Survey on Lightweight Secure Data Sharing Scheme for Mobile Cloud Computing

S. Vamsidhar[1], N.Samhitha [2], R. Yamini[3], M.Likhitha[4], S. Manasa sri[5], M.Nagaraju[6]
[1]*Asst.Prof, Dept of CSE, Tirumala Engineering College, Narasaropet, Guntur, A.P., India*
[2, 3,4,5,6] *B. Tech Students, Dept of CSE, Tirumala Engineering College, Narasaropet, Guntur, A.P., India*

Abstract: As even mobiles have a new scope to view and store personal data from all over the world, the growing use of cloud computing is feasible. This has triggered the data security problem over mobile computing due to increases in the data. There are lots of studies that used to protect the cloud, but most of them have mobile device limitations. The lower computational overhead should be given for the solution. In this paper the study for mobile cloud computing on Lightweight Secure Data Sharing Scheme (LDSS). It is necessary to use the resources provided by the cloud service provider (CSP) to store and share the data to deliver the satisfactory results.

Keywords: Cloud computing, Data Security, LDSS, CSP.

I. INTRODUCTION

The data sharing platform used for retroviral applications and data storage has strengthened with growing cloud storage and further smartphone use. Thanks to constraint of mobile data, cloud use has been increased significantly. The cloud has more room and infrastructure to store and share the data from the cloud provider. Mobile storage applications including images, pictures, documentation and other data can be exchanged with other people using these devices. The cloud service platform also offers control features, but personal information is confidential and should not be publicly shared. The privacy and data security which are the most important concern should be addressed. The cloud service provider's control mechanism is inadequate because it does not meet the needs of the data owner. The first and foremost issue is that when a user uploads the cloud information, the cloud service provider may spy on the file and the consumer will then use the code to decrypt the encrypted files. The data owner has to split the system consumer into different users based on who wants to share their password in the given group to overcome this problem. The handling of passwords is a major safety concern.

II. PROBLEM DEFINITION

A. Problem Statement

Secret key would protect the encrypted and decrypted files. The file should be shared by authorized users with privileges of access. The overhead of the traditional cryptographic algorithm should also be reduced and security systems with a lower overhead should be studied.

B. PROPOSED SYSTEM

The program is suggested as the mobile cloud computing framework with a lightweight data sharing system (LDSS). LDSS's major contributions are the following: it is used to effectively access cypher text by using the built LDSS-CP-ABE attribute-based crypting algorithm (AES). For encryption and decryption processes, the proxy servers are used here. ABE is used on proxy servers, which helps decrease overhead processing on the client side of mobile devices, for machine intensive operations. Data privacy through LDSS-CPABE is also retained. The changed decryption key edition is sent to the proxy servers in a secure way. The lazy encryption and decryption field of attributes is added to deal with the user cancelation issue. Finally, a system focused on LDSS for data sharing.

Advantages of proposed system:

LDSS has decreased database overhead owing to which increased application expenses have been minimized? The reliability of data sharing on mobile devices was improved with these strategies. In contrast to ABE leaves, LDSS has better results dependent on cipher text access control schemes. When several revocation procedures are merged together, the overhead is that. Overhead data storage is minimal.

III. LITERATURE SURVEY

The following literature analysis has been carried out in order to study and learn more about the lightweight secure data sharing system (LDSS).

In[1] the author introduced an efficacious fully homomorphic encryption system based on the LWE assumption. The worst

case has been used to address the safety problems of the short vector by the implementation of the tests observed of studying with errors. In this case, the improved results on the two previous works are firstly substituted by the homomorphic encryption focused on learning with error with a modern re linearisation strategy in the previous difficulty inference scheme relevant to dealings in various areas. Secondly, based on previous plays, the squashed model is adopted. Advanced modulus reduction methods are also essential to reduce the complexity of decryption.

The developers of[2] include prevention of data leakage in shared cloud access control. As the use of cloud computing has grown, frameworks used for Software as a Service (SaaS) become collective. There are more benefits of SaaS collaboration, but there are some safety issues. For enhanced SaaS communication, it is likely the knowledge could spill while the intruders are being exchanged. The idea of SaaS communication applications is to reduce data loss issues by reducing human errors. A set of methods can be used to minimize the knowledge leakage by encoding the organizations ' organizational safety rules which are mandatory to reach mutual judgments, prioritizing the possible recipients of user data to mitigate the mistake and scrutinize the odd recipients.

Within[3] the writers talk about the application of deniable mobile device data encryption. Data privacy is the most significant recognition and can be used by encryption. As security, the user information is included and the keys are used. The data were concealed to keep the intruders from viewing. Using Steganographic methods and unconstitutional coding algorithms to overcome specific problems. There are a few challenges that are plausibly deniable encryption (PDE) in mobile environments after assessment and finding something different. A new system named Mobiflage is intended to overcome these issues and PDE on mobile devices is used to conceal protected external storage volumes.

In[4] writers have access to outsourced data in a safe and effective manner. The main thing is that large-scale data are secure and available effectively. The new mechanism for solving the problem of read applications for owner-writers is introduced here. The robust, cryptographic access control is accomplished effectively with the use of encryption by using different keys for each data block. But the owner must have some information to use the main derivative methods. The use of Hash functions to extract the key to the study decreases the overhead for computation. The use of excessive encryption and/or lazy delete can also be used for entry to updated data files.

In[5] the writers introduce the basic feature of the input control in the cloud storage framework with effective cancellation. In this process, consumers can save data into the cloud and also guarantee the data is stored in the cloud. Since cloud servers and data owners are not within the same sphere of trust, the access policy can not be extended to a semiconfident cloud server. Traditional methods were used in order to solve problems when the data was secured and approved users obtained the decryption keys. The traditional method requires high overall control and is complex. To overcome these problems, the Cipher text-policy Encryption dependent attribute approach(CP-ABE) uses a new architecture for accessing the control mechanism for the Cloud storage systems.

IV. METHODOLOGY

Figure shows the LDSS system for lightweight data sharing in mobile cloud. The 4th of January. The device owner is used for transferring data to the internet cloud to sharing it with other users:

- The data owner (DO)-a data master. The access control procedures are decided by DO.
- Data User (DU): The Data User uses mobile storage to recover data.
- Trust Authority (TA): Trust Authority shall create and disperse the keys in the attribute.

As Fig reveals. 4.1, the Data Owner sends cloud-specific data. The data must be crypted before it is transmitted because the server is not sure. In order to determine the assignment of the data consumer to receive those data files through access control policies, the data owner accesses a control tree in the data files. The LDSS data files are symmetrically encrypted and the symmetric keys of the data encryption are carried out using the Dependent Enclyption Attribute (ABE). The symmetric key is stored in cipher code, the access control procedure. You will decode and recover the symmetrical key from your device owner with authority to enter the encrypted control policies.

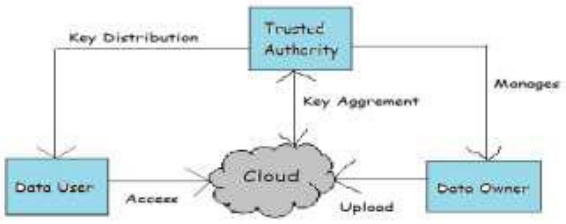

Fig -4.1: A lightweight data-sharing scheme (LDSS) framework

V. RESULT AND CONCLUSIONS

It uses the LDSS, the data protection for the mobile cloud and also reduces the overhead on the user's side of the mobile cloud. The data customer, data owner and trustee listing is given as shown in Fig 2. There are the data owner and device authentication calculations shown in Figure 3. The Home Page shown in Fig 4 is for the customer, data owner and secrecy. The following page pops up as shown in Fig 5 depending on the specific user. The file will be transferred to the data owner and sent for encryption in Figure 6.

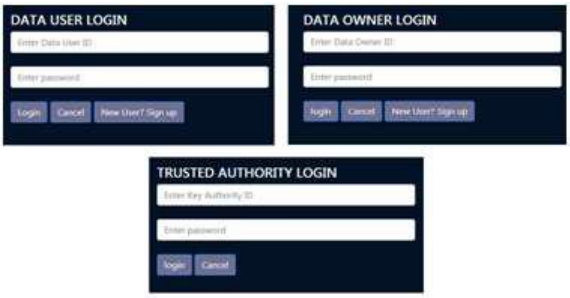

Fig -2: Login page for Data User, Data Owner and Trust Authority

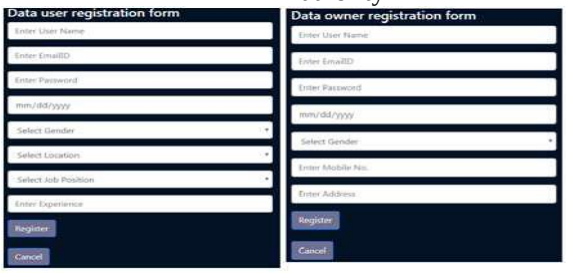

Fig -3: Signup Page for data user and data owner

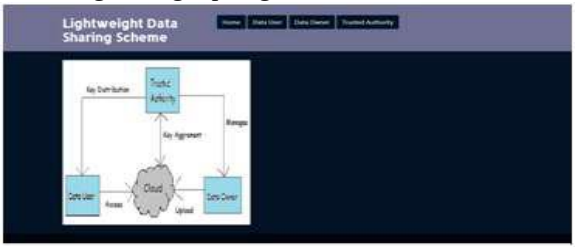

Fig -4: Home page

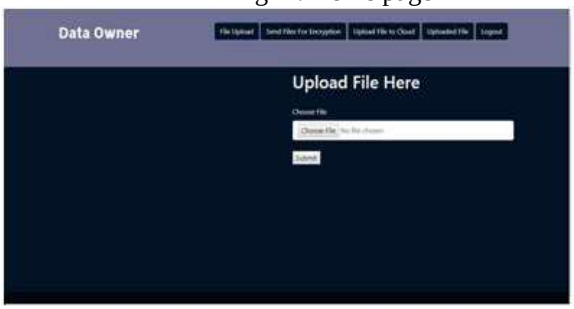

Fig -5: File uploads in Data owner

Fig -6: After clicking on send files for encryption in data owner page

VI. CONCLUSION

Most cloud access control experiments have been focused in recent years entirely on an attribute-based encryption algorithm (ABE). Nonetheless, conventional ABE is not sufficient for mobile cloud, since it is code intensive and the resources for mobile devices are restricted. We plan to address this problem in this paper LDSS. This introduces a new LDSS-CPABE algorithm to move large overhead computers from mobile devices into servers, thus helping to resolve the problem of secure data sharing in the mobile cloud. Results from the analysis demonstrate that LDSS provides data protection in the mobile cloud and through device overload. We must develop new strategies to maintaining data integrity in future work. To further leverage the capacity of mobile cloud to insure that you recover cipher text from existing systems for data sharing.

VII. REFERENCES

[1] Gentry C, Halevi S. Implementing gentry's fully homomorphic encryption scheme. in: Advances in Cryptology–EUROCRYPT 2011. Berlin, Heidelberg.

[2] Brakerski Z, Vaikuntanathan V. Efficient fully homomorphic encryption from (standard) LWE. in: Proceeding of IEEE Symposium on Foundations of Computer Science. California, USA.

[3] Qihua Wang, Hongxia Jin. "Data leakage mitigation for discertionary access control in collaboration clouds". the 16th ACM Symposium on Access Control Models and Technologies. [4] Adam Skillen and Mohammad Mannan. On Implementing Deniable Storage Encryption for Mobile Devices. the 20th Annual Network and Distributed System Security Symposium .

[5] Wang W, Li Z, Owens R, et al. Secure and efficient access to outsourced data. in: Proceedings of the 2009 ACM workshop on Cloud computing security. Chicago, USA.

[6] Maheshwari U, Vingralek R, Shapiro W. How to build a trusted database system on untrusted storage. in: Proceedings of the 4th conference on Symposium on Operating System Design & Implementation.

[7] Kan Yang, Xiaohua Jia, Kui Ren: Attributebased fine-grained access control with efficient revocation in cloud storage systems. [8] Crampton J, Martin K, Wild P. On key assignment for hierarchical access control. in: Computer Security Foundations Workshop.

[9] Shi E, Bethencourt J, Chan T H H, et al. Multidimensional range query over encrypted data. in: Proceedings of Symposium on Security and Privacy (SP).
[10] Cong Wang, Kui Ren, Shucheng Yu, and Karthik Mahendra Raje Urs. Achieving Usable and Privacy-assured Similarity Search over Outsourced Cloud Data.
[11] Yu S., Wang C., Ren K., Lou W. Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing.
[12] Kan Yang, Xiaohua Jia, Kui Ren, Bo Zhang, Ruitao Xie: DAC-MACS: Effective Data Access Control for Multiauthority Cloud Storage Systems. IEEE Transactions on Information Forensics and Security.
[13] Stehlé D, Steinfeld R. Faster fully homomorphic encryption. in: Proceedings of 16th International Conference on the Theory and Application of Cryptology and Information Security. Singapore.
[14] Junzuo Lai, Robert H. Deng ,Yingjiu Li ,et al. Fully secure key-policy attribute-based encryption with constant-size ciphertexts and fast decryption. In: Proceedings of the 9th ACM symposium on Information, Computer and Communications Security (ASIACCS).
[15] Bethencourt J, Sahai A, Waters B. Ciphertextpolicy attribute based encryption. in: Proceedings of the 2007 IEEE Symposium on Security and Privacy (SP). Washington, USA: IEEE Computer Society