A Survey on Variant Approaches to Secure Cloud Storage System Data

Sajida Tabasum U[1],
M-Tech CNE Student, Dept.of CSE, RITM, Bengaluru.

Mrs. K Anitha[2]
Assistant Professor, *Dept.of CSE, RITM, Bengaluru*

ABSTRACT: In the field of information technology, a revolution has been created by the cloud computing which purviews the elements from grid computing, utility computing and autonomic computing to obtain innovative deployment architecture. Cloud computing endeavors smooth access to the cloud data. High performance computing is achieved by storing the data on cloud because the cloud data can be accessed at any time from any place as long as there is network access, this rapid movement towards the cloud has created a major impact on the level of security provided to the cloud storage system data. In this paper a survey on existing approaches to provide security to cloud data is explored and a two-factor data security protection mechanism with factor revocability for cloud storage system has been proposed.

INDEX TERMS: Cloud, Internet, Malicious, Cloud service provider, KGC, Cryptography, Identity based encryption, Private Key Generator, Revocation.

I. INTRODUCTION

Cloud storage system is a network storage system in which the data is stored on the remote server and can be accessed through the internet. Cloud storage system is maintained, operated and managed by a cloud storage service provider. There are many benefits of using cloud storage, the most notable benefits data accessibility is made simple, data sharing between users is easy and fast, no risk of maintenance and also no limitation of storage space capacity.

Even though there are many advantages of cloud storage, there is also risk involved in outsourcing the data, which increases the attack surface area because, when data is distributed, the more locations it is stored the higher risk it contains for unauthorized physical access to the data. When networks and storage is shared among multiple users, the possibilities of unauthorized users to access the cloud data increases which may be due to faulty equipment, mistaken actions or due to the criminal intent.

To provide security to the data stored on the cloud, encryption techniques are used. Encryption is also used to protect the data which is being transmitted to and from the cloud service and data that is stored at the cloud service provider. If the unauthorized adversary gains access to the cloud but cannot get any information about the plaintext message, because the data is encrypted. A two-factor data security protection mechanism with factor revocability for cloud storage system has been proposed in this paper which is based on identity-based mechanism in which the sender needs to know only the identity of the receiver to encrypt the data.

II. LITERATURE SURVEY

Virtual Machine concept was created around 1970. In virtualization one or more operating systems run simultaneously under the presence of one physical hardware in the system. VM operating system uplifted the concept of shared access mainframe to the above level and virtualization became popular during this period.

Cloud computing Evolution has been briefly explained by the following concepts.

- Grid computing: Large problems can be solved with parallel computing.
- Utility computing: Resources were offered as a metered service.
- Cloud computing: IT resources are provided to the users as a service in a dynamic way anytime and anywhere.

Internet had roots in 1960s, but became useful during 1990s. In 1991 the World Wide Web was born and in 1993 a web browser was released. Internet became more and more fast and reliable. Hardware and software were purchased by Application Service Provider (ASP) and sold it for their customers on a monthly basis. In the late 1990s cloud computing started to appear.

From past few years, the cloud storage management has grown compellingly and about 47 billion dollar has been spent in the year 2013 on cloud services. It is also expected that over 108 billion dollar will be spent in the year 2017. Tremendous growth has been seen in the cloud computing industry. Cloud computing also faces same threats as faced by the

physical workspace. Many tactics are used by the attackers to attack the cloud by taking advantage of email to pass virus. Cloud data can be secured by maintaining the ownership of data. Guidelines and frameworks have been developed by the Cloud Security Alliance (CSA),a non-profit organization for implementing and enforcing a security in the cloud environment.

In [1], cloud storage systems allow users to store remotely their data and enjoy the on-demand high quality cloud applications and does not impose any burden of local hardware and software management. Though the benefits are clear in this service, it poses new security risks towards the correctness of the data in cloud because the data is outsourced. To address this new problem and also to achieve a secure and dependable cloud storage service, a mechanism is proposed in this paper which is a flexible distributed storage integrity auditing mechanism, which utilizes the homomorphic token and distributed erasure-coded data. The proposed system design allows users to audit the cloud storage with very lightweight communication and computation cost. The auditing results guarantees strong cloud storage correctness and also achieves fast data error localization that is the identification of misbehaving server. By considering the dynamic nature of cloud data, the proposed design further supports secure and efficient dynamic operations on outsourced data, which includes block modification, deletion, and append.

To ensure the security and dependability for cloud data storage, the following goals has to be achieved.

- **Storage correctness**: To ensure users that their data are indeed stored appropriately and kept intact all the time in the cloud.
- **Fast localization of data error:** To effectively locate the malfunctioning server when data corruption has been detected.
- **Dynamic data support:** to maintain the same level of storage correctness assurance even if users modify, delete or append their data files in the cloud.
- **Dependability**: To enhance data availability against Byzantine failures, malicious data modification and server colluding attacks, that is minimizing the effect brought by data errors or server failures.
- **Lightweight:** to enable users to perform storage correctness checks with minimum overhead.

Below Fig.1 illustrates a network architecture for cloud storage service architecture.

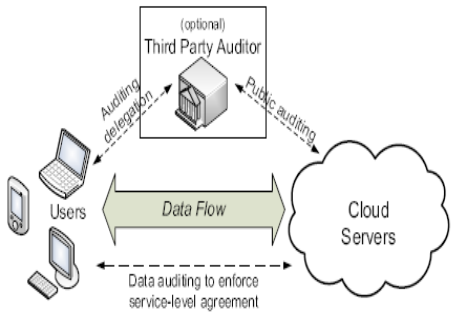

Fig. 1: Cloud storage service architecture

In cloud data storage, user stores his data through a CSP (cloud service provider) on to the cloud, which are running in a simultaneous, cooperated and distributed manner. The user interacts with the cloud servers via CSP to access or retrieve his data according to applications requirement .Depending upon some scenarios, the user performs block level operations on his data. Since on the cloud the data is not stored locally, it is very important to make sure that the user's data is stored and maintained correctly. There should exist some continuous security means through which users can make assurance about the data correctness of their data which is stored on cloud without the local copies existence. Suppose the users are not having sufficient time or resources to monitor their data online, then users can delegate the tasks of data auditing to trusted TPA. In this paper an assumption is made that the point-to-point communication channels between each cloud server and the user is authenticated and reliable, which can be achieved in practice with little overhead.

In [2], Nowadays cloud computing is having a major impact on academic and business environments because they are providing a dynamic and flexible infrastructure for them both. Data is moved to public cloud server (PCS) in a public cloud and the user will not have any control on remote data, in this situation the problem of information security in public cloud storage arises such as data confidentiality, integrity, and availability. In this paper, a study proxy provable data possession (PPDP) is conducted because in public clouds, PPDP is a matter of crucial importance

when the client cannot perform the remote data possession checking. An efficient PPDP protocol has been designed based on bilinear pairing technique.

Below Fig. 2 shows the PPDP system model. The massive data is moved to remote PCS in the cloud environment and the clients will be relieved from the storage and computation burden. The clients have to ensure that their data is correctly stored on the cloud and securely maintained. There should exist some security mechanisms which must give provision to the clients to periodically check their remote data integrity because their does not exist local copies. To perform the checking of remote data possession, the Client has to delegate this task to its proxy by the warrant ω. The *Client*'s remote data possession is checked by proxy, when it satisfies the conditions of the warrant ω. Whether an outsourced storage site retains a file which consists of a collection of n blocks is been checked by a PPDP protocol. The client who is a data owner pre-processes the file and generates a piece of metadata. Then, the client transmits the file and its metadata to the server PCS, and deletes its local copy. PCS stores the file and responds to the challenges issued by the *Proxy*.

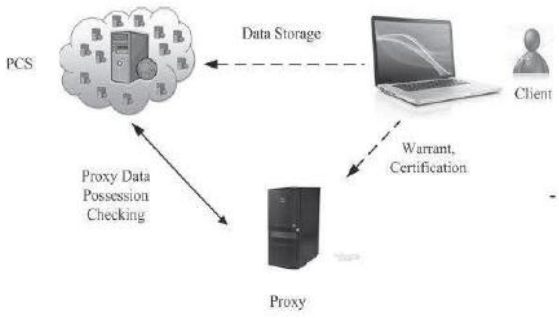

Fig. 2 PPDP System model

In [3], Usually an inherent key security issue exists in the identity-based cryptosystems because the Key Generation Center (KGC) always knows the user's secret key. If the KGC itself is malicious then it can always impersonate the user. Certificateless cryptography can solve the above briefed problem. Always an assumption is made that always certificateless schemes are malicious and KGC starts launching attacks such as Type II attacks, only after it has generated a master public/secret key pair correctly. In this paper, new security model are proposed which removes this assumption for both certificateless signature and encryption schemes.

The certificateless signature and certificateless encryption schemes share the same set of key generation algorithms. The key generation algorithms are briefed below. The certificateless cryptosystem has three key generation algorithms such as MasterKeyGen, PartialKeyGen, UserKeyGen.

1. MasterKeyGen (Master Key Generation): On input 1^k where $k \in \mathbb{N}$ is a security parameter, it generates a master public/secret key pair (mpk, msk). Let $\mathsf{MPK}(k)$ be set of all possible master public keys generated by $\mathsf{MasterKeyGen}(1^k)$. Without loss of generality, we assume that it is computable to determine if a master public key mpk is in $\mathsf{MPK}(k)$.
2. PartialKeyGen (User Partial Key Generation): On input msk and user identity $ID \in \{0,1\}^*$, it generates a user partial key $partial\_key$.
3. UserKeyGen (User Key Generation): On input mpk and user identity ID, it generates a user public/secret key pair (upk, usk).

In [4], Though there exists many alternatives for cloud data storage, but there is yet immaturity present in providing data confidentiality solutions for the database. In this paper a novel architecture is proposed which integrates cloud database services with data confidentiality and the possibility of executing concurrent operations on encrypted data. This solution helps in establishing the connection between the geographically distributed clients to connect themselves directly to an encrypted cloud database, and they can execute concurrent and independent operations which includes modifying the database structure. The proposed architecture eliminates intermediate proxies which limit the elasticity, availability, and scalability properties that are intrinsic in cloud.

The proposed architecture does not require modifications to the cloud database. The proposed system guarantees data confidentiality because it allows a cloud database server to execute concurrent SQL operations such as read or write and also modifications to the database structure over encrypted data. Below fig.3 shows the proposed system architecture.

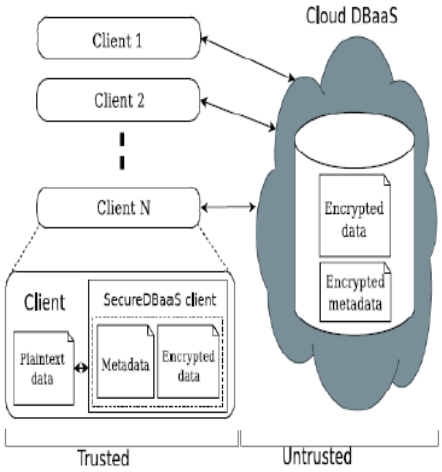

Fig.3 Proposed System Architecture

In [5], cloud computing has become an important need of computing today which has made a significant impact on the IT industry. Identity Based Encryption (IBE) is a mechanism which is a substitute for public key encryption. The limitation aspect of IBE is the private key computation during the user revocation. The aim in this proposed system is to handle the concerns of identity revocation. The Present scheme offloads the majority of the key generation related operations during key issuing and key update processes to a Key Update Cloud Service Provider, leaving only a constant numeral of simple operations for PKG and users to carry out locally. This objective is achieved by utilizing a novel collusion resistant technique that has employed a hybrid private key for each user, in which an AND gate is concerned to connect and leap the identity component and the time component. Below Fig.4 shows basic view of cloud computing.

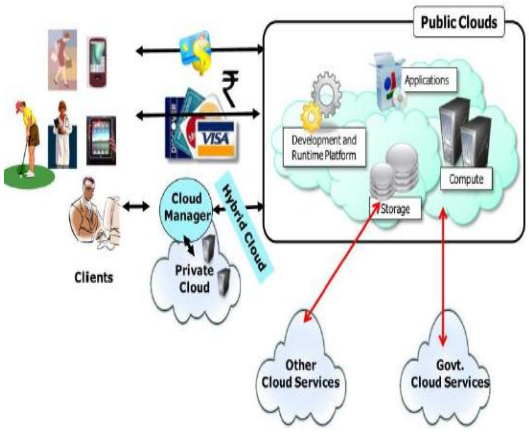

Fig.4 Basic View of Cloud Computing

Below Fig.5 shows the System Model for IBE with outsourced Revocation

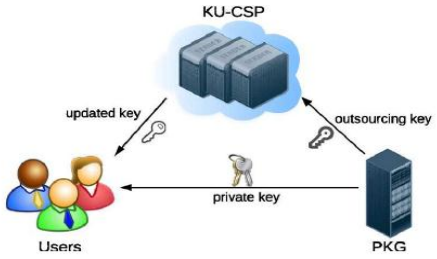

Fig. 5 System Model for IBE with Outsourced Revocation.

III. SCOPE OF PROJECT

The scope of the project includes the two level securities to the cloud services along with:

- Security against hacker by disabling decrypts the data unless he possesses the USB device.
- If the USB device is lost, then the receiver can report the USB device loss information.
- The lost device becomes an old device. The old device is revoked.
- The receiver will be given a new device.

Again same process is carried out to encrypt the data.

IV. PROPOSED APPROACH

Problem Statement: The project "Two Factor Data Security Protection Mechanism for Cloud Storage System" mainly deals with not using the public key. The project considers only using Identity-Based Encryption of the receiver.

Description: The proposed project intends to enable a System with IBE(Identity Based Encryption) mechanism in which the sender need to know only the identity of the user to send encrypted data and no other information of the receiver such as public key or certificate is required. Then the sender sends the ciphertext to the cloud where the receiver can download it at anytime.The proposed system provides two-factor data encryption protection.To decrypt the data stored in the cloud, receiver need to do two things, firstly secret key which is stored in the computer should be provided. Secondly receiver should have unique security device which will be connected to computer such as USB. It becomes impossible to decrypt the text without either of these things that is either secret key or security device.

The proposed system for the first time provides security device (one of the factors) revocability. Once the security device is stolen or reported as lost, this

device is revoked. The cloud will immediately execute some algorithms to change the existing cipher text to be un-decryptable by this device. While the user needs to use his new replacement device together with his secret key to decrypt the cipher text. This process is completely transparent to the sender. The cloud server cannot decrypt any cipher text anytime.

Methodology: Below Fig.6 shows the proposed system mechanism. In this approach, even if the security device is lost, new device can be issued from the SDI(Security Device Issuer). After getting the new USB device, the old device is revoked by taking the device id from the database and copying or writing the same device id to the new device. In this way new device can be used as the revocable option.

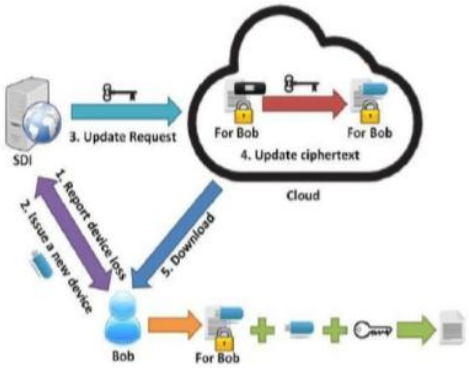

Fig.6 Proposed System Mechanism

First, device loss is reported to SDI then SDI issues a new device and update request is made. Cipher text is updated in the cloud using a new device. Then the file can be downloaded and decrypted twice using new device and secret key of the receiver. The system is based on IBE(Identity Based Encryption) mechanism. Sender or creator of the cipher text needs to know only the email address of the receiver but nothing else related to the receiver. Then the sender sends the file containing the cipher text to the cloud to let the receiver download the file.

The receiver has an email address. Using this email address as the unique identity the cipher text file stored in the cloud server is downloaded for decryption. Receiver first downloads the file to local machine and starts applying the decryption process. Receiver has a private key or secret key stored in his machine and a security device which contains an identity of the receiver. The decryption process of the cipher text requires both security device and a private key or secret key.

The proposed project methodology has Setup Phase, in which Master secret key is generated. The Master key is generated by Private Key Generator (PKG). Then there is a Key and Device Issued Phase. Security Device Issuer (SDI) will generate a security device. There are other phases like First-Level Cipher text Generation phase, Second-Level Cipher text Generation Phase, Device Updated Phase.

The control flow and the interaction among the modules and the sub modules in the proposed system is shown in the Structure chart shown in below Fig.7.

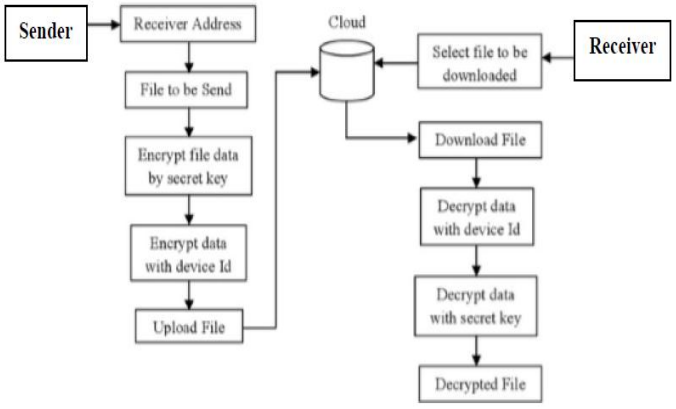

Fig.7 Structure chart of Data Security Protection Mechanism

VII. CONCLUSION

From the literature survey, we understood that the current research made to propose the various approaches for providing security to cloud data storage system has certain limitations. In the proposed approach, a novel attempt has been made to introduce a two factor data security protection mechanism for cloud storage system, in which sender needs to know only the identity of the receiver ,using this identity, sender encrypts the data twice and uploads the data to cloud server. The receiver then uses USB security device issued by security device issuer and secret key to decrypt the downloaded data. The proposed approach also supports the revocability of the device. If the USB security device is lost, then it can be replaced with the new USB security device.

VIII. ACKNOWLEDGEMENTS

I am grateful to my institution, REVA Institute of Technology and Management, for having provided me with the facilities for successfully completing this work “A Survey on Variant Approaches to Secure Cloud Storage System Data” and providing me all the necessary facilities for successful completion of this

paper. I take this opportunity to express my deep sense of gratitude to my guide Mrs. K Anitha for her valuable guidance. I also thank our university's management team for their continued support. Finally, I thank my family and friends for their motivation, moral, and material support.

References

[1] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou. Toward secure and dependable storage services in cloud computing. IEEE T. Services Computing, 5(2):220–232, 2012.

[2] H. Wang. Proxy provable data possession in public clouds. IEEE T. Services Computing, 6(4):551–559, 2013.

[3] M. H. Au, Y. Mu, J. Chen, D. S. Wong, J. K. Liu, and G. Yang. Malicious kgc attacks in certificateless cryptography. In ASIACCS, pages 302–311. ACM, 2007.

[4] L. Ferretti, M. Colajanni, and M. Marchetti. Distributed, concurrent, and independent access to encrypted cloud databases. IEEE Trans. Parallel Distrib. Syst., 25(2):437–446, 2014.

[5] C. Gentry. Certificate-based encryption and the certificate revocation problem. In EUROCRYPT, volume 2656 of Lecture Notes in Computer Science, pages 272–293. Springer, 2003.

[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, D. Song, "Provable Data Possession at Untrusted Stores," *Proc. 14th ACM Conference on Computer and Communications Security (CCS'07)*, pp. 598-609,2007.

[7] S. S. Al-Riyami and K. G. Paterson. Certificatelesspublic key cryptography. In *Proc. ASIACRYPT 2003*,pages 452–473. Springer-Verlag, 2003. LNCS 2894.

[8] F. Elwailly, C. Gentry, and Z. Ramzan, "Quasimodo: Efficient certificate validation and revocation," in Public Key Cryptography(PKC"04), F. Bao, R. Deng, and J. Zhou, Eds. Berlin, Germany: Springer, 2004, vol. 2947, pp. 375–388.