Multi-User Collaborative cloud Application With Privacy Protection And Access Control

B.HEMA Assistant Professor / IT
M.DILIPKUMAR , J.YOKESH ,V.MOHAN , B.Tech / IT
Velammal Insitute Of Technology,Chennai

Email: [1]baluhema@gmail.com, , [2]dilipkumarmeera@gmail.com

, [3]yokesh333@gmail.com , [4]vmohanlee@gmail.com

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. 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 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, 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 — **Cloud computing, authentication protocol, privacy preservation, shared authority, universal compos ability.**

INTRODUCTION

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 problemwe 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 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 [7], 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 resignatures 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 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.

RELATED WORK

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. In such situation cloud will directly access those data and resign and assign to the existing user in the group 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. Communication and computation cost is high the re-computation of any signature during user revocation should not affect the most attractive property of public auditing

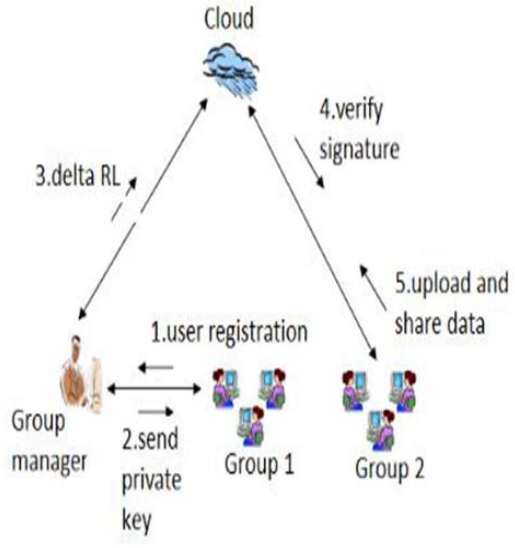
In this paper, we propose Panda, a novel public auditing mechanism for the integrity of shared data with efficient user revocation in the cloud. In our mechanism, by utilizing the idea of proxy re-signatures, once a user in the group is revoked, the cloud is able to resign the blocks, which were signed by the revoked user, with a re-signing key

Efficiency of user revocation can be significantly improved 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

Real Time Example:

In an Group file sharing environment if an user wishes to revocate from a group then the complexity added to the files shared by that user where someone else in the group need to take authority over their files by downloading and reassigning key to that file. In order to overcome that we appoint an third person where his work is to monitor the files of the revocated user and reassign it to someone else in the group based on owners priority without any overhead of download. Here we generate private and public key based on the prime no.The main aim of this paper is to search for private and public files. In case of public files users can modify their files and update to it.

Architecture Diagram



File Upload

File owner allowed to upload data on the cloud either for their private or public use. They act as an Group Manager for the file they upload in cloud. Both the original user and group users are able to access, download and modify shared data. Shared data is divided into a number of blocks. A user in the group can modify a block in shared data by performing an insert, delete or update operation on the block.

File Auditing

If an user edited an data then the auditor will monitor the user and report to the owner about the edited data. The group manager will monitor the changes in the file and if he founds any discrepancy auditor has full rights to revocate from his particular group. The public verifier can audit the integrity of shared data without retrieving the entire data from the cloud, even if some blocks in shared data have been re-signed by the cloud.

Re-assigning

On one hand, once a user is revoked from the group, the blocks signed by the revoked user can be efficiently re-signed. More specifically, the proxy is able to convert a signature of Alice into a signature of Bob on the same block. Meanwhile, the proxy is not able to learn any private keys of the two users, which means it cannot sign any block on behalf of either Alice or Bob.

Group Sharing

Data owner will store their data in the cloud and share the data among the group members. Who upload the data have rights to modify and download their data in the cloud. He can also set rights to other users in his group to edit or download data.

Access control

Cloud Server allows only the authorized group member to store their data in the cloud offered by cloud service providers as Sass and it won't allow unauthorized group member to store their data in the cloud.

User Revocation

If a user wishes to revoke from a group their request regarding revocation will be forwarded to the auditor where auditor will check to it and revoke the user from group. The user revocation is secure because only existing users are able to sign the blocks in shared data. even with a re-signing key, the cloud cannot generate a valid signature for an arbitrary block on behalf of an existing user. In addition, after being revoked from the group, a revoked user is no longer in the user list, and can no longer generate valid signatures on shared data.

SYSTEM MODEL

A system model for the cloud storage architecture, which includes three main network entities: users , a cloud server , and a trusted third party.

• **User:** an individual or group entity, which owns its data stored in the cloud for online data storage and computing. Different users may be affiliated with a common organization, and are assigned with independent authorities on certain data fields.

• **Cloud server:** an entity, which is managed by a particular cloud service provider or cloud application operator to provide data storage and computing services. The cloud server is regarded as an entity with unrestricted storage and computational resources.

• **Trusted third party**: an optional and neutral entity, which has advanced capabilities on behalf of the users, to perform data public auditing and dispute arbitration.

In the cloud storage, a user remotely stores its data via online infrastructures, platforms, or software for cloud services, which are operated in the distributed, parallel, and cooperative modes. During cloud data accessing, the user autonomously interacts with the cloud server without external interferences, and is assigned with the full and independent authority on its own data fields. It is necessary to guarantee that the users' outsourced data cannot be unauthorized accessed by other users.

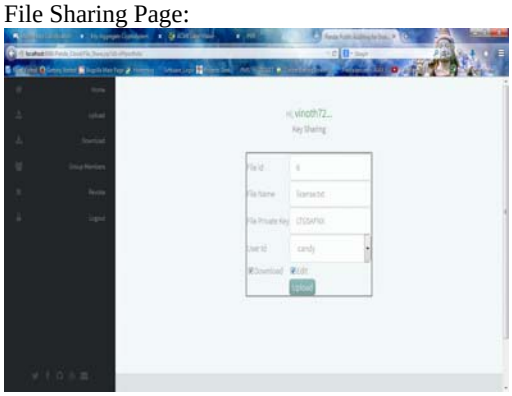
Algorithm

Algorithm 1 Controller Algorithm for CBS

1: Provide initial state z_0^m, x_0^{mk}, $t \leftarrow 0$
2: **loop**
3: At beginning of control period t:
4: Predict $N_{t+i|t}^k$, $p_{t+i|t}$ for horizons $t = 1, \ldots, W$ using a demand prediction model
5: Solve $DCP - RELAX$ to obtain $\delta_{t+i|t}^m$,$\sigma_{t+i|t}^{mk}$ for $i = 0, \cdots, W-1$
6: Sort new containers based on their utilities
7: **for** $m \in M$ **do**
8: Select $z_{t|t}^m$ machines of type m as active machines
9: **end for**
10: Compute a re-packing configuration for all selected active machines
11: Turn on selected machines, perform re-parking using FF, turn off other machines
12: $t \leftarrow t+1$
13: **end loop**

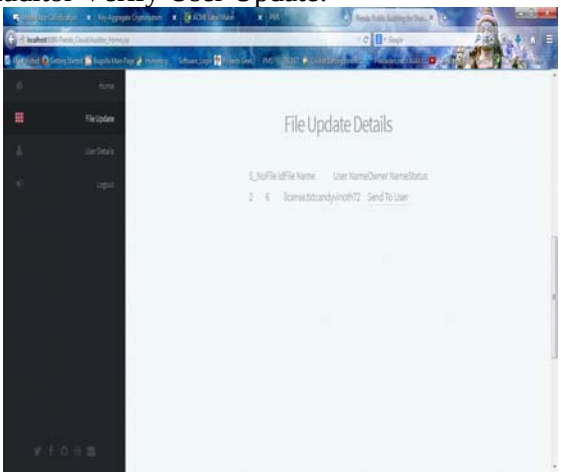
EXPERIMENTAL RESULTS

File Sharing Page:

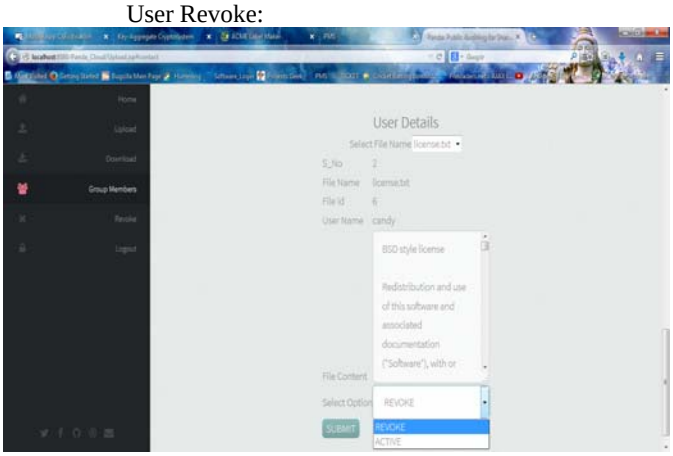


User Access Check:

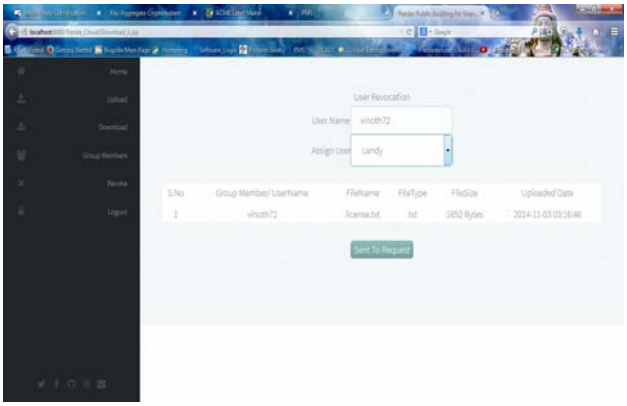
Auditor Verify User Update:



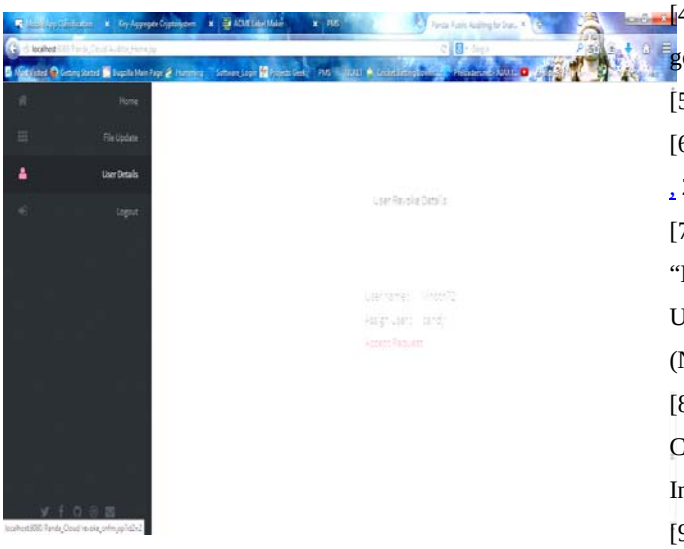
User Revoke:



Group Revocation:



Auditor Approval:



Conclusion

Dynamic capacity provisioning has become a promising solution for reducing energy consumption in data centers in recent years. However, existing work on this topic has not addressed a key challenge, which is the heterogeneity of workloads and physical machines. we first provide a characterization of both workload and machine heterogeneity found in one of Google's production compute clusters. Then we present Harmony, a heterogeneity-aware framework that dynamically adjusts the number of machines to strike a balance between energy savings and scheduling delay, while considering the reconfiguration cost. Through experiments using Google workload traces, we found Harmony yields large energy savings while significantly improving task scheduling delay.

Future Enhancement

In future we can had additional performance category to reduce the energy flow and to measure the saved energy and lost energy.

REFERENCES

[1] Amazon Elastic Computing Cloud, http://aws.amazon.com/ec2/, 2013.

[2] Energy Star Computer Server Qualified Product List, energy star. gov/ia/products/prod_lists/enterprise_servers_prod_list.xls, 2014.

[3] Eucalyptus community, http://open.eucalyptus.com/, 2014

[4] Googleclusterdata - traces of google workloads, http://code.google.com/p/googleclusterdata/, 2014.

[5] Technology research - Gartner Inc, www.gartner.com, 2014.

[6] U.S. Energy Information Adminstration, http://www.eia.gov/, 2014.

[7] G. Ananthanarayanan, A. Ghodsi, S. Shenker, and I. Stoica, "Effective Straggler Mitigation: Attack of the Clones," Proc. 10th USENIX Conf. Networked Systems Design and Implementation (NSDI), 2013.

[8] R. Boutaba, L. Cheng, and Q. Zhang, "On Cloud Computational Models and the Heterogeneity Challenge," J. Internet Services and Applications, vol. 3, pp. 77-86, 2012.

[9] G.E.P. Box, G.M. Jenkins, and G.C. Reinsel, Time Series Analysis, Forecasting, and Control. Third ed., Prentice-Hall, 1994.

[10] S. Boyd et al., Convex Optimization. Cambridge Univ. Press, 2004.

[11] C. Chekuri and S. Khanna, "On Multi-Dimensional Packing Problems," Proc. Symp. Discrete Algorithms, 1999.

[12] Y. Chen, A. Das, W. Qin, A. Sivasubramaniam, Q. Wang, and N. Gautam, "Managing Server Energy and Operational Costs in Hosting Centers," ACM SIGMETRICS Performance Evaluation Rev., vol. 33, pp. 303-314, 2005.

[13] Y. Chen et al., "Analysis and Lessons from a Publicly Available Google Cluster Trace," Technical Report UCB/EECS-2010-95, 2010.

[14] J. Diaz et al., "A Guide to Concentration Bounds," Handbook on Randomized Computing. Springer, 2001.