Novel Approach for Data Integrity Maintenance with User Revocation System

G.Kiruthika
Assistant Professor/CSE, Akshaya College of Engineering and Technology, Coimbatore, India.

Abstract – With data storage and sharing services in the cloud, users can easily modify and share data as a group. To ensure share data integrity can be verified publicly, users in the group need to compute signatures on all the blocks in shared data. Different blocks in shared data are generally signed by different users due to data modifications performed by different users. For security reasons, once a user is revoked from the group, the blocks which were previously signed by this revoked user must be re-signed by an existing user. The straight forward method, which allows an existing user to download the corresponding part of shared data and re-sign it during user revocation, is inefficient due to the large size of shared data in the cloud. In this paper, we propose a novel public auditing mechanism

For the integrity of shared data with efficient user revocation in mind. By utilizing the idea of proxy re-signatures, we allow the cloud tore-sign blocks on behalf of existing users during user revocation, so that existing users do not need to download and re-sign blocks by themselves. In addition, a public verifier is always able to audit the integrity of shared data without retrieving the entire data from the Cloud, even if some part of shared data has been re-signed by the cloud. Moreover, our mechanism is able to support batch auditing by verifying multiple auditing tasks simultaneously. Experimental results show that our mechanism can significantly improve the efficiency of user revocation. In this work, we design a novel public verifiable authenticator, which is generated by a couple of keys and can be regenerated using partial keys. Thus, our scheme can completely release data owners from burden and make him feel secure.

Index Terms – Data Integrity, Cloud, Proxy, Re-signatures.

1. INTRODUCTION

Advances in networking technology and an increase in the need for computing resources have prompted many organizations to outsource their storage and computing needs. This new economic and computing model is commonly referred to as cloud computing and includes various types of services such as: infrastructure as a service (IaaS), where a customer makes use of a service provider's computing, storage or networking infrastructure; platform as a service (PaaS), where a customer leverages the provider's resources to run custom applications; and finally software as a service (SaaS), where customers use software that is run on the provider's infrastructure.

Cloud infrastructures can be roughly categorized as either private or public. In a private cloud, the infrastructure is managed and owned by the customer and located on-premise (i.e., in the customer's region of control). In particular, this means that access to customer data is under its control and is only granted to parties it trusts. In a public cloud the infrastructure is owned and managed by a cloud service provider and is located off-premise (i.e., in the cloud service provider's region of control). This means that customer data is outside its control and could potentially be granted to untrusted parties.

Storage services based on public clouds such as Microsoft's Azure storage service and Amazon's S3 provide customers with scalable and dynamic storage. By moving their data to the cloud customers can avoid the costs of building and maintaining a private storage infrastructure, opting instead to pay a service provider as a function of its needs. For most customers, this provides several benefits including availability (i.e., being able to access data from anywhere) and reliability (i.e., not having to worry about backups) at a relatively low cost. While the benefits of using a public cloud infrastructure are clear, it introduces significant security and privacy risks. In fact, it seems that the biggest hurdle to the adoption of cloud storage (and cloud computing in general) is concern over the confidentiality and integrity of data. While, so far, consumers have been willing to trade privacy for the convenience of software services (e.g., for web-based email, calendars, pictures etc…), this is not the case for enterprises and government organizations. This reluctance can be attributed to several factors that range from a desire to protect mission-critical data to regulatory obligations to preserve the confidentiality and integrity of data. The latter can occur when the customer is responsible for keeping personally identifiable information (PII), or medical and financial records.

So while cloud storage has enormous promise, unless the issues of confidentiality and integrity are addressed many potential customers will be reluctant to make the move. To address the concerns outlined above and increase the adoption of cloud storage, we argue for designing a virtual private storage service based on new cryptographic techniques. Such a service should aim to achieve the "best of both worlds" by providing the security of a private cloud and the functionality and cost savings of a public cloud. More precisely, such a service should provide (at least):

Confidentiality: the cloud storage provider does not learn any information about customer data integrity: any unauthorized

modification of customer data by the cloud storage provider can be detected by the customer non-repudiation: any access to customer data is logged, while retaining the main benefits of a public storage service:

Availability: customer data is accessible from any machine and at all times.

Reliability: customer data is reliably backed up efficient retrieval: data retrieval times are comparable to a public cloud storage service.

Data sharing: customers can share their data with trusted parties.

An important aspect of a cryptographic storage service is that the security properties described above are achieved based on strong cryptographic guarantees as opposed to legal, physical and access control mechanisms. We believe this has several important benefits.

2. OVERVIEW

The main aim of the project is to support dynamic groups efficiently. We provide secure and privacy-preserving access control to users, which guarantees any member in a group to anonymously utilize the cloud resource. To achieve secure data sharing for dynamic groups in the cloud, we expect to combine the group signature or group master key and dynamic broadcast encryption techniques.

Specially, the group signature scheme enables users to anonymously use the cloud resources, and the dynamic broadcast encryption technique allows data owners to securely share their data files with others including new joining users. The proxy server computes the revocation parameters and make the result public available by migrating them into the cloud. Such a design can significantly reduce the computation overhead of users to encrypt files and the cipher text size. Specially, the computation overhead of users for encryption operations and the cipher text size is constant and independent of the revocation users. The proxy maintains the signature delegation work which generates the private and public key of each group so that the permission for the access of file can be restricted. Revocation is user is performed if any user makes unauthenticated action on any data in the cloud. Also if a data has been modified by the user it will be detected, penalized and the code will be regenerated by the proxy.

In this the user revocation is performed by the group manager via a public available revocation list (RL), based on which group members can encrypt their data files and ensure the confidentiality against the revoked users. The list is characterized by time stamp t1,t2,…tr. In the proposed system once the user time stamp over does not wait for the group manager to update the time stamp or revocation list here once the time over the user immediately send request for extra time for access the data to the cloud. Then the cloud will send that request to the group manager once the see it and give permission then the cloud will time to access the data but if the group manager did not give permission then the cloud will not give permission for access of the data.

3. SYSTEM MODEL

It consists of four modules,

- Cloud Module.
- Proxy Server Module.
- Group Member Module.
- User Revocation Module.

3.1. User and Data Maintenance

The registered users and data will be maintained using a cloud server. A local Cloud which provides priced abundant storage services are been created in this module. The users can upload their data in the cloud. This module can be developed where the cloud storage can be made secure. The cloud is not fully trusted by users since the CSPs are very likely to be outside of the cloud users' trusted domain. Similar to that the cloud server is honest but curious. That is, the cloud server will not maliciously delete or modify user data due to the protection of data auditing schemes, but will try to learn the content of the stored data and the identities of cloud users. This essentially means that the owner (client) of the data moves its data to a third party cloud storage server which is supposed to - presumably for a fee - faithfully store the data with it and provide it back to the owner whenever required. The cloud server provides privilege to generate secure multi-owner data sharing scheme called MONA. It implies that any user in the group can securely share data with others by the cloud. This scheme is able to support dynamic groups efficiently. Specifically, new granted users can directly decrypt data files uploaded before their participation without contacting with data owners but within the group.

3.2 Authentication and Signature Generation

This module makes the following functions

1. Signature Generation,
2. Signature Verification,
3. Content Regeneration.

A proxy agent acts on behalf of the data owner to regenerate authenticators and data blocks on the servers during the repair procedure. Notice that the data owner is restricted in computational and storage resources compared to other entities and may becomes off-line after the data upload procedure. The proxy, who would always be online, is supposed to be much more powerful than the data owner but less than the cloud servers in terms of computation and memory capacity. To save resources as well as the online burden potentially brought by the periodic auditing and accidental repairing, the data owners resort to the TPA for integrity verification and delegate the

reparation to the proxy. Considering that the data owner cannot always stay online in practice, in order to keep the storage available and verifiable after a malicious corruption, we introduce a semi-trusted proxy into the system model and provide a privilege for the proxy to handle the reparation of the coded blocks and authenticators. It generates signature using OAEP based key delegation which provides unique private and public key for each group registered in the cloud. So the users can access the document provided by its own group only. The users can view other groups document using private key of the other groups. If he modifies other group content he will be revoked by the cloud server.

3.3. User logs

Group members are a set of registered users that will

1. Store their private data into the cloud server and
2. Share them with others in the group.

This modules maintains the user's details in it. The group membership is dynamically changed, due to the staff resignation and new employee participation in the company. The group member has the ownership of changing the files in the group. All the users in the group can view the files which are uploaded in their group and also can modify it. Also each group will have private key and public key in it. The public key is used for viewing the document in the cloud whereas the private is the meant for providing modification rights for an user.

3.4 User Revocation and file regeneration:

User revocation is performed by the proxy via a public available RL based on which group members can encrypt their data files and ensure the confidentiality against the revoked users. No unauthorized access to the document is encouraged in the cloud storage. So the data should be provided rights to modify only by the group's own users. Other members cannot modify the content. Once if any user tries to hack the private key of another group and trying to modify this will be detected by the cloud server and the user's account will be revoked by the user. The user could never enter his login again. This function will be performed by the cloud server. Also if content is modified by unauthorized user it will be rollback to its original state by the cloud server.

4. CONCLUSION

In this paper, we introduce a public investigating scheme that which can be used for regenerating code over the cloud storage system, where the data owners are privileged to delegate TPA for their data validity checking for keep the original data privacy opposition to the TPA. There is the blind technique during the auditing process and no data owners can always stay in online so in order to keep the storage available and verifiable after a malicious corruption and we also propose a semi trusted proxy into the system model that which can be used to provide a privilege for the proxy to handle the reparation of the coded blocks and authenticators. The best approach for the regenerating code scenario is mapping our authenticator based on the BLS signature. So that the data owners can generate code based on cloud storage system with high efficiently and very secure.

REFERENCES

[1] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl, "A secure data regeneration scheme for cloud storage," in *Technical Report*, 2013.
[2] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer, "Reclaiming space from regenerate files in a serverless distributed file system." in *ICDCS*, 2002, pp. 617–624.
[3] P. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de-regeneration," in *Proc. of USENIX LISA*, 2010.
[4] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Serveraided encryption for deregenerated storage," in *USENIX Security Symposium*, 2013.
[5] G. R. Blakley and C. Meadows, "Security of ramp schemes," in *Advances in Cryptology: Proceedings of CRYPTO '84*, ser. Lecture Notes in Computer Science, G. R. Blakley and D. Chaum, Eds. Springer-Verlag Berlin/Heidelberg, 1985, vol. 196, pp. 242–268.
[6] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, "Secure regeneration with efficient and reliable convergent key management," in *IEEE Transactions on Parallel and Distributed Systems*, 2014, pp. vol. 25(6), pp. 1615–1625.
[7] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman- Peleg, "Proofs of ownership in remote storage systems." in *ACM Conference on Computer and Communications Security*, Y. Chen, G. Danezis, and V. Shmatikov, Eds. ACM, 2011, pp. 491–500.
[8] C. Liu, Y. Gu, L. Sun, B. Yan, and D. Wang, "R-admad: High reliability provision for large-scale de-regeneration archival storage systems," in *Proceedings of the 23rd international conference on Supercomputing*, pp. 370–379.
[9] M. Li, C. Qin, P. P. C. Lee, and J. Li, "Convergent dispersal: Toward storage-efficient security in a cloud-of-clouds," in *The 6th USENIX Workshop on Hot Topics in Storage and File Systems*, 2014.
[10] J. S. Plank and L. Xu, "Optimizing Cauchy Reed-solomon Codes for fault-tolerant network storage applications," in *NCA-06: 5th IEEE International Symposium on Network Computing Applications*, Cambridge, MA, July 2006.
[11] D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Side channels in cloud services: Regeneration in cloud.
[12] J. S. Plank, S. Simmerman, and C. D. Schuman, "Jerasure: A library in C/C++ facilitating erasure coding for storage applications - Version 1.2," University of Tennessee, Tech. Rep. CS-08-627, August 2008.