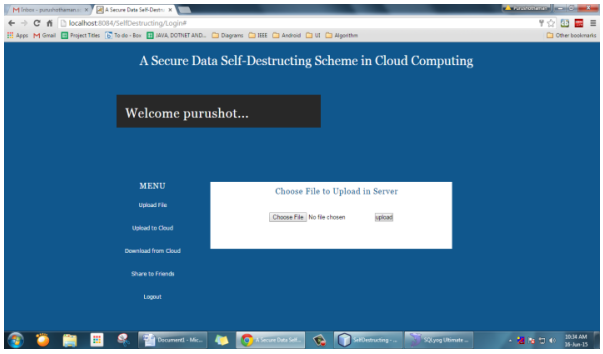

Fig 2. Upload File to Secure Data using Self-Destructing Scheme

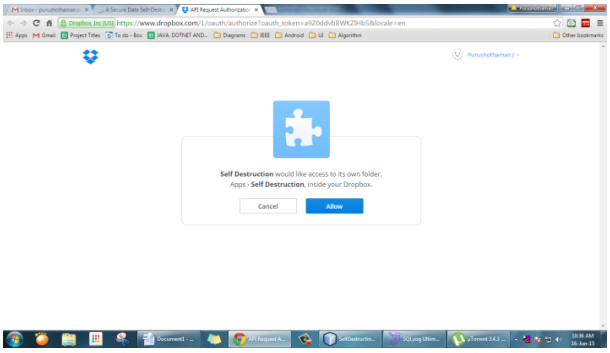

Fig 3. Cloud Authentication

Fig 4. Secured File Sharing

IV. CONCLUSION AND FUTURE ENHANCEMENT

1. Conclusion

In this paper, the notion of authorized data deduplication was proposed to protect the data security by including differential privileges of users in the duplicate check. We also presented several new deduplication constructions supporting authorized duplicate check in hybrid cloud architecture, in which the duplicate-check tokens of files are generated by the private cloud server with private keys. Security analysis demonstrates that our schemes are secure in terms ofinsider and outsider attacks specified in the proposed security model. As a proof of concept, we implemented a prototype of our proposed authorized duplicate check scheme and conduct testbed experiments on our prototype. We showed that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.

2. Future Enhancement

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 developed that the user can select the Cloud Server such as Google Drive, Hosting, Dropbox, Approx.He/She want.

REFERENCES

[1] B. Wang, B. Li, and H. Li, “Oruta: Privacy-preserving 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 identity-based 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 fine-grained 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, server-passive, 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].

[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.