International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 7 4467 - 4470

Distributed Accountability Technique for All File Types

[1]Priyanka More
[1]PG.Fellow, Dept of computer Engg.
G.H.R.C.E.M,
Pune, India
e-mail: pmore1899@gmail.com

[2]Prof. Mayuri Lingayat
[2]Assistant Professor, Dept of computer Engg.
G.H.R.C.E.M,
Pune, India
e-mail: smile.mayuri@gmail.com

***Abstract*—** Cloud computing is most excellent emerging paradigm in computer industry which hides the details of cloud services from cloud user. Basically users may not know the machines which truly process,host their data and also that cloud data is outsourced to other entities on cloud which cause issues related to accountability. So it is very important to develop such approach which allows owners to keep track of their personal data over the cloud. To solve all security related issues raised on cloud we propose a Cloud Data Security as well as Accountability (CDMA) method which is based on Information Accountability. This allows owner to keep track of all usage of data over cloud. Such method is applied to generic files in our proposed concept. Also it can increases security level. Basically in our approach main focus is on generic files, size of jar, time for uploading and downloading jar.

Keywords- *Accountability, Cloud computing, Logging, Data sharing, Security.*

*****

I. INTRODUCTION

Cloud computing is the technology, which will provide data storage capability and access over the internet. It provides various cloud services like Amazon, Google, Microsoft and their delivery to the multiple users. It will dynamically deliver information, resources as services over the Internet. Cloud computing service providers offer their services according to three fundamental models that are IaaS, PaaS, SaaS known as Infrastructure, platform and software as a service. All these allow user to run their applications and to store data online. SaaS allows user to run existing online applications, PaaS allows to create their own cloud applications using different tools and languages, and Iaas allows to run any application they please on cloud hardware of their own choice. Popularity of this cloud technology is increasing rapidly in distributed computing environment. Large access to the data, resources, hardware without installing any software is one of the main feature of cloud computing. Multiple clients can access the cloud data from anywhere in the world. While enjoying such benefits brought by this technology, sometimes users also start worrying about losing control of their personal data. The data processed on clouds are generally outsourced that leads to a Number of issues related to accountability. Data becomes public hence security issues increases towards private data of users. Hence, effective mechanism is required for all users to monitor the usage of their data. Previously, some access control approaches are already developed for databases and operating systems, but that are not suitable in this situation, because of some features like information is outsourced by the cloud service provider (CSP) in the cloud, entities are allowed to join and leave the cloud in a flexible way. So as a result, all the data handling in cloud goes through a complex hierarchical service chain which does not exist in conventional environments. To overcome above problems, cloud information accountability mechanism is introduced here. By using such technique other peoples cannot read the owners data without having access to that data so owner should not bother about his personal data, and should not worry about destruction of his data by hackers.

A. What is Accountability?

In this paper, Cloud data accountability framework is introduced which does automatic Logging and distributed auditing mechanisms. Accountability is required for monitoring usage of cloud data also it provides customers with transparency as well as control over their data available on cloud environment. In our distributed accountability and auditing technique, data owner will set the policies for his data first, which he wants to place on cloud and send it to service provider of cloud enclosed in JAR files, every access to owners information will be automatically checked for its authentication and logs record for each document will be formed and that logs periodically sent to owner for monitoring the data usage. It has logger component and log harmonizer component. Logger component is coupled with data and that is copied when particular data are copied. It handles instance of data and also responsible for logging access to that instance. Every log entry is encrypted before appending to log record. The log harmonizer perform auditing. It supports two modes push and pull. Push mode shows that logs being periodically sent to the owner and Pull mode refer to an alternative approach where the user can retrieve the logs as on need basis. This data accountability framework presents substantial challenges, including uniquely identifying cloud service providers, defining the reliability of the logs etc. Our main approach toward addressing such issues with the help of JAR

files means JAVA Archive which is used to keep all the documents in only one folder by compressing their length in this way we are saving the memory by using JAR files. Such files contains set of access rules that tells us that whether and how the cloud servers other entity's are allowed or authorized to access the documents. After Completion of all authentications, the cloud service provider will get access to access the data from the JAR file. Currently, focus is on generic files contain all types of files where this distributed accountability applied to all data types.

II. LITURATURE SURVEY

In this section we focus on some related techniques which define security related issues in cloud data storage. This section gives the review of existing techniques, their limitations and solutions on that. Dan Lin, smitha introduced automatic logging mechanism. First time, authors\defined data accountability concept through JAR files. They applied Data accountability concept for images not for all files types. Hence it is necessary to implement that framework for all type of files. Our proposed system gives implementation for that. Proposed framework is platform independent means it does not required any dedicated storage system or authentication in place but multiple inner jars takes so much time to execute and latency is noticed by data users [1].

Q Wang proposed a Protocol known as dynamic auditing protocol which supports the dynamic operations of the users data on the cloud servers. Authors work studies the difficulty of defining the integrity of storage data in Cloud Computing. Authors consider the task of allowing a third party auditor on behalf of the cloud client, to validate the reliability of the dynamic data which is stored on the cloud. Basically their prior works on to ensure remote data reliability which often lacks the support of public auditability or dynamic data operations, their paper achieves both the things. But this technique may disclose the data content to the auditor because it requires the cloud server to continuously send the linear combinations of data blocks to the auditor. Authors presented TPA scheme in cloud computing using RSA algorithm and Bilinear Diffie-Hellman techniques[3].

R. Corin design a logic that allows agents to prove their action as well as authorization to use particular cloud data. In this technique data owner add various policies with their data that contain a detail description of which actions are allowed with which documents. Agents audited by particular authority at arbitrary moment in time. But there is the big problem of Continuous auditing of agent. But there work has also provided solution that monitors incorrect behavior of agent and agent has to give detail explanation for their actions, after that authority will check the justification [10]. D. Boneh proposed a fully functional identity-based encryption scheme (IBE). Their method has chosen ciphertext security in the random oracle model by assuming an elliptic curve variant of the computational Diffie-Hellman problem. Our proposed system is based on bilinear maps between groups. The Weil pairing on elliptic curves is an example of this map[11].

Hsio Lin defines one approach known as A Secure Erasure Code-Based Cloud data Storage System with Secure Data Forwarding. Threshold proxy re-encryption method is proposed and integrates it with a decentralized erasure code such that a secure distributed data storage system is formulated. This distributed data storage system not only supports secure and robust cloud data storage and retrieval, but also allows a user forwarding of his data on the cloud servers to another user without retrieving the data back. The main contribution is that the proxy re-encryption idea which supports encoding operations over encrypted messages as well as forwarding operations over encoded and encrypted messages. This scheme integrates encrypting, encoding, and forwarding [2].

From above study of related papers we are motivated by some points which are required to implement like the cloud data users logging should be decentralized to adapt to the dynamic nature of the cloud environment. Each and every access to the cloud data should be appropriately and automatically logged so that data integrity can be verified. Distributed account-ability mechanism should apply to all types of files. Several recovery techniques are also desirable to restore damaged files caused by technical problems. It is necessary to use strong encryption decryption schemes.

III. PROBLEM STATEMENT

To propose a highly data accountability mechanism to keep track of the actual usage of the owner's data in the cloud to meet the Authentication, accountability and auditability requirement for data sharing on cloud, which is applied to generic file types contains all file types and also it supports a variety of security policies, by considering size, time attributes.

IV. PROPOSED SYSTEM

As we know that, data process on clouds lead to a number of issues mostly related to accountability, so it is essential to provide a effective technique which monitors all the usage of the data on cloud. There are some schemes which already available as we discuss in related work but they have some limitations. So here we are trying to reduce those limitations by implementing distributed Accountability with Jar files. Data can be securely share on cloud is the main intention of this system.

In the existing system, this technique is only applied to only images. But our proposed scheme apply this framework to generic file types which contains all file types, also it supports a multiple types of security policies, like indexing policies for text files, usage control for executable files. In existing system, they directly upload image file with Jar but in our idea we uploads two files: original file as well as demo file. If a nyone has only viewing rights means no saving rights t en our system gives only demo file not original file to them. So it pres erves the privacy. Also they mentioned pure log idea wh ich is hypothetical, in our work we analyze it by connecting directly to server. The complete flow of our proposed work is shown in next section. In proposed system our main contributions are:- Propose data accountability techniquue with the help of JAR

 ISSN: 2321-8169
4467 - 4470

files. -Implement data accountability for all file types. Reduced JAR file size as compared to existing system also reduced time to access files. - Analyze Pure Log concept. We upload two files instead of single file. Provide a certain degree of usage control for protected data after these are delivered to the receiver.

A. Proposed System Flow:

In this section, how proposed system will work is elaborated through the flow diagram shown in fig.1. Basically our whole work is divided into four phases. In the first phase owner creates jar and uploads it on the cloud. Basically here owner not only upload single file but also uploads two files: demo file as well as original file. If someone has only viewing rights not saving rights then demo file will shown to that user not original file. In second phase the user who wants to access that data will request to view that data. In third phase authentication is performed to check whether the user is authentiated user or not. In forth phase access is given to users according to access policies.

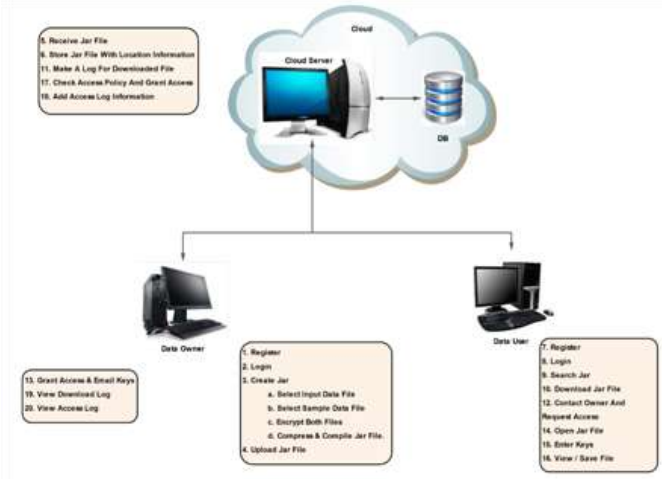

Fig.1:System Fow

If owner want to upload his data on clou d server first step is the data owner should performs registration. After that owner creates jar file. In that creation process he chhooses two files original file as well as demo file. After that owner encrypt his file using strong encryption algorithm. In encryption unique key is generated for every selected document . Once encryption process over, jar compression and compilation is the next step which is performed by owner. Then he uploa ds that jar file on a cloud server. These are few steps which are include d in a jar creation procedure.

After that cloud server receive owners file and stores it with location information on cloud. If data user wants to access data which is available on cloud then he has to first resister himself. Then user search jar file as well as download that file. He contact to owner and make a request to acce ss that file. That time only cloud server makes a log for that downloaded file. Then owner grant access and send key to user through email securely. Once user receives key then only he open that jar file and enter key sent by owner. Then cloud server checks access policies and then only grants access to that user. User decrypts data by using key. Whenever user use owner's data that time it generates log with time, type, location and hash attributes. Then these decryption logs store on the cloud server means server add access log information to log file. Access log information and downloaded logs are periodically sent to data owner.

Hence all the actions performed by clients will informed to data owner with the help of these log records. Like this way data owner upload his data on cloud and user use such data in securely manner with the help of our proposed system approach. In Our system, we can add any type of data to a jar file means not only images but also videos and all other file type also. We use strong encryption and decryption for jar content to avoid spoofing attack.

B. Automated Logging method

- **The Logger structure:**

JAR file consists of one external JAR enclosing one or more internal JARs. The main task of the outer JAR is to handle authentication of users which want to access the data stored in the JAR file. It supports 1) Pure Log technique: Its main task is to record every access to the owners data. Log files are used for auditing reason.2) Access Log technique: defines actions of logging and enforcing access control. In case an data access request is denied, the JAR will record the exact time when the request is made. If the data access request is granted, the JAR will record the access information along with the duration for which the access is permitted.

- **Creation of Log Record:**

Log records are generated with the help of logger component. Logging occurs at any access to the documents in the JAR, and new log entries are appended serially, in order of creation LR={r1, r2…rk }. Each record ri is encrypted separately and appended to the log file. In particular, log record defined as:

ri = **(ID,Action,Tm,Locn, h((ID,Action,Tm,Locn)| ri – 1|. ..|r1),sig)** Here, **ri** defines that an entity identified by ID has performed an action Action on the user's data at time Tm at location Locn. The component **h((ID,Action,Tm,Locn)| ri – 1|. . . |r1)** related to the checksum of the records preceding the newly inserted one, which is concatenated with the main content of the record .The checksum is computed with the help of collision-free hash function. The time of access is determined using the Network Time Protocol (NTP) to avoid suppression of the accurate time by a malicious entity.

- **Retrieval of Logs:**

We described auditing technique including algorithm for owner to query the logs related to their data. The algorithm performs logging as well as synchronization steps with the harmonizer in case of Pure Log. First, the algorithm checks whether the size of the JAR has exceeded a fixed size or the normal time between two consecutive dumps has elapsed. The size as well as time threshold for a dump are specified by the data owner at the time of JAR creation. Also this algorithm Checks whether the owner have requested a dump of the log files on not. If none of these actions occurred, it proceeds to

 ISSN: 2321-8169
4467 - 4470

encrypt the record and write the error-correction information to the log harmonizer.

V. EXPERIMENTAL RESULTS

In Fig.2, It shows that the size of logger component which means the size of created JARs by varying the size and number of documents held by them. After creation and compilation of JAR we analyze it by how many percent jar size will increases after insertion of data. This is the first result of our system which is compared with existing system result.

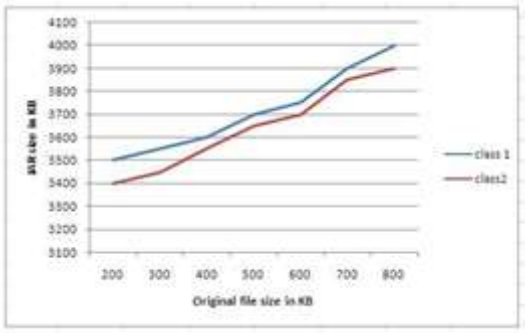

Fig.2: Size of the logger component

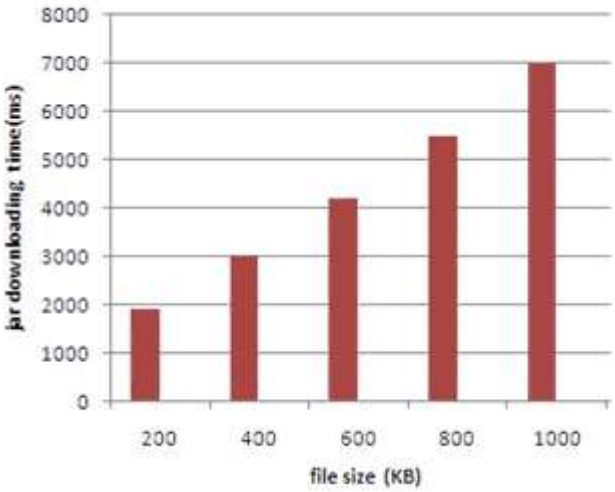

Fig. 3:Time for downloading of jar

In fig3, how much time required for downloading of jar is shown. As per results minimum time is required for that.

VI . CONCLUSION

In this paper we introduced new methods for automatically login with auditing mechanism. This method allows the owner of the data to not only audit his data items but also impose strong back-end security. This framework is applied to image data type as well as generic file types including all file types and also it supports variety of security policies. In case of damaged file data caused by technical problems for those recovery techniques are available to restore.

REFERENCES

[1] S.Sundareswaran, Anna C. Squicciarini and Dan Lin, ”Ensuring Dis-tributed Accountability for Data Sharing in the Cloud”, IEEE Transaction on dependable a secure computing, VOL. 9, 2012.
[2] Hsio Ying Lin,Tzeng.W.G, ”A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding”,IEEE transactions on parallel and distributed systems,2012.
[3] Q. Wang, C. Wang, K. Ren, W. Lou and J. Li ”Enabling public auditability and data dynamics for storages security in cloud computing”, IEEE transaction on parallel and distributed systems,2011.
[4] S. Sundareswaran, A. Squicciarini, Dan Lin, and S. Huang, ”Promoting Distributed Accountability in the Cloud,”, Proc. IEEE Intl Conf. Cloud Computing, 2011.
[5] S. Pearson , Y. Shen, ” A privacy Manager forCloud Computing,”, Proc. Int’l Conf Cloud Computing,2009.
[6] S. Pearson and a. Charlesworth, ”accountability as a way forward for privacy protection in the cloud, ”, proc first int’l conf.cloud computing, 2009.
[7]]. W.Lee, A. Cinzia, ”The Design and evaluation of accountable grid computing system,”, Proc. 29thIEEE Intl Conf. Distributed Computing Systems (ICDCS 09),pp. 145-154, 2009.
[8] P.T. Jaeger, J. Lin, and J.M. Grimes, ”Cloud Computing and Information Policy: Computing in a Policy Cloud,”,J. Information Technology and Politics, vol. 5, no. 3, pp. 269-283, 2009.
[9] A. Pretschner, F. Schuo tz, C. Schaefer, and T. Walter, ”Policy Evolution in Distributed Usage Control,”, Electronic Notes Theoretical Computer Science, vol. 244, pp. 109-123, 2009.
[10] R. Corin, S. Etalle, J.I. den Hartog, G. Lenzini, and I. Staicu, ”A Logic for Auditing Accountability in Decentralized Systems,”,Proc. IFIP TC1 WG1.7 Workshop Formal Aspects in Security and Trust, 2005.
[11] D. Boneh and M.K. Franklin, ”Identity-Based Encryption from the WeilPairing,”, Proc. Intl Cryptology Conf. Advances in Cryptology,pp. 213-229, 2001.