International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 3, March 2015
Available at http://internationaljournalofresearch.org

Providing Security and Integrity of Public Data with Proficient User Revocation in the Cloud

[1]R. Sravanthi ; [2]Dr. P. Venkateswarlu ; [3]M. Naresh Choudary

[1] M.Tech (CSE), Department of Computer Science & Engineering Nagole Institute of Technology & Science, Kuntloor (V), Hayathnagar (M), RR District, Hyderabad, India.
E-mail id: sravz.rudra@gmail.com
[2] Professor & HOD, Department of Computer Science & Engineering.
E-mail id: venkat123.pedakolmi@gmail.com
[3] Assistant Professor, Department of Computer Science & Engineering.
E-mail id: naresh.makkena@hotmail.com

Abstract:

With popularization of cloud services, multiple users easily share and update their data through cloud storage. For data integrity and security in the cloud storage, users in the group need to compute signatures on all the blocks in shared data, and due to data modifications performed by different users, different blocks are signed by different users. For security reasons, when 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 straightforward 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 the integrity of shared data with efficient user revocation in mind. By utilizing the idea of proxy re-signatures, we allow the cloud to re-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.

Keywords—

Integrity; Efficiency; User Revocation; Re-sign; Cloud Service Provider (CSP); Public Auditing; Trusted Third party (TPA); Client Repudiation

1. INTRODUCTION

In recent years, the emerging cloud-computing paradigm is rapidly gaining momentum as an alternative to traditional Information technology. Cloud storage, an important service of cloud computing, allows users to move data from their local storage systems to the cloud and enjoy the on-demand high quality cloud services. For data storage and sharing services like Google Drive and Drop box provided by cloud, people can easily shared data and work together in group. Once a user creates shared data in cloud, every user in the group is able to not only access data but also modify shared data. Although cloud providers promise a more secure and reliable environment to the users, the integrity of data in the cloud may still be compromised, due to the existence of hardware/software failures and human errors [2]. To protect the integrity of data in the cloud, a signature is attached to each block in data, and the integrity of data relies on the correctness of all the signatures. One of the most significant and common features of these it allow a public verifier to efficiently check data integrity in the cloud without downloading the entire data, referred to as public auditing. This public verifier could be

A) Client: who would like to utilize cloud data for particular purposes (e.g., search, computation, data mining, etc.)Or

B) a third-party auditor (**TPA**) who is able to provide verification services on data integrity to users. For security reasons, when a user leaves the group or is behave, this user must be revoked from the group. As a result, this revoked user should no longer be able to access and modify shared data, and the signatures generated by this revoked user are no longer valid to the group. Therefore, although the content of shared data is not changed during user revocation, the blocks, which were previously signed by the revoked user, still need to be re-signed by an existing user in the group. As a result, the integrity of the entire data can still be verified with the public keys of existing users only. Since shared data is outsourced to the cloud and users no longer store it on local devices, a straightforward method to re-compute these signatures during user revocation (as shown in Fig. 1) is to ask an existing user (i.e., Alice) to first download the blocks previously signed by the revoked user (i.e., Bob), verify the correctness of these blocks, then re-sign these blocks, and finally upload the new signatures to the cloud. However, this straightforward method may cost the existing user a huge amount of communication and computation resources by downloading and verifying blocks, and by re-computing and uploading signatures, especially when the number of re-signed blocks is quite large or the membership of the group is frequently changing. Clearly, if the cloud could possess each user's private key, it can easily finish the re-signing task for existing users without asking them to download and re-sign blocks. However, since the cloud is not in the same trusted domain with each user in the group, outsourcing every user's private key to the cloud would introduce significant security issues. Another important problem we need to consider is that the re-computation of any signature during user revocation should not affect the most attractive property of public auditing auditing data integrity publicly without retrieving the entire data. Therefore, how to efficiently reduce the significant burden to existing users introduced by user revocation, and still allow a public verifier to check the integrity of shared data without downloading the entire data from the cloud, is a challenging task. In this paper, we propose by utilizing the idea of proxy re-signatures, once a user in the group is revoked, the cloud is able to resign the blocks, which were signed by the revoked user, with a re-signing key. The square measure many studies collateral integrity of outsourced information at untrusted storages [8] Most of them [8] square measure however to contemplate a state of affairs wherever a similar information is shared by multiple users. In these approaches, solely one user is allowed to update his own information. And he will audit the information either by himself [8, 9] or with help from a third-party auditor (TPA. Recent studies think about audit for shared information however they solely support a restricted range of information updates. Additionally, the CSP will cheat censorship in these schemes since Associate in nursing index table used for verification is managed solely by the CSP. A way to forestall such a cheat is to form users and therefore the TPA also maintains the index table. Gratitude to storage and synchronization overhead, however, it'd because a major delay and degrade the Quality of service (QoS) because the range of information updates will increase. In order to style a secure and economical audit mechanism for dynamic shared information in cloud storage, said challenges ought to be expeditiously addressed. In different words, the theme should guarantee the subsequent properties.

(1) Audit for Outsourced information. The TPA is ready to see the integrity of outsourced information while not retrieving all information contents.

(2) Shared Dynamic information. Users square measure allowed to source, share, insert, delete, or modify their information contents while not restriction.

(3) Efficiency. Procedure overhead for information outsourcing and update at users facet similarly because the ones for auditing at the TPA ought to be low.

(4) Soundness. The CSP isn't allowed to deceive users or the TPA into passing a censorship of broken information contents. We propose Associate in Nursing audit mechanism satisfying the on top of necessities by utilizing mixture signature and sample auditing [8]. For information integrity and consistency, the TPA manages Associate in tending index table and therefore the CSP keeps invigorating Associate in tending symbol for information update. Additionally, the audit mechanism provides potency to users and therefore the TPA through creating the auditing operations easy. Specially, during this paper, we tend to think about forge attack and replace attack as regards soundness for the sake of secure audit. These attacks square measure delineated in and that they are often summarized as follows. Forge attack is Associate in Nursing attack to forge a collateral term for a knowledge content, that wasn't really outsourced by users. Replace attack is Associate in Nursing attack to pass a censorship by selecting another information content for verification in situation of the broken information content correctness of shared information within the cloud. With shared information, once a user modifies a block, she additionally must figure a replacement signature for the changed block. As a result of the modifications from completely different users, totally completely different blocks square measure signed by different users. For security reasons, once a user leaves the cluster or misbehaves, this user should be revoked from the cluster. As a result, this revoked user ought to now not be able to access and modify shared information, and therefore the signatures generated by this revoked user are not any longer valid to the cluster .Therefore, though the content of shared information isn't modified throughout user revocation, the blocks, that were antecedently signed by the revoked user, still ought to be re-signed by Associate in Nursing existing user within the cluster, so that, when the revocation, the integrity of the whole information will still be verified with the general public keys of existing users solely. Since shared information is outsourced to the cloud and users now not store it on native devices, the simple technique to re-compute these signatures throughout user revocation

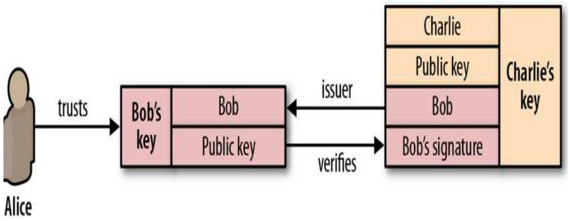


Fig 1. File revoking with signatures in cloud

2. PROBLEM STATEMENT

In this section, we tend to describe the system and threat model of this paper, and illustrate the look goals of our public auditing mechanism.

A. System and Threat Model

In this paper, the system model includes 3 entities: the cloud, the third party auditor (TPA), and users UN agency share information as a gaggle (as illustrated in Fig. 2). The cloud offers information storage and sharing services to users. The TPA is ready to publically audit the integrity of shared information within the cloud for users. In a group, there's one original user and variety of cluster users. The first user is that the original owner of information. This original user creates and shares information with alternative users within the cluster through the cloud. Each the first user and cluster users' area unit ready to access, transfer and modify shared information. Shared information is additional divided into variety of blocks. A user will modify a block in shared information by acting associate insert, delete or update operation on the block.

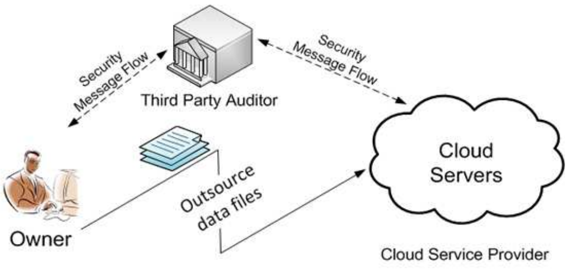


Fig. 2. The system model includes the cloud, the TPA, and users.

Generally, the integrity of shared information is vulnerable by 3 factors. First, the cloud service supplier might unwittingly shared information because of hardware/software failures and human errors. Second, associate external opponent might try and corrupt shared information within the cloud, and stop users from mistreatment shared information properly. Third, a revoked user, UN agency not has the correct as existing users, might try and illicitly modify shared information. Considering these threats, users don't totally trust the cloud with the integrity of shared information. To shield the integrity of shared information, every block in shared information is hooked up with a signature that is computed by one in all the users within the cluster. Once shared information is at the start created by the first user within the cloud, all the signatures on shared information area unit are computed by the first user. After that, once a user modifies a block, this user conjointly must sign the changed block with his/her own personal key. By sharing information among a gaggle of users, completely different blocks could also be signed by different users because of modifications from different users.

When a user within the cluster leaves or misbehaves, the cluster must revoke this user. Generally, because the creator of shared information, the first user acts because the cluster manager and is ready to revoke users on behalf of the cluster. Once a user is revoked, the signatures computed by this revoked user become invalid to the cluster, and also the blocks that were antecedently signed by this revoked user got to be re-signed by associate existing user, in order that the correctness of the whole information will still be verified with the general public keys of existing users solely.

B. Design Goals

To correctly verify the integrity of shared information with economical user revocation, our public auditing mechanism ought to succeed the subsequent properties: (1) Correctness: The TPA is ready to properly check the integrity of shared information. (2) Economical and Secure User Revocation: On one hand, once a user is revoked from the cluster, the blocks signed by the revoked user will be with efficiency re-signed. On the opposite hand, solely existing users within the cluster will generate valid signatures on shared information, and also the revoked user will not work out valid signatures on shared information. (3) Public Auditing: The TPA will audit the integrity of shared information while not retrieving the whole information from the cloud, even if some blocks in shared information are re-signed by the cloud.

3. SYSTEM FRAMEWORK:

Presented System:

With the bestowed system framework the file transferred in cloud that not signed by user in every time of upload. In order that integrity of shared information isn't attainable in existing system. However, since the cloud isn't within the same trusty domain with every user within the cluster, outsourcing each user’s personal key to the cloud would introduce important security issue.

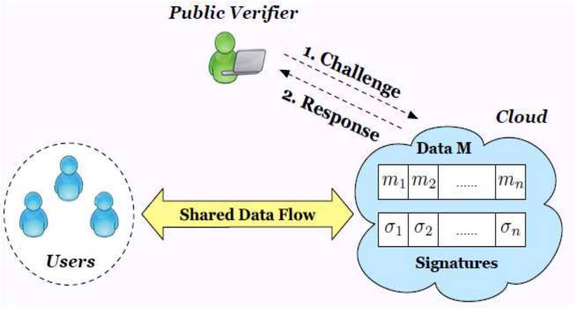
Proposed System:

In our projected system framework might hoodwink verifiers concerning the incorrectness of shared information so as to save lots of the name of its information services and avoid losing cash on its information services. Additionally, we tend to conjointly assume there's no collusion between the cloud and any user throughout the look of our mechanism. Generally, the incorrectness of share information beneath the higher than semi trusty model will be introduced by hardware/software failures or human errors happened within the cloud. Considering these factors, users don't totally trust the cloud with the integrity of shared information.

Advantage:

1. Block User accounts 2. Security question 3.Login with secret key in every time

SYSTEM ARCHITECTURE:



4. SYSTEM FUNCTIONING:

1. User: with this user module user will do registration, transfer the file, transfer the file, reupload and unblock the file:

Registration:

With this case every user has to be compelled to registers with his/her details for mistreatment files. Solely registered user will able to login in cloud server for invoking cloud services

File Upload:

With this case users transfer a block of files within the cloud with secret writing by mistreatment his/her secret key. This ensures the files to be shielded from unauthorized user.

Download:

With this case server permits the user to transfer the file mistreatment his/her secret key to decode the downloaded information of blocked user and verify the info and reupload the block of file into cloud server with secret writing .This make sure the files to be shielded from unauthorized user.

Reupload:

With this case server permit the user to reupload the downloaded files of blocked user into cloud server with resign the files(i.e.) the files is uploaded with new signature like new secret with secret writing to protected the info from unauthorized user.

Unblock:

With this case server permit the user to unblock his/her user account by respondent his security question relating to answer that provided by his/her at the time of registration. Once the solution is matched to the solution of registration time answer then solely account is unfastened.

2. Auditor (TPA): TPA (Trusted third party auditor) will perform File Verification and look at File section

File Verification:

The general public voucher is in a position to properly check the integrity of shared information. The general public voucher will audit the integrity of shared information while not retrieving the complete information from the cloud, although some blocks in shared information are re-signed by the cloud.

Files View:

With this case public auditor read the all details of transfer, downloads, blocked user, reupload.

3. Admin: here Admin will perform following tasks i.e. read Files and Block user

View Files:

With this case admin will manage all the files that square measure uploaded by the user so as to manage.

Block User:

With this case admin block the misconduct user account to shield the integrity of shared information

5. CONCLUSIONS:

In this paper, we tend to propose a unique public auditing mechanism for the integrity of shared information with economical user revocation in an untrusted cloud. In our mechanism, by utilizing the thought of proxy re-signatures, once a user within the cluster is revoked, the cloud is in a position to re-sign the blocks that were signed by the revoked user, with a re-signing key. As a result, the potency of user revocation may be considerably improved, and computation and communication resources of existing users may be simply saved

REFERENCES:

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson,and D. Song, “Provable Data Possession at Untrusted Stores,” in the Proceedings of ACM CCS 2007, 2007, pp. 598–610.

[2] H. Shacham and B. Waters, “Compact Proofs of Retrievability,” in the Proceedings of ASIACRYPT 2008. Springer-Verlag, 2008, pp. 90–107.

[3] C. Wang, Q. Wang, K. Ren, and W. Lou, “Ensuring Data Storage Security in Cloud Computing,” in the Proceedings of ACM/IEEE IWQoS 2009, 2009, pp. 1–9.

[4] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, “Enabling Public Verifiability and Data Dynamic for Storage Security in Cloud Computing,” in the Proceedings of ESORICS 2009. Springer-Verlag, 2009, pp. 355–370

[5] C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,” in the proceedings of IEEE infocom 2010,pp355-370.

[6] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S. S. Yau, “Dynamic Audit Services for Integrity Verification of Outsourced Storage in Clouds,” in n the Proceedings of ACM SAC 2011, 2011, pp. 1550–1557.

[7] B. Wang, B. Li, and H. Li, “Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud,” in the Proceedings of IEEE Cloud 2012, 2012, pp. 295–302.

[8] N. Cao, S. Yu, Z. Yang, W. Lou, and Y. T. Hou, “LT Codes-based Secure and Reliable Cloud Storage Service,” in the Proceedings of IEEE INFOCOM 2012, 2012, pp. 693–701.

[9] B. Wang, B. Li, and H. Li, “Knox: Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud,” in the Proceedings of ACNS 2012, June 2012, pp. 507–525.

AUTHOR DETAILS:

R. Sravanthi, pursuing M.Tech (CSE) from Department of Computer Science & Engineering, Nagole institute of technology & science", Hyderabad. India and received B.Tech Degree in Computer Science and Engineering from Jawaharlal Nehru Technological University, Hyderabad, India in 2012. Her interests are in the areas of cryptography, security and privacy issues of cloud computing.