A Protected And Lightweight Data Distribution Program For Mobile Cloud Computing

GUNDEBOINA NAGARAJU
M.Tech Student, Dept of CSE, Malla Reddy College of Engineering & Technology, Kompally, Hyderabad, T.S, India

G. RAVI
Associate Professor, Dept of CSE, Malla Reddy College of Engineering & Technology, Kompally, Hyderabad, T.S, India

Abstract: **Because of the widespread adoption of cloud computing, mobile devices may now store and access personal data from any location at any time. As a result, the data security issue in mobile cloud is becoming increasingly serious, impeding the growth of mobile cloud. There have been several researches undertaken in order to enhance cloud security. However, because mobile devices have limited processing capabilities and power, the majority of them are not suitable for mobile cloud. Mobile cloud applications require solutions with a low computational overhead. We propose a lightweight data sharing mechanism for mobile cloud computing in this work. It provides attribute description fields to achieve lazy-revocation, which is a difficult problem in CP-ABE systems based on programs. The experimental findings suggest that when users share data in mobile cloud settings, a lightweight data sharing technique may effectively minimize the overhead on the mobile device side.**

Keywords: **Access Control; Mobile Cloud Computing; Data Encryption; Mobile Cloud Computing;**

1. INTRODUCTION:

Data owners are driven to outsource their local complicated data management systems to the cloud because of its excellent flexibility and cost savings as the amount of data created by individuals and businesses that has to be stored and exploited grows fast [1]. However, because sensitive cloud data may need to be encrypted before being outsourced, making the typical data utilization service based on plaintext keyword search outdated, figuring out how to provide privacy-assured cloud data utilization mechanisms is critical. The problem is particularly tough to solve, given the enormous number of on-demand data users and vast quantity of outsourced data files in the cloud, since it is highly difficult to fulfill the practical criteria of performance, system usability, and high-level user searching experiences. The challenge of secure and efficient similarity search across outsourced cloud data is investigated in this study [2]. Similarity search is a basic and effective method for retrieving plaintext information, but it has yet to be fully explored in the encrypted data realm. First, we use a suppressing approach to generate a storage-efficient similarity keyword set from a given document collection, using edit distance as the similarity measure. We next construct a private tire-traverse searching index based on this, and demonstrate that it successfully provides the specified similarity search capabilities while maintaining a constant search time complexity. Under stringent security treatment, we officially verify the proposed mechanism's privacy-preserving promise. To prove the universality of our method and broaden the range of applications, we show that our novel design naturally allows fuzzy search, a previously researched concept that only tolerates typos and representation errors in the user searching input [3]. Extensive trials on Amazon's cloud platform with real data sets further illustrate the suggested mechanism's validity and applicability.

2. RELATED STUDY:

A cloud storage service enables data owners to offload their data to the cloud and offer users with access to it. The semi-trusted cloud server cannot be depended on to enforce the access policy since the cloud server and the data owner are not in the same trust domain. Traditional solutions for dealing with this problem often involve the data owner to encrypt the data and provide decryption keys to authorized users. These solutions, on the other hand, usually include complex key management and a significant level of overhead for the data owner. In this study, we propose an access control framework for cloud storage systems that uses a modified Cipher text-Policy Attribute-based Encryption (CP-ABE) technique to accomplish fine-grained access control. An efficient attribute revocation approach is proposed in the proposed strategy to deal with dynamic changes in users' access rights in large-scale systems [4]. In the random oracle model, the suggested access control method is both provably safe and efficient to use in reality, according to the study. Data access control is a good technique to keep your data safe in the cloud. However, due to data outsourcing and trusting cloud servers, data access control in cloud storage systems has become a difficult issue. Existing access control approaches are no longer suitable to cloud storage systems since they either generate numerous encrypted copies of the same data or necessitate the usage of a fully trusted cloud server [5]. Users may freely nominate a proxy, defined by qualities, who could re-encrypt a cipher text associated with one access policy to another with a different access policy. Without random

oracles, the suggested method is shown to be selective-structure selected plaintext secure and master key secure. In addition, in our scheme, we build a new type of key delegation capacity and examine certain relevant topics, such as a better security model and applications. Users may freely nominate a proxy, defined by qualities, who could re-encrypt a cipher text associated with one access policy to another with a different access policy [6]. Without random oracles, the suggested method is shown to be selective-structure selected plaintext secure and master key secure. In addition, in our scheme, we build a new type of key delegation capacity and examine certain relevant topics, such as a better security model and applications.

3. AN OVERVIEW OF EXISTING SYSTEM AND DISADVANTAGES:

In general, these systems may be divided into four categories: basic cipher text access control, hierarchical access control, completely homomorphism encryption access control, and attribute-based encryption access control (ABE). All of these suggestions are intended for use in a non-mobile cloud context. Tysowski et al. looked at a specific cloud computing environment in which data is accessed by resource-constrained mobile devices and proposed new ABE modifications that shifted the higher computational overhead of cryptographic operations to the cloud provider while lowering the total communication cost for the mobile user. Many data owners are concerned about the protection of personal sensitive data. The CSP's state-of-the-art privilege management and access control techniques are either insufficient or inconvenient. They are unable to address all of the needs of data owners. They use up a lot of storage and processing power, which isn't available on mobile devices [7]. The user privilege change problem is not properly addressed by current methods. A large revocation cost might come from such a procedure. This also does not apply to mobile devices. Clearly, there is no adequate answer to the problem of safe data exchange in the mobile cloud.

4. AN OVERVIEW OF PROPOSED SYSTEM AND ADVANTAGES:

For the mobile cloud computing context, we suggest a Lightweight Data Sharing Scheme. The following are the primary contributions of the Lightweight Data Sharing Scheme: To provide effective access control over encrypted text, we devised the Lightweight Data Sharing Scheme-CP-ABE algorithm, which is based on the Attribute-Based Encryption (ABE) approach. Encryption and decryption processes are handled by proxy servers. In our technique, ABE's computationally expensive operations are performed on proxy servers, reducing the computational burden on client-side mobile devices significantly. Meanwhile, in Lightweight Data Sharing Scheme-CP-ABE, a version property is introduced to the access structure to protect data privacy. The decryption key format is changed so that it may be securely transmitted to proxy servers. When dealing with the user revocation problem, we add sluggish re-encryption and an attribute description field to lower the revocation overhead. Finally, using the Lightweight Data Sharing Scheme, we build a data sharing prototype framework. The results demonstrate that using the Lightweight Data Sharing Scheme may significantly minimize client overhead while only adding a small amount of extra cost to the server. A realistic data sharing security method for mobile devices may be implemented using such an approach. In comparison to previous ABE-based access control methods over encrypted text, the results suggest that the Lightweight Data Sharing Scheme performs better. Multiple revocation operations are merged into one, reducing the overall overhead. In Lightweight Data Sharing Scheme, the storage overhead needed for access control is very small compared to data files.

5. SYSTEM IMPLEMENTATION:

People are progressively becoming acclimated to a new era of data sharing model in which data is kept on the cloud and mobile devices are used to store/retrieve data from the cloud, thanks to the rise of cloud computing and the popularity of smart mobile devices. Individuals (data owners) can use these programs to upload their papers and other things to the cloud and share them with other people (data users) they want to share them with. CSPs also give data owners the ability to govern their data. Due to the sensitivity of personal data files, data owners have the option of making their files public or just sharing them with selected data users. Data privacy of personal sensitive data is obviously a major concern for many data owners. When a data owner (DO) registers with TA, TA uses the Setup() procedure to create a public key PK and a master key MK. MK is retained on TA while PK is moved to DO. DO create its own set of qualities and assigns them to its contacts. All of this data will be delivered to TA as well as the cloud. The information is received and stored by the cloud. Because the cloud is untrustworthy, data must be encrypted before being uploaded. The DO assigns which characteristics a DU should receive if he wants to access a certain data file by defining access control policy in the form of an access control tree on data files. User of Data (DU): DU logs on to the system and sends a request for authorization to TA. The authorization request contains attribute keys (SK) that DU already possesses, so it accepts the request, validates it, and generates attribute keys (SK) for

DU. DU makes a data request to the cloud. When Cloud gets the request, it examines the DU to see whether it fits the access requirements. The cipher is delivered to DU. with the help of DSP; DU decrypts the symmetric key's cipher text. The symmetric key is used by DU to decode the encrypted text of data files. A trusted authority (TA) is used to make the Lightweight Data Sharing Scheme work in practice. It's in charge of creating public and private keys, as well as assigning attribute keys to users. Users can share and access data without being aware of the encryption and decryption activities using this approach. We assume that TA is completely trustworthy, and that a secure channel exists between the TA and each user. The existence of a trusted channel does not imply that the data can be exchanged across that channel, because the data might be huge. TA is only used to securely transfer tiny amounts of keys between users. Furthermore, TA must be available at all times since data users may access data at any time and require TA to update attribute keys. Provider of Cloud Services: The data for DO is stored in CSP. It faithfully implements the activities asked by DO, and it may look into data saved in the cloud by DO. It also makes a request for data to the cloud. When Cloud gets the request, it examines the DU to see whether it fits the access requirements. If DU is unable to satisfy the condition, it will reject the request; otherwise, the encrypted text will be sent to DU. the Uploaded Files are managed by CSP.

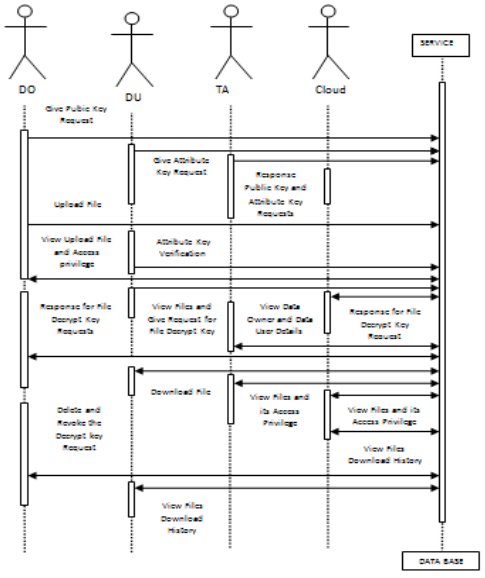
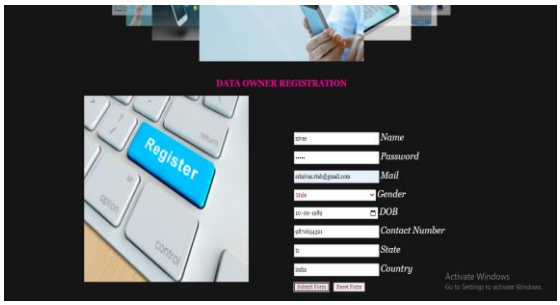


Fig.5.1. Application Work Flow.

RESULT SCREENS:

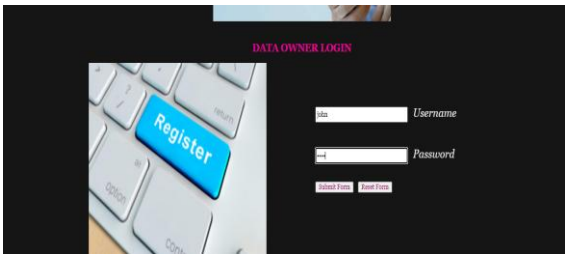
Home page:

Owner register:

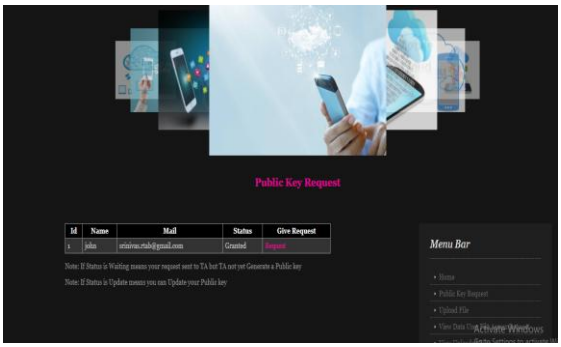


User register:

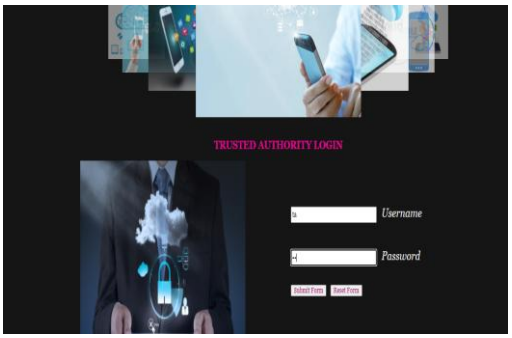
Owner login:



Public key request:



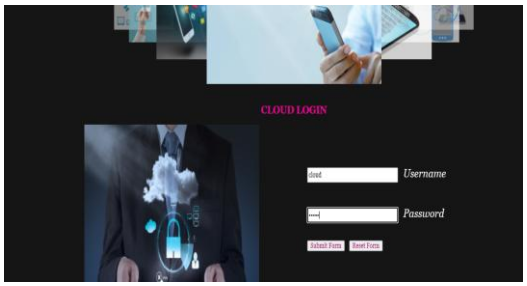
Ta login:



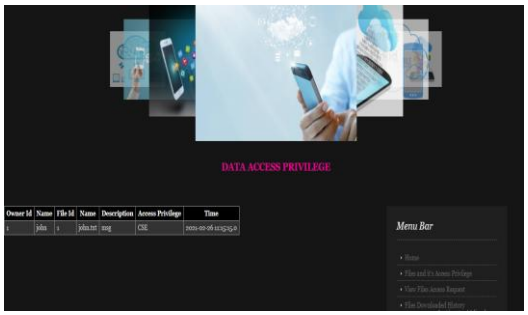
Upload:

Upload files:

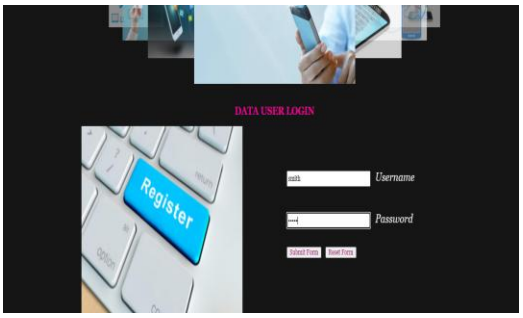
Cloud login:



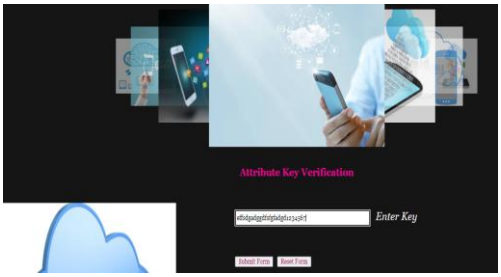
Files:



User login:



File verified:



4. CONCLUSION & FUTURE WORK:

Many researches on cloud access control have relied on attribute-based encryption algorithms in recent years (ABE). Traditional ABE, on the other hand, isn't appropriate for mobile cloud since it's computationally demanding, and mobile devices have limited capabilities. To solve this problem, we suggest the Lightweight Data Sharing Scheme. It presents the CP-ABE technique, which moves main computing cost from mobile devices to proxy servers, solving the safe data exchange problem in mobile cloud. The results of the experiments reveal that the Lightweight Data Sharing Scheme can secure data privacy in the mobile cloud while also reducing the overhead on the users' side. We will develop innovative techniques to assure data integrity in the future. We'll also look into ways to accomplish encrypted text retrieval via existing data sharing protocols to maximize the possibilities of mobile cloud.

REFERENCES

[1] Adam Skilled and Mohammad Manna. On Implementing Deniable Storage Encryption for Mobile Devices. the 20th Annual Network and Distributed System Security Symposium (NDSS), Feb. 2013.

[2] 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-Volume 4. USENIX Association, pp. 10-12, 2000.

[3] Cong Wang, Kui Ren, Shucheng Yu, and Karthik Mahendra Raje Urs. Achieving Usable and Privacy-assured Similarity Search over Outsourced Cloud Data. IEEE INFOCOM 2012, Orlando, Florida, March 25-30, 2012

[4] 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: ACM pp. 55-66, 2009.

[5] Kan Yang, Xiaohua Jia, Kui Ren, Ruitao Xie, Liusheng Huang: Enabling efficient access control with dynamic policy updating for big data in the cloud. INFOCOM 2014, pp.2013-2021, 2014.

[6] Jia W, Zhu H, Cao Z, et al. SDSM: a secure data service mechanism in mobile cloud computing. in: Proceedings of 30th IEEE International Conference on Computer Communications. Shanghai, China: IEEE, pp. 1060-1065, 2011.

[7] D. Huang, X. Zhang, M. Kang, and J. Luo. Mobicloud: A secure mobile cloud framework for pervasive mobile computing and communication. in: Proceedings of 5th IEEE International Symposium on Service-Oriented System Engineering. Nanjing, China: IEEE, pp. 90-98, 2010.

