Decentralized erasure code assisted cloud with smartcard implementation

P.Yogalakshmi, S.P.Maniraj

Abstract— Cloud Computing is becoming next stage platform in the evolution of the internet. It provides the customer an enhanced and efficient way to store data in the cloud with different range of capabilities and applications. The data in the cloud is stored by the service provider. To ensure security and to prevent the data from discloser by unauthorized users, encryption of the data at the local device, e.g. desktop, tablet, or smartphone, prior to the data transfer to the remote cloud-based storage is the only possible solution. But if a cloud system is performing a task of storage of data and encryption and decryption of data on the same cloud then there are much more chances of getting access to the confidential data without authorization. This increases the risk factor in terms of security and privacy. This paper describes an enhancement for the already existing data security model in cloud environment. The proposed data security model provides user authentication and data protection using digital signature. It also uses locking protocol in cloud data storage with the help of CSP for data update. The main technical contribution is that the proxy re-encryption scheme supports encoding operations over encrypted messages as well as forwarding operations over encoded and encrypted messages. Our method fully integrates encrypting, encoding, and forwarding. We analyze and suggest suitable parameters for the number of copies of a message dispatched to storage servers and the number of storage servers queried by a key server. These parameters allow more flexible adjustment between the number of storage servers and robustness.

Index Terms- cloud storage system, cloud service provider, decentralized erasure code, digital signature, storage and key servers, proxy re-encryption, smartcard.

1 INTRODUCTION

In recent years, cloud computing becomes one of the new big shining stars in the global technology in industry. Cloud computing is simply an internet based computing. Cloud Computing is also described as "on-demand computing" because the user can access as per their requirement and demand (e.g. networks, storage, applications, servers, and services).Some of the major firms like Amazon, Microsoft and Google have implemented the "CLOUD" and have been using it to speed up their business. Previously, before the development of the concept of cloud computing, critical industrial data was stored on the storage media. This data was protected by firewalls to prevent getting disclosure of the confidential data externally and with the help of the organizational regulations it's possible to prevent the internal unauthorized access. Whereas, in the cloud computing, storage service provider must provide data security from getting prevented by unauthorized access. User confidential data is not secured and safe in this fast developing of distributed computing technologies.

- *P.Yogalakshmi is currently pursuing master degree program in Computer science and engineering in SRM University, India, PH-8870003806. E-mail: yogapunch87@gmail.com*
- *S.P.Maniraj is currently working as Assistant professor (O.G) in Computer science and engineering Department, SRM University, India, PH-9445384676. E-mail: spmaniraj@gmail.com*

As there are much more changes of getting data hacked by any unauthorized user. Encryption is effective but isn't a cure-all. The cloud provider should provide evidence that encryption schemes were designed and tested by experienced specialists. Data storage in the cloud is a process where the owner stores his data, files and applications through a cloud storage provider (csp) into a set of cloud servers. In order to ensure the integrity and availability of data in Cloud and enforce the quality of cloud storage service, efficient methods that enable on-demand data correctness verification on behalf of cloud users have to be designed. Even if the cloud provider encrypts the data and stores it in encrypted format, the provider is always in possession of the decryption key. The current Cloud security model is based on the assumption that the user/customer should trust the provider. Cloud Computing services are delivered through software as service (SaaS), platform as Service (PaaS), and Infrastructure as Service (IaaS).

To achieve adequate security the five goals should be achieved namely availability, confidentiality, data integrity, control and audit. Few cloud computing can achieve the five goal together now a days.

In this paper, we focus on designing a cloud storage system for robustness, confidentiality, and integrity. Data robustness is a major requirement for storage systems. One way to provide data robustness is to replicate a message such that each storage server stores a copy of the message. It is very robust because the message can be retrieved as long as one storage server survives. Another way is to encode a message of k symbols into a code word of n symbols by erasure coding. To store a message, each of its

code word symbols is stored in a different storage server. A storage server failure corresponds to an erasure error of the code word symbol. A decentralized erasure code is an erasure code that independently computes each code word symbol for a message.

2 Related Works

We briefly review distributed storage systems, proxy re encryption schemes, smartcard authentication, and decentralized erasure code.

2.1 Distributed Storage Systems

At the early years, the Network-Attached Storage (NAS) and the Network File System (NFS) provide extra storage devices over the network such that a user can access the storage devices via network connection. Lin and Tzeng [1] addressed robustness and confidentiality issues by presenting a secure decentralized erasure code for the networked storage system. A decentralized architecture for storage systems offers good scalability, because a storage server can join or leave without control of a central authority.

2.2 Proxy Re-Encryption Schemes

Proxy re-encryption schemes are proposed by Mambo and Okamoto and Blaze etal[11]. In a proxy re-encryption scheme, a proxy server can transfer a cipher text under a public key of user A to a new one under another public key of user B by using the re-encryption key RKA! B. The server does not know the plaintext during transformation. The cloud storage server uses a re-encryption algorithm to transfer the cipher text into the format that can be decrypted by the recipient's private key. That is the re-encryption key is generated from the data owner's private key and a recipient's public key. Ateniese etal[12] proposed some proxy re-encryption schemes and applied them to the sharing function of secure storage systems. In their work, messages are first encrypted by the owner and then stored in a storage server.

2.3 Decentralized Erasure Code

A decentralized erasure code is a random linear code with a sparse generator matrix. The generator matrix G is constructed by an encoder is as follows: First, for each row, the encoder randomly marks an entry as 1 and repeats this process for an in k/k times with replacement. Second, the encoder randomly sets a value from IF for each marked entry. This finishes the encoding process. A decoding is successful if and only if k × *k* sub matrix formed by the k-chosen columns is invertible. Thus, the probability of a success decoding is the probability of the chosen sub matrix being invertible. The owner randomly selects v servers with replacement and sends a copy of M to each of them. Each server randomly selects a coefficient for each received cipher text and performs a linear combination of all received cipher texts. Those coefficients chosen by a server form a column of the matrix and the result of the linear combination is a code word element. Each server can perform the computation independently. This makes the code decentralized.

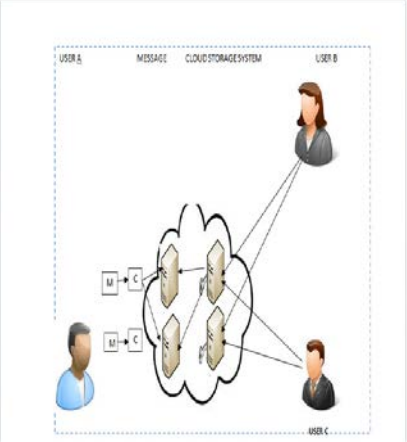

Figure 1 General system model of our work

We consider the system model that consists of distributed storage servers and key servers. Since storing cryptographic keys in a single device is risky, a user distributes his cryptographic key to key servers that shall perform cryptographic functions on behalf of the user. These key servers are highly protected by security mechanisms like IP sec. To well fit the distributed structure of systems, we require that servers independently perform all operations. With this consideration, we propose a new threshold proxy re-encryption scheme and integrate it with a secure decentralized code to form a secure distributed storage system. The multiplicative homomorphic encryption scheme supports encoding operations over encrypted messages and forwarding operations over encrypted and encoded messages. We convert a proxy re-encryption scheme with multiplicative homomorphic property into a threshold version. A secret key is shared to key servers with a threshold value t. To decrypt for a set of k message symbols, each key server independently queries 2 storage servers and partially decrypts two encrypted code word symbols. As long as t key servers are available, code word symbols are obtained from the partially decrypted cipher texts. The tight integration of encoding, encryption, and forwarding makes the storage system efficiently meet the requirements of data robustness, data confidentiality, and data integrity. Accomplishing the integration with consideration of a distributed structure is challenging. Our system meets the requirements that storage servers

independently perform encoding and re-encryption and key servers independently perform partial decryption.

2.4 Smart Card Authentication

Smart cards are credit card-sized devices that hold a small computer chip, which is used to store public and private keys and other personal information used to identify a person and authenticate him or her to the system. Logging onto the network with a smart card requires that you physically insert the card into (or slide it through) a reader and then enter a Personal Identification Number (PIN) in much the same way that you use an ATM card to access an automatic teller machine. Smart cards use cryptography-based authentication and provide stronger security than a password because in order to gain access, the user must be in physical possession of the card and must know the PIN.
Here we are implementing the same for login authentication using smart card device only for authorized users. The message of the sender along with private key taken from smart card cryptographic service provider and then forwarding to receiver using web service. The encrypted message stored in cloud storage provider, proxy server and key server for later decryption process

Important functionalities of smartcard

(1) Cloud user identification and authentication
(2) Generation of qualified electronic signatures and
(3) Data encryption and decryption.

3 Our Contribution

Assume that there are n distributed storage servers and m key servers in the cloud storage system. A message is divided into k blocks and represented as a vector of k symbols. Our contributions are as follows:

1. We construct a secure cloud storage system that supports the function of secure data forwarding by using a threshold proxy re-encryption scheme. Our system supports encrypting a text message using secure algorithm and then stored in cloud storage server and n distributed server and m key server with independent private key as well as public key.
2. Cloud storage provider performs independently with proxy server, database server and key server with the help of web service combined with private/public key pair for message encryption and decryption provided by webserver.

3. Our system is highly distributed where storage Servers independently encode and forward messages and key servers independently perform partial decryption.

4 Architecture and Implementation

In this section we explain the architecture and implementation of our smart card-based approach for storing data securely and confidentially in the public cloud.

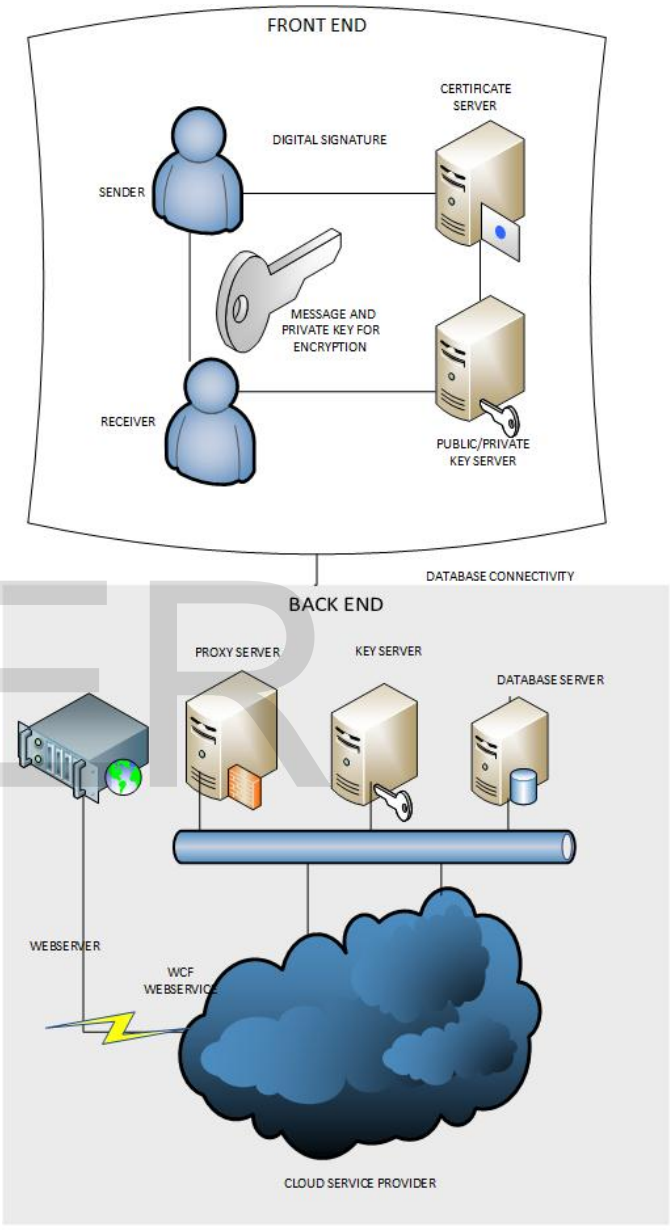

Figure 2: Architecture for securely storing data in the public cloud using the smartcard

Fig. 2 illustrates our architecture for secure encryption and decryption of data by using the smartcard functionality and storing the encrypted data in the public cloud. In this architecture, in fact three different entities are involved: (1) the cloud user who wants to store some file or directory securely in the public cloud, (2) to those the files or directory should be encrypted for and, (3) the public cloud

provider where the encrypted files will be stored.

4.1 Implementation

To implement our design, we need to achieve some goals in our model by allowing the smartcard to verify the correctness over the cloud data. Additionally, we need to ensure that the cloud server manipulate or alter the user data in the cloud .The proposed model is achieved using the digital signature technique. The digital signature works by taking the user data first, then perform a hash function over it using Secure Hash Algorithm (SHA) using sha2 technique. After that, computes the signature for the generated hash value by encrypting it with the private key taken from smart card provider. In the other side, the decryption is done by the public key but the result will be a hash value, and the hash value is not reversible to its original data. There are three procedures in our model to satisfy the integrity concept:

1. Digital signature part will be done by the user.

2. The CS verifies over the user data in the cloud to check over the manipulation or intrusions in the cloud data.

3. The smart card authentication verifies over the cloud server part to check if the cloud server was manipulating in the user data or not.

5 Conclusion

The study of existing system has revealed the use of centralized server, micro bench mark and Third Party Auditor (TPA). The implementations of the traditional systems have resulted in crashes, DOS attacks and unavailability due to regional network outages. In this paper, we studied the problems of data security in cloud data storage, which is essentially a distributed storage system. Thus our main idea is to give protection to the cloud storage area with strong trustworthiness so that user can feel free of worry for his uploaded data in his allocated space.

6 Future Work

As this is a modern world demand for cloud storage system reaches to peak. There are several directions which could be taken for further development of our work. Currently, there are two versions of the software for iOS and Android in development. We plan to integrate them with the desktop solution presented in this work. Another possible task for the future is the integration of the tool into the web browser.

7 References

[1] Hsiao-Ying Lin, Member, IEEE, and Wen-Guey Tzeng, Member" A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding" IEEE Transactions on Parallel and Distributed Systems, VOL. 23, NO. X, XXX 2012.

[2] J. Kubiatowicz, D. Bindel, Y. Chen, P. Eaton, D. Geels, R. Gummadi, S. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Zhao, "Oceanstore: An Architecture for Global-Scale Persistent Storage," Proc. Ninth Int'l Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS), pp. 190- 201, 2000

[3] P. Druschel and A. Rowstron, "PAST: A Large-Scale, Persistent Peer-to-Peer Storage Utility," Proc. Eighth Workshop Hot Topics in Operating System (HotOS VIII), pp. 75-80, 2001.

[4] A. Adya, W.J. Bolosky, M. Castro, G. Cermak, R. Chaiken, J.R. Douceur, J. Howell, J.R. Lorch, M. Theimer, and R. Wattenhofer, "Farsite: Federated, Available, and Reliable Storage for an Incompletely Trusted Environment," Proc. Fifth Symp. Operating System Design and Implementation (OSDI), pp. 1-14, 2002.

[5] A. Haeberlen, A. Mislove, and P. Druschel, "Glacier: Highly Durable, Decentralized Storage Despite Massive Correlated Failures," Proc. Second Symp. Networked Systems Design and Implementation (NSDI), pp. 143-158, 2005.

[6] Z. Wilcox-O'Hearn and B. Warner, "Tahoe: The Least-Authority Filesystem," Proc. Fourth ACM Int'l Workshop Storage Security and Survivability (StorageSS), pp. 21-26, 2008.

[7] H.-Y. Lin and W.-G. Tzeng, "A Secure Decentralized Erasure Code for Distributed Network Storage," IEEE Trans. Parallel and Distributed Systems, vol. 21, no. 11, pp. 1586-1594, Nov. 2010.H. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1985, ch. 4.

[8]. C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring data storage security in cloud computing," in Proc. of IWQoS'09, July 2009, pp. 1–9.

[9]. Cong Wang, Qian Wang, Kui Ren, Wenjing Lou, "Towards Secure and Dependable Storage Services in Cloud Computing," IEEE transactions on Services Computing, 06 May 2011.

[10].A.G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Decentralized Erasure Codes for Distributed Networked Storage," IEEE Trans. Information Theory, vol. 52, no. 6 pp. 2809-2816, June 2006.

[11]. M. Mambo and E. Okamoto, "Proxy Cryptosystems: Delegation of the Power to Decrypt Cipher texts," IEICE Trans. Fundamentals of Electronics, Comm. and Computer Sciences, vol. E80-A, no. 1, pp. 54-63, 1997.

[12]. G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," ACM Trans. Information and System Security, vol. 9, no. 1, pp. 1-30, 2006.