ISSN: 2347-5420 Impact Factor : 1.5015(UIF) Volume - 2 | Issue - 3 | Feb - 2016

Industrial Science

CONFIDENTIALITY - PRESERVING AUTOMATED AUDITING FOR SHARED DATA IN CLOUD STORAGE

R. Chitra[1] and P. Vidya[2]

[1]Assistant Professor, Department of Computer Science & Engineering,
Indira Institute of Engineering and Technology, Pandur, Thiruvallur, Tamil Nadu.

[2]Master of Engineering (2nd Year), Department of Computer Science & Engineering,
Indira Institute of Engineering and Technology, Pandur, Thiruvallur, Tamil Nadu.

ABSTRACT

The cloud security is one of the important roles in cloud; here we can preserve our data into cloud storage. The security issues are the major thing in cloud but Cloud service is necessary. Here we can overcome the security issues in our project. In existing they are using a remote verification technique to audit by the third party or private auditing. In this technique the data owners need to be online to manage that auditing. In our system we are using the own auditing based on the token generation. Using this token generation technique compare the token values from original tokens we can find out the changes about the file. Users can login into their account then they upload our files. The files will be stored into the cloud storage. In our system we provide the two tier security for our uploaded files. The files does not stored directly it will be converted into the files, it will be stored into three different cloud server locations. The original file content split into three parts and it will be store into each files. Not only stored also the content will be encrypted in the cloud server. If anyone try to hack at the cloud end is not possible to break the two tier block. They need first decrypt the files and also combine the splited files from three different locations. This is not possible by anyone. Anyone can download the files from the server with file owner permission. At the time of download key generated (code based key generation) and it will send to the file owner. We can download the file need to use the key for verification and some other users want to download file owner permission is necessary.

KEYWORDS :Cloud storage, regenerating codes, automated audit, confidentiality preserving, authenticator regeneration, proxy, privileged, provable secure.

INTRODUCTION :

CLOUD storage is now gaining popularity because it offers a flexible on-demand data outsourcing service with appealing benefits: relief of the burden for storage management, universal data access with location independence and avoidance of capital expenditure on hardware, software and personal maintenances, etc. Nevertheless, this new paradigm of data hosting service also brings new security threats toward user's data, thus making individuals or enterprisers still feel hesitant. It is noted that data owners lose ultimate control over the fate of their outsourced data; thus, the correctness, availability and integrity of the data are being put at risk. In our project, we propose an effective and flexible distributed scheme with data in the cloud. Here we are using the erasure code technique for distribute the data to cloud locations and access the data from cloud. User can register and login into their account. They have a option to store, share and access the data from cloud storage. Here we are using the two tier security scheme for storing data into the cloud. The first tier security is your data or file splited into multiple parts and it will store into different cloud server locations. Each and every file generates the token for auditing. Then second tier security is each and every splited file will encrypt before store into different locations. The shared users can edit the file in the cloud with file owner's permission. That file eligible of own public auditing. Then user can have to login and access the own files or some other files. User first can search and download the files, at the time of download user should use the security key. If authentication success it will be decrypt and combine to get the original data from cloud

CLOUD COMPUTING

Cloud computing is the delivery of computing and storage capacity as a service to a heterogeneous community of end-recipients. Cloud computing is a general term for anything that involves delivering hosted services over the Internet. A model for delivering information technology services in which resources are retrieved from the internet through web-based tools and applications. Cloud computing is so named because the information being accessed is found in the "clouds", and does not require a user to be in a specific place to gain access to it. Cloud computing refers to applications and services offered over the Internet. These services are offered from data centers all over the world, which collectively are referred to as the "cloud." The idea of the "cloud" simplifies the many network connections and computer systems involved in online services. Cloud computing is computing model, not a technology. In this model of computing, all the servers, networks, applications and other elements related to data centers are made available to IT and end users. Cloud computing is a type of computing that is comparable to grid computing. It relies on sharing computing resources rather than having local servers or personal devices to handle applications.

EVIDENT REASONS FOR SHIFTING TO CLOUD

• Reduced cost: Cloud computing reduce the capital cost and operating expenses because resources are needed on the basis of pay per use.

• Refined usage of personnel: In cloud computing, the users focus on delivering values rather than maintaining hardware and software.

• Robust scalability: Cloud computing allows to have immediate scaling, either increase or decrease, at any time for feature commitment.

Software as a Service

SaaS has become a common delivery model for many business applications, including

accounting, collaboration, customer relationship management (CRM), management information systems.

Platform as a Service

It is a category of cloud computing services that provide a computing platform and a solution stack as a service. Along with SaaS and IaaS, it is a service model of cloud computing.

Infrastructure as a Service (IaaS)

IaaS refers not to a machine that does all the work, but simply to a facility given to businesses that offers users the leverage of extra storage space in servers and data centers.

FEATURES OF CLOUD COMPUTING

Consumption based billing: Pay per use seems to be the winning characteristic for cloud.
Rapid Elasticity: Consumers can increase or decrease the capacity within a matter of minutes. This can be by adding instances in cases of EC2 or by just increasing memory in some other cases.
Self Service Based Model: Users have the ability to upload, build, deploy, schedule, manage, and report on their business services on demand.
Location and Device Independence: Users can access the server through the internet from any place and from any device.
Multi-tenancy: There is no need for additional resources like real estate, electricity etc for the servers.
Reliability: Reliability is improved by having multiple sites for the same service, such that if one faces an outage, the other can take over the load for the time being.
Ease of Maintenance: Since, the applications don't need to be deployed on every person's computer; maintenance of such a centralized application becomes much easier.
Reduce Financial Overheads: By utilizing cloud computing, you could decrease financial overheads which are related to possessing huge IT departments and your system could do away with larger portions of the process of IT.
Less Prone to Failure: Also, without huge files stored upon your personal systems, they'll be more effective and less prone to failure – whether it is temporary or otherwise.
Safety: With every one of these elements, safety.

ISSUES AND CHALLENGES

Cloud computing comes with numerous security issues because it encompasses many technologies including networks, databases, operating systems, virtualization, resource scheduling, transaction management, load balancing, concurrency control and memory management. Hence, security issues of these systems and technologies are applicable to cloud computing. For example, it is very important for the network which interconnects the systems in a cloud to be secure. Also, virtualization paradigm in cloud computing results in several security concerns. For example, mapping of the virtual machines to the physical machines has to be perform very securely.

Data security not only involves the encryption of the data, but also ensures that appropriate policies are enforced for data sharing. In addition, resource allocation and memory management algorithms also have to be secure. The big data issues are most acutely felt in certain industries, such as telecoms, web marketing and advertising, retail and financial services, and certain government activities. The data explosion is going to make life difficult in many industries, and the companies will gain considerable advantage which is capable to adapt well and gain the ability to analyze such data explosions over those other companies

The challenges of security in cloud computing environments can be categorized into network level, user authentication level, data level, and generic issues.

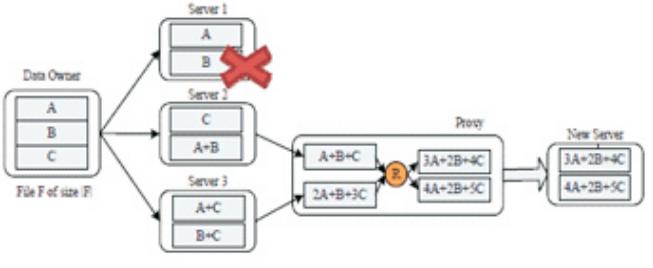
Network level: The challenges that can be categorized under a network level deal with network protocols and network security, such as distributed nodes, distributed data, Inter node communication.
Authentication level: The challenges that can be categorized under user authentication level deals with encryption/decryption techniques, authentication methods such as administrative rights for nodes, authentication of applications and nodes, and logging.
Data level: The challenges that can be categorized under data level deals with data integrity and

availability such as data protection and distributed data.

Generic types: The challenges that can be categorized under general level are traditional security tools, and use of different technologies

PRELIMINARIES AND PROBLEM STATEMENT

1) Regenerating Codes: Regenerating codes are first introduced by A.G. Dimakis et al. [18] for distributed storage to reduce the repair bandwidth. Viewing cloud storage to be a collection of n storage servers, data file F is encoded and stored redundantly across these servers. Then F can be retrieved by connecting to any k-out-of-n servers, which is termed the MDS2-property. When data corruption at a server is detected, the client will contact l healthy servers and download β′ bits from each server, thus regenerating the corrupted blocks without recovering the entire original file. Dimakis [18] showed that the repair bandwidth γ′ = l β′ can be significantly reduced with l ≥k. Furthermore, they analyzed the fundamental tradeoff between the storage cost α and the repair bandwidth γ′, then presented two extreme and practically relevant points on the optimal tradeoff curve: the minimum bandwidth regenerating (MBR) point, which represents the operating point with the least possible repair bandwidth, and the minimum storage regenerating (MSR) point, which corresponds to the least possible storage cost on the servers..Moreover, according to whether the corrupted blocks can be exactly regenerated, there are two versions of repair strategy: exact repair and functional repair. Exact repair strategy requires the repaired server to store an exact replica of the corrupted blocks, while functional repair indicates that the newly generated blocks are different from the corrupted ones with high probability. As one basis of our work, the functional repair regenerating codes are non-systematic and do not perform as well for read operation as systematic codes, but they really make sense for the scenario in which data repair occurs much more often than read, such as regulatory storage, data escrow and long-term archival storage.



2) Linear Subspace from Regenerating code: As mentioned above, each coded block represents the linear combination of m native blocks in the functional repair regenerating code scenario. Thus, we can generate a linear subspace with dimension m for file F in the following way:

Before encoding, F is split into m blocks, and the original m s-dimensional vectors (or blocks indistinguishably) and the remaining elements indicate the coding coefficients. Notice that the blocks regenerated in the repair phase also meet the form thus we can construct a linear subspace V of m-dimension by spanning the base vectors w1, w2, ..., wm; all valid coded blocks appended with coding coefficients would belong to subspace V .Under the construction of linear subspace V, we can generate tags for vectors in V efficiently, i.e., we only need to sign the m base vectors in the beginning. Such a signature scheme can be viewed as similar with signing on the subspace V.

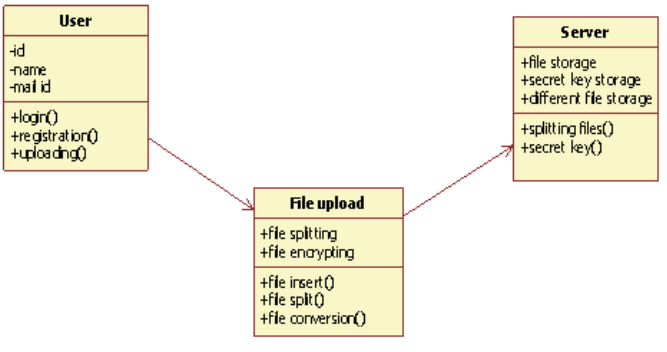
3) System Model: We consider the auditing system model for Regenerating-Code-based cloud storage which involves four entities: the data owner, who owns large amounts of data files to be stored in the cloud; the cloud, which are managed by the cloud service provider, provide storage service and have significant computational resources; the third party auditor(TPA), who has expertise and capabilities to conduct public audits on the coded data in the cloud, the TPA is trusted and its audit result is unbiased for both data owners and cloud servers; and a proxy agent, who is semi-trusted and acts on behalf of the data owner to regenerate authenticators and data blocks on the failed 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 even 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.

THE PROPOSED APPROACH

In this section we start from an overview of our auditing scheme, and then describe the main scheme and discuss how to generalize our privacy-preserving automated auditing scheme. Furthermore, we illustrate some optimized methods to improve its performance.

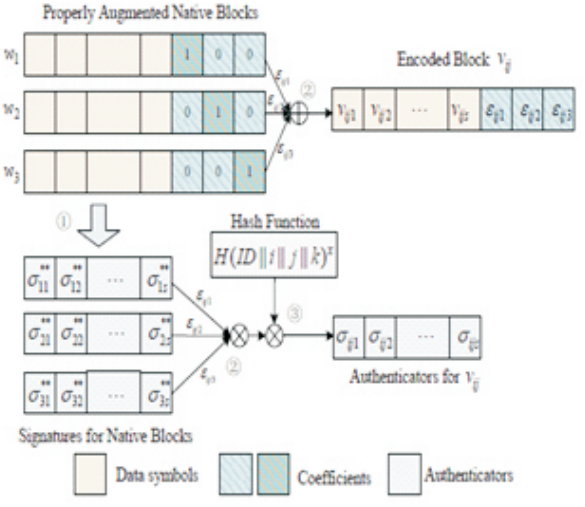
Overview

Although introduced private remote data checking schemes for regenerating-code-based cloud storage, there are still some other challenges for us to design a public auditable version. First, although a direct extension of the techniques can realize public verifiability in the multi-servers setting by viewing each block as a set of segments and performing spot checking on them, such a straightforward method makes the data owner generate tags for all segments independently, thus resulting in high computational overhead. Considering that data owners commonly maintain limited computation and memory capacity, it is quite significant for us to reduce those overheads. Second, unlike cloud storage based on traditional erasure code or replication, a fixed file layout does not exist in the regenerating-code-based cloud storage. During the repair phase, it computes out new blocks, which are totally different from the faulty ones, with high probability. A direct solution, which is adopted in, is to make data owners handle the regeneration. However, this solution is not practical because the data owners will not always remain online through the life-cycle of their data in the cloud; more typically, it becomes off-line even after data uploading. In our system we are using the own auditing based on the token generation. Using this token generation technique compare the token values from original tokens we can find out the changes about the file.



Construction of our Auditing Scheme

Considering the regenerating-code-based cloud storage with parameters (n, k, l , α, β), we assume β = 1 for simplicity. Let G and GT be multiplicative cyclic groups of the same large prime order p, and e: G × G →GT be a bilinear pairing map as introduced in the preliminaries. Let g be a generator of G and H (•): {0, 1}∗ →G be a secure hash function that maps strings uniformly into group G.



SECURE ERASURE CODE ALGORITHM:

Step 1: Storing data in a third party's cloud system causes serious concern over data confidentiality.

Step 2: Constructing a secure storage system that supports multiple functions is challenging when the storage system is distributed and has no central authority.

Step 3: A threshold proxy re-encryption scheme and integrate it with a decentralized erasure code such that a secure distributed storage system is formulated.

Step 4: The distributed storage system not only supports secure and robust data storage and retrieval, but also lets a user forward his data in the storage servers to another user without retrieving the data back.

Step 5: 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 we encrypted messages.

Step 6: It's stored the number of copies of a message dispatched to storage servers and the number of storage servers queried by a key server.

User interface

Users can use them from anywhere at any time. For example, the email service is probably the most popular one. Cloud computing is a concept that treats the resources on the Internet as a unified entity, a cloud.

Users just use services without being concerned about how computation is done and storage is managed. In this paper, we focus on designing a cloud storage system for robustness, confidentiality, and functionality. A cloud storage system is considered user interface entry level creation in this module.

File uploading process

Storing data over storage servers one way to provide data robustness is to replicate a message such that each storage server stores a message. Another way is to encode a message of k symbols into a codeword of n symbols by erasure coding. To store a message, each of its codeword symbols is stored in a different storage server. A storage server corresponds to an erasure error of the codeword symbol. As long as the number of servers is under the tolerance threshold of the erasure code, the message can be recovered from the codeword symbols stored in the available storage servers by the decoding process.

Secret key generation

The data forwarding phase, user forwards his encrypted message with an identifier ID stored in storage servers to user such that can decrypt the forwarded message by using his secret key. The secret keys of target users, and the shared keys stored in key servers.

Mail alert process

The uploading and downloading process of the user is first get the secret key in the corresponding user email id and then apply the secret key to encrypted data to send the server storage and decrypts it by using his secret key to download the corresponding data file in the server storage system's the secret key conversion using the Share Key Gen (SKA, t, m). This algorithm shares the secret key SKA of a user to a set of key servers.

File Downloading process

File downloading process is to get the corresponding secret key to the corresponding file to the user mail id and then decrypt the file data. The file downloading process re-encryption key to storage servers such that storage servers perform the re- Encryption Operation. The length of forwarded message and the computation of re-encryption is taken care of by storage servers. Proxy re-encryption Schemes significantly reduce the overhead of the data Forwarding function in a secure storage system.

CONCLUSION

In this paper, we propose a confidentiality-preserving automated auditing system for data storage security in cloud computing. It not only eliminates the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage by performing automated auditing and splitting the encrypted files and stores in different locations.

REFERENCES

1.B. Wang, B. Li, and H. Li, "Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud," Proc. IEEE Fifth Int'l Conf. Cloud Computing, pp. 295-302, 2012.
2.M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.
3.K. Ren, C. Wang, and Q. Wang, "Security Challenges for the Public Cloud," IEEE Internet Computing, vol. 16, no. 1, pp. 69-73, 2012.
4.D. Song, E. Shi, I. Fischer, and U. Shankar, "Cloud Data Protection for the Masses," Computer, vol. 45, no. 1, pp. 39-45, 2012.
5.C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," Proc. IEEE INFOCOM, pp. 525-533, 2010.
6.Chief Information Officers Council, "Privacy Recommendations for Cloud Computing",

http://www.cio.gov/Documents/Privacy-RecommendationscloudComputing-8-19-2010.docx
7.NIST SP 800-144, "Guidelines on Security and Privacy Issues in Public Cloud Computing", http://csrc.nist.gov/publications/drafts/800-144/Draft-SP-800-144_cloud-computing.pdf
8.H. Chen and P. Lee, "Enabling data integrity protection in regenerating coding-based cloud storage: Theory and implementation," Parallel and Distributed Systems, IEEE Transactions on, vol. 25, no. 2, pp. 407–416,Feb 2014.
9.K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," Parallel and Distributed Systems, IEEE Transactions on, vol. 24, no. 9, pp. 1717–1726, 2013.
10.Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," Parallel and Distributed Systems, IEEE Transactions on, vol. 23, no. 12, pp. 2231–2244, 2012.
11.A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," Proceedings of the IEEE, vol. 99,no. 3, pp. 476–489, 2011.
12.H. Shacham and B. Waters, "Compact proofs of retrievability," in Advances in Cryptology-ASIACRYPT 2008. Springer, 2008, pp. 90–107.
13.Y. Hu, H. C. Chen, P. P. Lee, and Y. Tang, "Nccloud: Applying network coding for the storage repair in a cloud-of-clouds," in USENIX FAST, 2012.
14.C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in INFOCOM, 2010 Proceedings IEEE. IEEE, 2010, pp. 1–9.
15.A. Juels and B. S. Kaliski Jr, "Pors: Proofs of retrievability for large files," in Proceedings of the 14th ACM conference on Computer and communications security. ACM, 2007, pp. 584–597.