Validating Integrity for Shared Data with Efficient User Revocation in the Cloud

Divya Vikas, Jai Prakash Singh, Trigun Singh, Dhatri P,
Department of ISE, SJBIT, Bangalore.

Abstract: *With data storage and sharing services in the cloud, users can easily modify and share data as a group. To ensure shared 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. In this paper, we propose a novel public auditing mechanism for the integrity of shared data with efficient user revocation in cloud. By utilizing the idea of proxy re-signatures, we allow the cloud to resign 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.*

Index Terms*—Integrity, shared data, user revocation, cloud computing*

Introduction: WITH data storage and sharing services (such as Dropbox and Google Drive) provided by the cloud, people can easily work together as a group by sharing data with each other. More specifically, once a user creates shared data in the cloud, every user in the group is able to not only access and modify shared data, but also share the latest version of the shared data with the rest of the group. 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], [3]. To protect the integrity of data in the cloud, a number of mechanisms [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] have been proposed. In these mechanisms, 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 mechanisms is to allow a public verifier to efficiently check data integrity in the cloud without downloading the entire data, referred to as public auditing (or denoted as Provable Data Possession [3]). 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 a third-party auditor (TPA) who is able to provide verification services on data integrity to users. Most of the previous works [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13] focus on auditing the integrity of personal data. Different from these works, several recent works [14], [15] focus on how to preserve identity privacy from public verifiers when auditing the integrity of shared data. Unfortunately, none of the above mechanisms, considers the efficiency of user revocation when auditing the correctness of shared data in the cloud. With shared data, once a user modifies a block, she also needs to compute a new signature for the modified block. Due to the modifications from different users, different blocks are signed by different users. For security reasons, when a user leaves the group or misbehaves, thisuser 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 [16]. 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 recompute 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. To make this matter even worse, existing users may access their data sharing services provided by the cloud with resource-limited devices, such as mobile phones, which further prevents existing users from maintaining the correctness of shared data efficiently during user revocation. In this paper, we propose Panda, a novel public auditing mechanism for the integrity of shared data with efficient user revocation in the cloud. In our mechanism, by utilizing the idea of proxy re-signatures [17], once a user in the group is revoked, the cloud is able to re-sign the blocks, which were signed by the revoked user, with a re-signing key. As a result, the efficiency of user revocation can be significantly improved, and computation and communication resources of existing users can be easily saved. Meanwhile, the cloud, who is not in the same trusted domain with each user, is only able to convert a signature of the revoked user into a signature of an existing user on the same block, but it cannot sign arbitrary blocks on behalf of either the revoked user or an existing user.

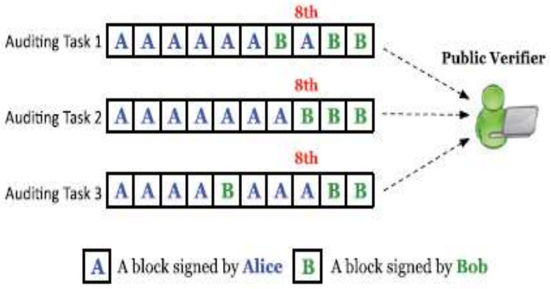

Fig-1:Alice and Bob sharing data file in the cloud

By designing a new proxy re-signature scheme with nice properties, which traditional proxy re-signatures do not have, our mechanism is always able to check the integrity of shared data without retrieving the entire data from the cloud. Moreover, our proposed mechanism is scalable, which indicates it is not only able to efficiently support a large number of users to share data and but also able to handle multiple auditing tasks simultaneously with batch auditing. In addition, by taking advantages of Shamir Secret Sharing [18], we can also extend our mechanism into the multiproxy model to minimize the chance of the misuse on resigning keys in the cloud and improve the reliability of the entire mechanism.

System And Security Model:

The cloud data storage majorly containing three entities, as given in the cloud server, the third party auditor (TPA) and users. Two kinds of users are within the group. They're the Original user and the members of the group. Group members are allowed to get into and modify shared data produced by the original user on the basis of the access control policies. Shared data and its verification information (i.e. Signatures) are generally stored in the cloud server. The third party auditor has the capacity to verify the integrity of shared data in the cloud server with respect to the group members [1]. In this paper, we simply how exactly to audit the integrity of shared data in the cloud with static groups and how to the integrity of shared data in the cloud with dynamic groups. In Static group, it is pre-defined before shared data is established in the cloud and the membership of users in the group is not changed during data sharing. The original user is accountable for deciding who has the capacity to share her data before outsourcing data to the cloud. In dynamic group, a new user could be added to the group and a pre-existing group member may be revoked during data sharing while still preserving identity privacy [1] [2]. When a user (either the original user or a group user) wishes to check the integrity of shared data, she first sends an auditing request to the TPA. After receiving the auditing request, the TPA generates an auditing message to the cloud server, and retrieves an auditing proof of shared data from the cloud server. Then the TPA verifies the correctness of the auditing proof. Finally, the TPA sends an auditing report to the user based on the result of the verification . System model includes the cloud server, the third party auditor and users.

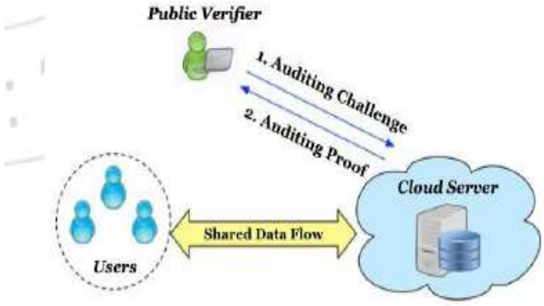

Fig-2: The system model includes the cloud, the user

Existing System:

In existing mechanisms, 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 mechanisms is to 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 or a third party auditor (TPA) who is able to provide verification services on data integrity to users. With shared data, once a user modifies a block, she also needs to compute a new signature for the modified block. Due to the modifications from different users, different blocks are signed by different users. For security reasons, when a user leaves the group or misbehaves, 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.

DISADVANTAGES OF EXISTING SYSTEM:

1.Straightforward method may cost the existing user a huge amount of communication and computation resources.

2.The number of re-signed blocks is quite large or the membership of the group is frequently changing verifier, and user.

Proposed System:

In this paper, we propose Panda, a novel public auditing mechanism for the integrity of shared data with efficient user revocation in the cloud. In our mechanism, 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. As a result,the efficiency of user revocation can be significantly improved, and computation and communication resources of existing users can be easily saved. Meanwhile, the cloud, which is not in the same trusted domain with each user, is only able to convert a signature of the revoked user into a signature of an existing user on the same block, but it cannot sign arbitrary blocks on behalf of either the revoked user or an existing user. By designing a new proxy re-signature scheme with nice properties, which traditional proxy re-signatures do not have, our mechanism is always able to check the integrity of shared data without retrieving the entire data from the cloud.Moreover, our proposed mechanism is scalable, which indicates it is not only able to efficiently support a large number of users to share data and but also able to handle multiple auditing tasks simultaneously with batch auditing. In addition, by taking advantages of Shamir Secret Sharing, we can also extend our mechanism into the multi-proxy model to minimize the chance of the misuse on re-signing keys in the cloud and improve the reliability of the entire mechanism.

ADVANTAGES OF PROPOSED SYSTEM:

1.It follows protocols and does not pollute data integrity actively as a malicious adversary.

2. Cloud data can be efficiently shared among a large number of users,and the public verifier is able to handle a large number of auditing tasks simultaneously and efficiently.

Conclusion:

With our mechanism, the identity of the signer on each block in shared data is kept private from a multiple public verifier, who is still able to publicly verify the integrity of shared data without retrieving the entire file. using multiple public verifier to improve the efficiency of auditing task and also using response Protocol between user and cloud server to protect the user confidential data.

REFERENCES

[1] B. Wang, B. Li, and H. Li, “Public Auditing for Shared Data with Efficient User Revocation in the Cloud,” Proc. IEEE INFOCOM, pp. 2904-2912, 2013.

[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] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable Data Possession at Untrusted Stores,” Proc. 14th ACM Conf. Computer and Comm. Security (CCS’07), pp. 598-610, 2007.

[4] H. Shacham and B. Waters, “Compact Proofs of Retrievability,” Proc. 14th Int’l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT’08), pp. 90- 107, 2008.

[5] C. Wang, Q. Wang, K. Ren, and W. Lou, “Ensuring Data Storage Security in Cloud Computing,” Proc. 17th ACM/IEEE Int’l Workshop Quality of Service (IWQoS’09), pp. 1-9, 2009.

[6] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, “Enabling Public Verifiability and Data Dynamic for Storage Security in Cloud Computing,” Proc. 14th European Conf. Research in Computer Security (ESORICS’09), pp. 355-370, 2009.

[7] 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.

[8] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S.S. Yau, “Dynamic Audit Services for Integrity Verification of Outsourced Storages in Clouds,” Proc. ACM Symp. Applied Computing (SAC’11), pp. 1550-1557, 2011.

[9] C. Wang, Q. Wang, K. Ren, and W. Lou, “Towards Secure and Dependable Storage Services in Cloud Computing,” IEEE Trans. Services Computing, vol. 5, no. 2, pp. 220-232, Jan. 2012.

[10] Y. Zhu, G.-J. Ahn, H. Hu, S.S. Yau, H.G. An, and C.-J. Hu, “Dynamic Audit Services for Outsourced Storages in Clouds,” IEEE Trans. Services Computing, vol. 6, no. 2, pp. 227-238, Apr.-June 2013.

[11] N. Cao, S. Yu, Z. Yang, W. Lou, and Y.T. Hou, “LT Codes-Based Secure and Reliable Cloud Storage Service,” Proc. IEEE INFOCOM, pp. 693-701, 2012.

[12] J. Yuan and S. Yu, “Proofs of Retrievability with Public Verifiability and Constant Communication Cost in Cloud,” Proc. ACM Int’l Workshop Security in Cloud Computing (ASIACCS-SCC’13), pp. 19- 26, 2013.

[13] H. Wang, “Proxy Provable Data Possession in Public Clouds,” IEEE Trans. Services Computing, vol. 6, no. 4, pp. 551-559, Oct.- Dec. 2013.

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

[15] S.R. Tate, R. Vishwanathan, and L. Everhart, “Multi-User Dynamic Proofs of Data Possession Using Trusted Hardware,” Proc. Third ACM Conf. Data and Application Security and Privacy (CODASPY’13), pp. 353-364, 2013.