iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** IV **Month of publication:** April 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Security Of Cloud Data With Logging Using Jar File

Bhagyashree Shendkar[1] Sankalp Bhagat[2] Rohan Rane[3] Monali khandare[4]
Department of Computer Engineering, Sinhgad Institute of Technology and Science, Pune

Abstract— Now-a-days, Cloud Computing technology is growing at a very fast pace. Currently, most of the people are using Cloud Computing technology. Cloud computing allows extremely scalable services to be consumed at ease over the Internet associated on an as-needed basis. Most important feature of the cloud services is that users' data are sometimes processed remotely in some unknown machines that the users don't own or operate. But with all the convenience brought by this new rising technology, fear of users of losing the control over their own data (for instance, financial and health data) is becoming a significant barrier to the wide adoption of cloud services. To unravel the problem above in this paper, we offer effective mechanism of using Cloud Information Accountability framework to keep the track of actual usage of users' data within the cloud. Specifically, we propose associate object-centered approach that allows enclosing our logging mechanism together with users' data and the policies. Accountability is none but checking of authorization policies and it is vital for transparent data access. We provide a mechanism of automated logging using JAR programming which improves privacy and security of data in the cloud. We also provide a distributed auditing mechanism in order to strengthen the user's control over their data. We also provide problems of a user along with solutions that demonstrate the efficiency and effectiveness of our proposed system.
Keywords—cloud computing, logging mechanism, auditing ability, accountability, data sharing.

I. INTRODUCTION

Cloud computing is a technology that uses internet and remote servers to store information and application. In cloud, it does not have to be compelled to install specific hardware, software package on user machine, therefore user will get the specified infrastructure on his machine in low rates. Cloud enables highly scalable services to be easily consumed over the Internet on as-needed basis. It provides computation, software data access and storage services that do not require end-user knowledge and physical location and configuration that derives service. Conventional access management approaches developed for closed domains like databases and in operating systems, or approaches employing a centralized server in distributed environments, don't seem to be appropriate, attributable to the subsequent options characterizing cloud environments. There are possibilities that data can be outsourced by the direct cloud service provider (CSP) to other entities in the cloud and these entities are able to allocate the task of data management to others and so on. Being flexible in nature entities are allowed to join and leave cloud on their wish. This causes data handling takes place through a complex and dynamic hierarchical service chain which does not exist in conventional environments. To overcome the above stated problems, a new approach called Cloud Information Accountability (CIA) framework was proposed, based on the concept of information accountability. This CIA framework provides end-to-end accountability in a highly distributed fashion. By means of the CIA, data owners can track whether the service-level agreements are being honored, but also make obligatory access and usage control rules as needed. The design of the Cloud Information Accountability framework presents substantial challenges, together with unambiguously distinctive CSPs, guaranteeing the dependableness of the log, adapting to an extremely decentralized infrastructure. The main and basic approach towards addressing these problems is to leverage and extend the programmable capability of JAR (Java Archives) files to mechanically log the usage of the users' data by any entity within the cloud. Users can send their data together with any policies like access control policies and logging policies that they need to enforce, enclosed in JAR files to the cloud service providers. Any access to the data can trigger an authenticated and automated logging mechanism which is local to the JARs. Such type of enforcement can be referred as "strong binding" as the policies as well as the logging mechanism travel with the data. This strong binding remains in existence even if the copies of the JARs are created. Thus, the user can have control over his data at anywhere. This decentralized logging mechanism not only satisfies the dynamic nature of the cloud but imposes challenges also by ensuring the integrity of the logging. To cope up with this issue, we have provided the JARs having a central point of contact which can form a link in between them and the user. It also records the information of the error correction sent by the JARs that allows it to keep monitoring the loss of

any logs from any of the JARs. In addition to this, if the JAR cannot contact its central point, the access to its enclosed data will be refused. At present, we focus on image files as the images represent a very common file type for the end users and the organizations (as it is proved by the popularity of Flickr and are increasingly hosted in the cloud as a part of the storage services offered by the utility computing paradigm featured by cloud computing. Further, images usually reveal social and personal habits of the users, or are used for archiving important files from organizations. Moreover, our approach can handle personal identifiable information provided that they are stored as image files (they contain an image of any textual content, for example, the SSN stored as a .jpg file).

II. PROBLEM STATEMENT

We have illustrated an example below which is thoroughly concerns to the problem statement and will be used throughout the paper demonstrating the main features of our system.
Example: A professional photographer named Slim Aarons, wants to sell his photographs using cloud services.

For the business in the cloud he needs following requirements:

A. Only paid users can be able to download the photographs uploaded by Slim on his cloud.

B. The pictures can't be allowed to be downloaded by the potential users until and unless the payment is done.

C. As some of his work is related to the particular regions, only users belonging to those specific regions are allowed to view and download the photographs.

D. Some photographs can be viewed for a limited time from a particular user so as his work will not be reproduced easily.

E. If in case any dispute gets arisen with a client, he wants to have all the privileges to access the information of the client.

F. He wants to make sure that the cloud service provider would not share his photographs with other services providers so as to expect the accountability provided for individual users from the cloud service provider.

By keeping in mind the above scenario, the common requirements have been identified by us and for achieving the data accountability in cloud, we have developed some guidelines. When a user gets subscribed to a particular cloud service, he/she always needs to send his/her data along with the associated access control policies if provided by the provider. After receiving the data, cloud service provider grant the access rights to the user such as read, write, copy on the data. Using conventional access control mechanism, once the access rights gets granted, the entire data will entirely be available to the service provider. Thus for tracking the data and its usage, we've aimed to develop logging and auditing techniques which satisfy the requirements below:

A. The logging should be decentralized in order to adapt to the dynamic nature of the cloud. More specifically, log files should be tightly bounded with the corresponding data being controlled, and require minimal infrastructural support from any server.
B. Every access to the user's data should be correctly and automatically logged. This requires integrated techniques to authenticate the entity who accesses the data, verify, and record the actual operations on the data as well as the time that the data have been accessed.
C. Log files should be reliable and tamper proof to avoid illegal insertion, deletion, and modification by malicious parties. Recovery mechanisms are also desirable to restore damaged log files caused by technical problems.
D. Log files should be sent back to their data owners periodically to inform them of the current usage of their data. More importantly, log files should be retrievable anytime by their data owners when needed regardless the location where the files are stored.
E. The proposed technique should not intrusively monitor data recipients' systems, nor it should introduce heavy communication and computation overhead, which otherwise will hinder its feasibility and adoption in practice.

III. Cloud Information Accountability

The Cloud Information Accountability is nothing but a framework which is used to conduct automated logging and distributed auditing of access performed by any entity, carried out at any point of time at any cloud service provider. There are two

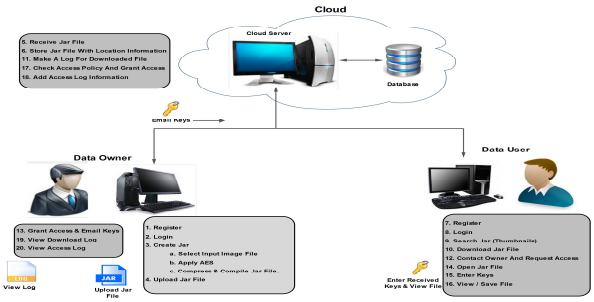
components, one is Logger and other is Harmonizer.

The logger is used to include automatically logging of the information of the access to the data as logger is totally coupled with the data of the user. Thus, whenever there is access to the data for the data to be modified or deleted, the logger automatically sends the logs to the data owner providing the data is secured.

There are three main components in the cloud information accountability. They are Cloud Server, Data Owner and the Data User.

Data owner first register to the cloud server for creating the Jar.

Then the compiled Jar file is then uploaded to the cloud server. When the server receives the compiled file, it stores that JAR file with its location information. In order for a data user to download the JAR file, he has to be registered and



logged in to the cloud server and has to search the JAR. As soon as the file gets downloaded, its log is automatically gets created. Then in order to access the JAR file, data user has to contact and request the data owner. Data owner then grant the request and mail the data user the keys which are generated randomly. Data user then open the JAR file by entering the keys and can save or view the data which was encrypted in the JAR file. As soon as the data user saves the contents in the JAR, an access log and the download log with location is created in the cloud server which can be further viewed by the data owner.

IV. LOGGING MECHANISM

This section describes the distributed auditing mechanism which includes the algorithms for data owners for querying the logs regarding their data.

A. Push Mode

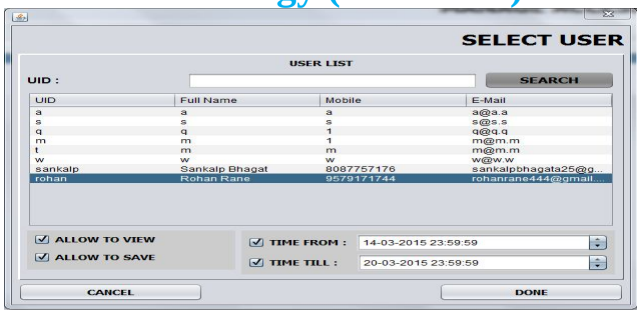
This auditing method for getting the logs according to teh erequirement. In this mode, the logs are pushed to the data owner or auditor by the log harmonizer automatically and periodically. Whenever the data is tried to be accessed, inserted, deleted or modified, all the information about each access to data is written in the log along with the location where the access is done and sent reliably to the data owner.

B. Algorithms

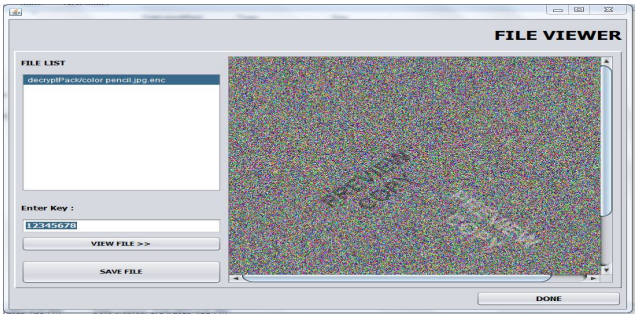
For the purpose of encryption and compression and hashing, three different algorithms AES (Advance Encryption Standard), compression algorithm and SHA (Secure Hash Algorithm) algorithms are used respectively. AES algorithm is used for encryption and decryption using the same key. It is also known as the symmetric key algorithm. SHA algorithm is used to encrypt and decrypt data using password. The compression algorithm is used to compress the data in the JAR file.

V. IMPLEMENTATION

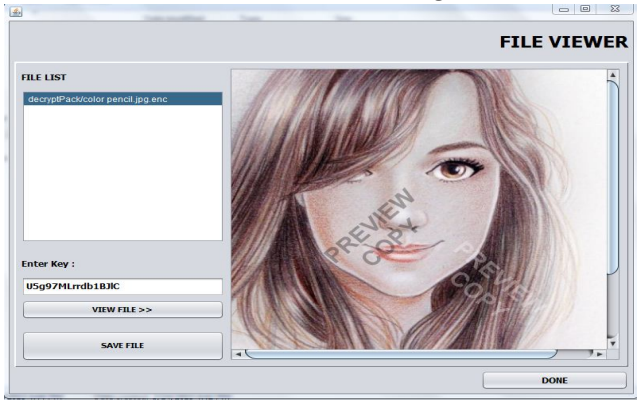
The photographer who has uploaded the photo will give the privileges to the particular user among all users to view and save the file. And also he can set the timing only during which the photo could be downloaded.



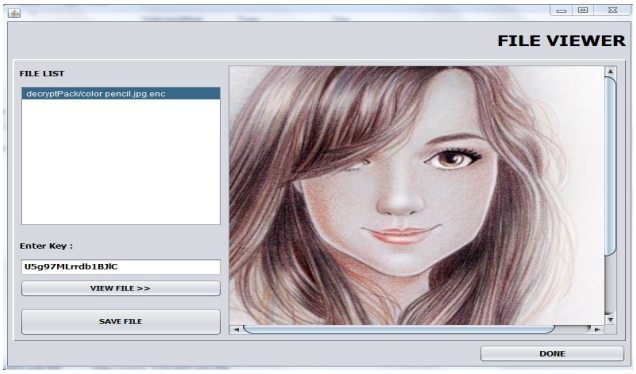
Until and unless the date user can not be given the key to open the image encrypted in jar file, the data user cannot be able to view photos.



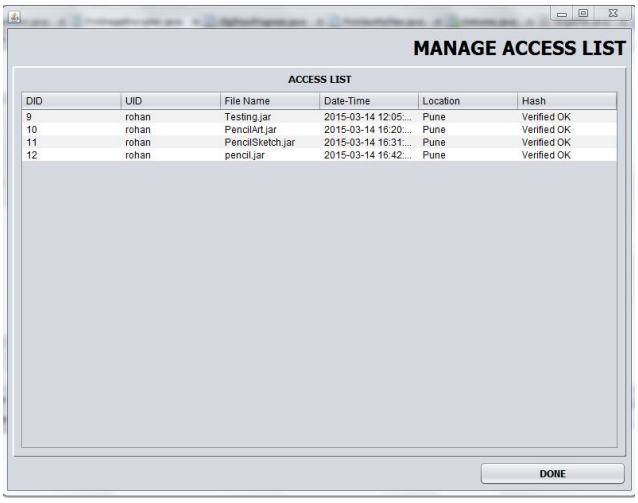
Data user will then request the data owner i.e up loader of the jar for the key. After data user gets an e-mail containing keys he can view the photo. If he tries to take the screenshot he cannot do that as the image contains the watermark.



Data user need to be privileged the access to save the file by the data owner. Only then he can get the original image as follows.



Once the data user downloads the file, an automatic data log gets created showing at what time and which date a user has downloaded the photo including his location.



VI. CONCLUSION

This paper presents effective mechanism that performs automatic authentication of users and make automated log records of every information accessed by the user along with its location and then send the logs to the data owner. Data owner will audit his content on cloud and get the confirmation that his information is safe on the cloud. Data owner additionally will be able to recognize the duplication data of information created while not his data. Data owner must not get anxious concerning his knowledge on cloud exploitation. Data owner should not worry about his data on cloud using this mechanism as the data usage is transparent.

REFERENCES

[1] Smitha Sundareswaran, Anna C. Squicciarini,"Ensuring Distributed Accountability for data sharing in the cloud,"ieee transactions on dependable and secure computing vol.9 no.4 year 2012

[2] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing," Proc. European Conf. Research in Computer Security(ESORICS), pp. 355-370, 2009.

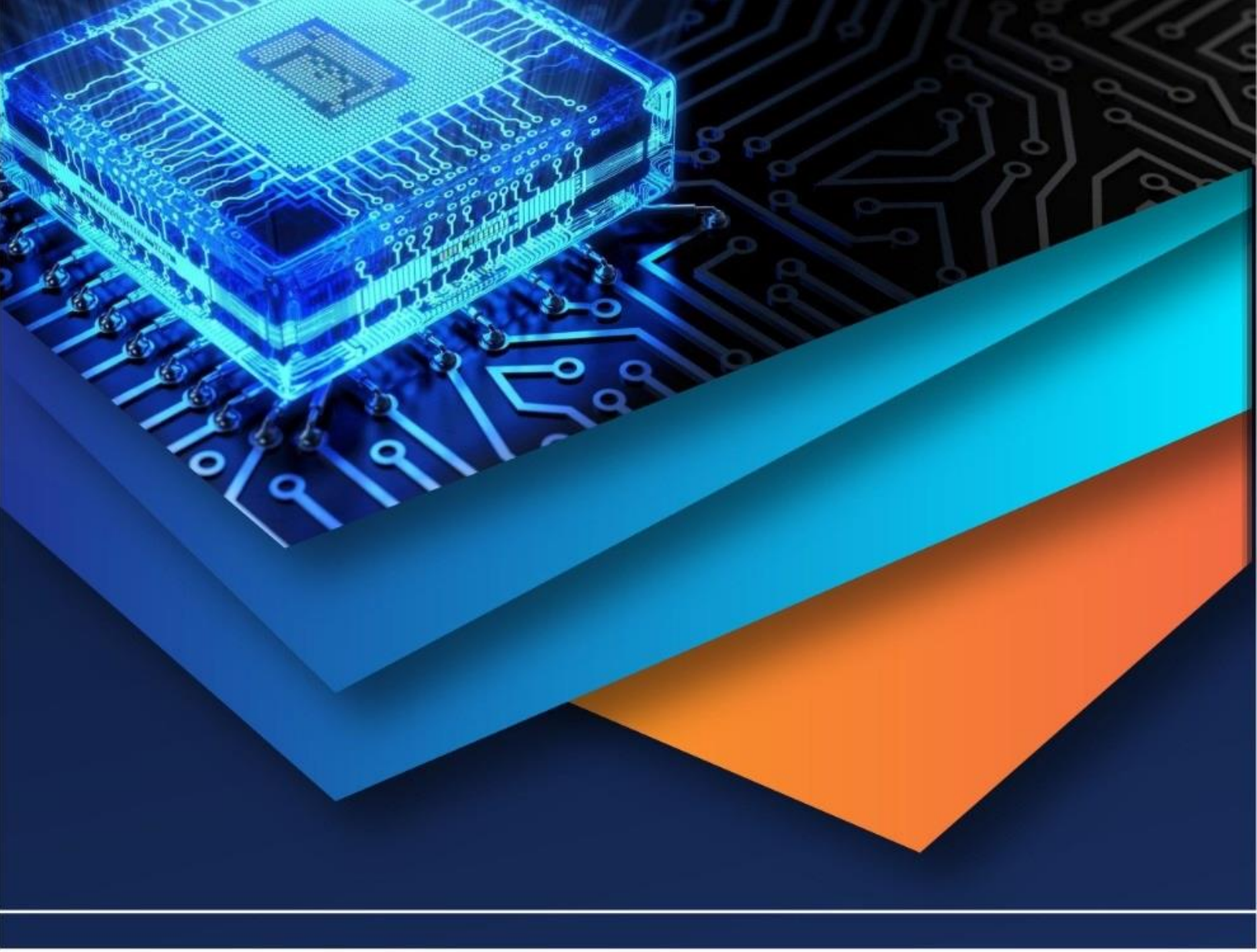
[3] D. Boneh and M.K. Franklin, "Identity-Based Encryption from the Weil Pairing," Proc. Int'l Cryptology Conf. Advances in Cryptology,pp. 213-229, 2001.

[4] D.J. Weitzner, H. Abelson, T. Berners-Lee, J. Feigen-baum, J. Hendler, and G.J. Sussman, "Information Accountability," Comm.ACM, vol. 51, no. 6, pp. 82-87, 2008.

[5] J. Hightower and G. Borriello, "Location Systems for Ubiquitous Computing," Computer, vol. 34, no. 8, pp. 57-66, Aug. 2001.

[6] T.J.E. Schwarz and E.L. Miller, "Store, Forget, and Check: Using Algebraic Signatures to Check RemotelAdministered Storage," Proc. IEEE Int'l Conf. Distributed Systems, p. 12, 2006.

[7] J.H. Lin, R.L. Geiger, R.R. Smith, A.W. Chan, S.Wancho,"Method for Authenticating a java Archive (jar) for Portable Devices", US Patent 6,766,353, July 2004.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY