MJRET
Open Access
Multidisciplinary Journal of Research in Engineering and Technology
Journal homepage: www.mjret.in
ISSN:2348 - 6953

PROMOTING DISTRIBUTED POLICY ENFORCEMENT AND ACCOUNTABILITY FOR DATA SHARING IN THE CLOUD

Ms. Priyanka S. Shinkar, Prof. Dipa D. Dharmadhikari
Department of Computer Science and Engineering
Marathwada Institute of Technology
Aurangabad, India
priyankashinkar@gmail.com,dipa.dharmadhikari@gmail.com

Abstract: *Cloud Computing is a distributed computing mechanism that focuses on providing a large number of users with visualized hardware and software infrastructure over the internet. In the most legal and commercial transactions, the organization holds the individual accountable transaction is very important to access the data. For this role, we require Accountability concept in the cloud environment. This will check for every action of a user is executed in his account and it will also store the information of that user. Accountability concept is important for safety and security purpose. The main feature is that users' data usually processed on the unknown machine accessing remotely do not own or operate on the unknown machine using cloud services. Cloud computing is highly scalable services to easily processed over the internet on an as useful basis.To address this problem, this is a highly decentralized information accountability framework to maintain track of the actual use of the user data.An object centeredapproach that enables enclosing of our logging mechanism with users' data and policies. Now, we control the JAR programmable capabilities to create a dynamic or travelling object and to make sure that any access to user data will trigger automated and authentication logging local to the JARs. To maximize users control, we provide distributed auditing mechanism for protection.*

Keywords: *Cloud Computing, Logging mechanism, CIA framework, Cloud Service Provider (CSP), Java Archives (JAR).*

1. INTRODUCTION

Cloud Computing is an emerging paradigm in the computer industry that puts entire computing infrastructure hardware and software applications online. It uses the internet and remote, central servers to maintain users' information, data and applications. Cloud computing technology is flexible, highly scalable and provide us technology enables services

that can be easily used over the many Internet applications on as needed basis. Cloud computing presents a new direction to the current use of goods and services and delivery model for the cloud IT services based on the Internet through providing for dynamically scalable and repeatedly virtualized resources as a service over the Internet/Intranet.

Now a day, there are a number of important individual and commercial cloud computing services like Amazon, Google, Microsoft, Yahoo, and Salesforce [7]. Any user may not know the machines where it is actually processed and host their data. While using the technology, user also starts to bother about losing control of his own data [1] [8]. The data running on clouds are often utilized by a number of issues associated with accountability, including the checking of personally identifiable information. Accountability [3] is the obligation to act as a responsible steward of the user personal information to take responsibility for the protection and appropriate use of that information beyond legal requirements, and to be accountable for any exploitation of that personal information.

The cloud provides three service models are Platform as a service (Paas), Infrastructure as a service (Iaas) and Software as a service (Saas). The four sections of the Database as a service is as follows:

1.1 Encryption and Decryption - The best solution for security is the encryption technique which will store data safely in the cloud.

1.2 Key Management - If encryption technique is important for data storing in the cloud, the encryption keys can't be stored, so user requires key management.

1.3 Authentication - For accessing stored data in the cloud by authorized users.

1.4 Authorization – Rights given to the user as well as cloud providers.

To solve the security issues in cloud; the user can't read or write the respective users' data without having his access rights. Data owner does not worry about his data, and should not get hesitate about the harm to his data by hacker; there is a need of security mechanism which will trace where the data is used in the cloud. Accountability is important for monitoring data usage, in this case all actions of users such as sending or receiving of files are cryptographically connected to the server, it will perform them and server uses secured record of all the actions of previous and server can use the earlier records to know the correctness of the action.

It also provides reliable information about the use of data and it checks all the records, so it will help to make relationship and reputation. So accountability is used for verification of authentication and authorization in this paper [9]. It is very important tool to check the authorization policies in the cloud.

2. LITERATURE SURVEY

2.1 Phases of Accountability

Accountability specifies authorization requirement for data usage policies in the cloud to show the services. Accountability mechanisms which display after the fact verification is very attractive to enforce authorization rules/policies [10].

There are seven phases of accountability in the cloud as follows:

a. Policy setting with data
b. Use of data by users
c. Logging
d. Merge logs
e. Error correcting in the log
f. Auditing
g. Rectify and improvement. These phases may change according to the framework.

First step is the data owner will set the rules/policies with data and send it to CSP (Cloud Service Provider), data will be accessed by users only and logs of each record will be created at that time, after that log will be clubbed together and error correction in the log file has been done and in auditing logs are checked and in last phase improvement has been done [11]. All steps are important to perform the next step, accountability is nothing but validation of user actions means user having rights for accessing that data or not. Suppose the user is doing misuse of any data or resources, then network or data owner will take action on it. So, users should not worry about his data for sharing in the cloud.

2.2 Security and Privacy issues in cloud

Cloud computing is a technique which is used for data sharing in the cloud. Basically, this concept is based on security and privacy issues. So, users' data and applications are stored in the cloud. The user purchases that data for a certain amount of time, which is owned and maintained by a third party. Accountability mechanisms are based on privacy concerns with end users and then develop a privacy manager. Privacy manager provides only limited features to the user with limited rights and does not guarantee about protection to once data are being disclosed.

There are seven cloud computing security threats as follows:

a. Abuse and Nefarious Use of Cloud computing
b. Insecure Application Programming Interfaces
c. Malicious Insiders
d. Shared Technology Vulnerabilities
e. Data Loss/Leakage
f. Account, Service and Traffic Hijacking

g. Unknown Risk Profile

2.3 Identity based Encryption (IBE)

A fully functional identity based encryption scheme (IBE) has selected for cipher text security in the random mysql model. This system is based on bilinear maps between a group. The Weil pairing on elliptic curves is an example of a map. This will bind the content with the rules and provide encrypted content and log the files, using a security against chosen cipertext and plaintext attacks. This will simply provide security to our data in the cloud environment [6].

2.4 Self Depending Objects (SDO)

Self-defending objects [12] are newer version of the object-oriented programming standard. In which, Software objects offers useful functions or hold useful data are responsible for caring those data. Similarly, we also expand the concepts of object-oriented programming in SDO. The main purpose of our implementations is that the user still uses centralized database to maintain the access records, while the items being protected or held as separate files. In previous papers, the author provided a Java-based approach to prevent privacy [4] leakage from indexing, which could be integrated with the Cloud Information Accountability (CIA) framework.

2.5 Proof Carrying Authentication (PCA)

The terms of authentication techniques, Appel and Felten [2] proposed the Proof-Carrying authentication (PCA) structure. The PCA contains a high order logic language that allows quantification over predicates, and it will focus on access control for web services. The PCA's goal is highly focused on validating code, not monitoring content.

3. PROBLEM STATEMENT

Now a day, Security is very important for user data. In a cloud environment, a user can secure his data by using setting some policy to the user. Accountability is generally referred by Log. In the existing system, there was problem that more time was required when it generates the Log. Generally, this Log file created with every data suppose if the Cloud subscriber retrieves any data from a cloud service provider then Log file attached to that data. Generally user only required data there will be no need of Log information. If the size of data is large according to the need of user then abruptly the large log file is also attached so for copying that log file more time is required. For merging the log files if the log file size is small then merging time is small but if the size of log file increases then tremendously it affect on merging time.

Generally performance is the ratio of expected output and total load of the system. If load or files are large then CSP performance is poor. Each time whenever data retrieve from

Database the JAR file or Log File attached to that data so it will be more hectic processed when log file is large this problem is overcome by our System due to which application is used as filter So there will be no problem because most of the restrictions are provided by the application.

4. PROPOSED WORK

To propose a different method, namely Cloud Information Accountability (CIA) framework. It will be based on the concept of information accountability. The basic idea is that simply Data Owner can upload the data into the cloud server after that data will be encrypted. So, User can subscribe the data into the cloud server with certain access rules and policies such as read, write and copy of the original information. With the help of Loggers and Log Harmonizer will have to track the access logs and reports of the data owner. This Process guarantees the security.

4.1 Cloud Information Accountability (CIA) Framework

Cloud Information Accountability (CIA) framework is nothing but the maintaining lightweight and powerful accountability concept that combines aspect if access control, usage control and authentication. This means that, data owners can track not only whether or not the service-level agreements are being privileged, but also enforce access and usage control rules as needed.

4.1.1 Major Components of CIA

There are two main components of the CIA, the first is the logger, and the second is the log harmonizer. The logger is nothing but the component which is strongly coupled with the user's data. So, It is downloaded when the data are retrieved, and the data are copied in the any system. It checks for a particular instance or copy of the user's data or is responsible for logging access to that instance or copy. The log harmonizer creates the central component which permits the user access to the generated log files.

4.1.2 Data Flow

The overall CIA framework is combining data, users, logger and harmonizer in Fig. 1. At the starting, each user creates a pair of private and public keys based on Identity-Based Encryption (step 1 in Fig. 1). Using the produced key, user will create a logger component which is a Java Archives (JAR) file, to store its data items. The JAR file contains a combination of simple access control rules/policies specifying whether and how the cloud servers and perhaps other data stakeholders (users, companies) are

authorized to access the content itself and then, he sends the JAR file to the cloud service provider that he will subscribe it.

To check the authentication of the Cloud Service Provider to the JAR (steps 3-5 in Fig. 1), we use OpenSSL- based certificates. Verifying the user's identity based on his username. Once the authentication succeeds, the user (Service Provider) will be permitted to access the data enclosed in the JAR. To check the configuration setting defined at the time of creation, depending on the JAR file, it will generate usage control related with logging mechanism, or it will display only logging functionality. As per the logging, each user having access to their data at any time. The JAR file will automatically create a log record and encrypt using public key distributed by the data owner.

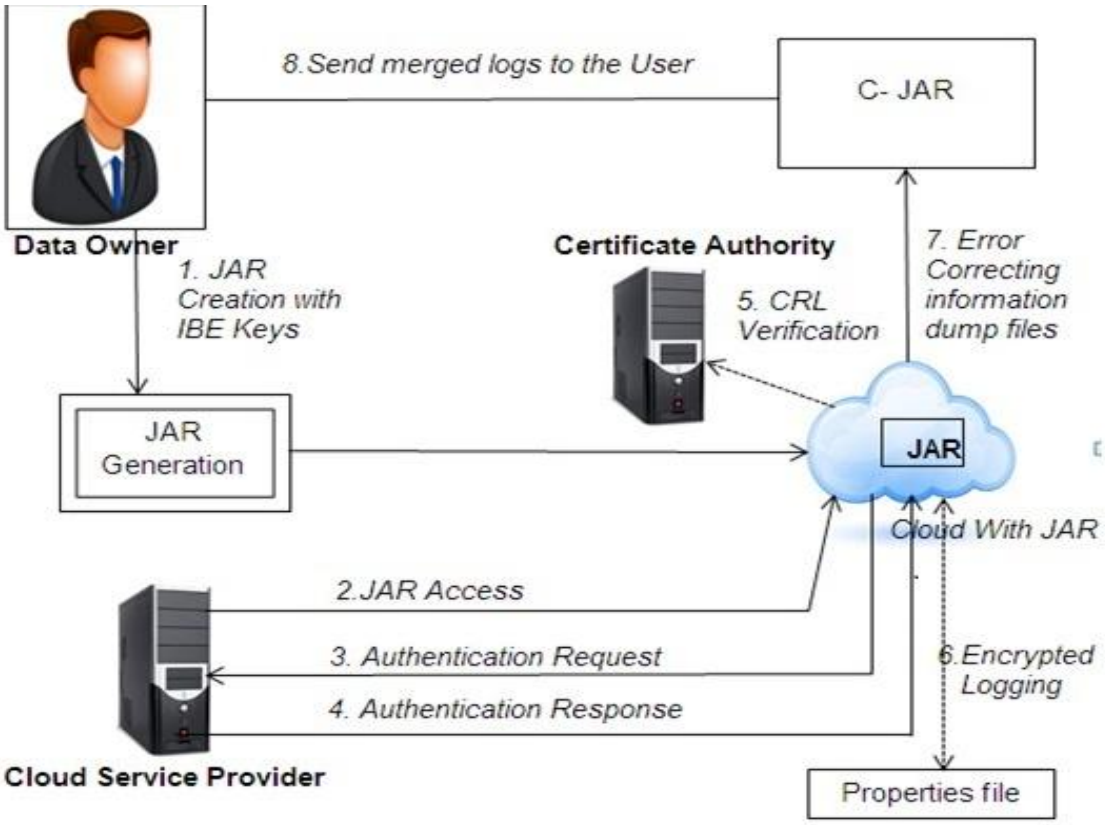

Fig. 1 Proposed Cloud Information Accountability(CIA) Framework

Any one user can not change the encrypted file without having authority or permission. The data owner could not reuse the same pair of key for all JAR files and create different key pairs for separate JARs files. The separate keys can increase the security without introducing any overhead except in the starting phase. A few error correction information will be sent to the log harmonizer to handle possible log file corruption (step 7 in Fig. 1).

4.2 Distributed Auditing Mechanism

Distributed auditing mechanism, including the algorithms for data owners to query the logs regarding their data.

4.2.1 Push mode

In this mode, the logs are periodically pushed to the audit (or data owner) by the harmonizer. The push action can be triggered by using two events:

a. Time elapse for a certain period, according to the time inserted as a part of the JAR file.
b. JAR file exceeds the specified size by the data owner at the time of creation.

4.2.2 Pull mode

In this mode, the auditors allow to retrieve the logs anytime when he wants to check the updated access to their own data. The request will be sent to the harmonizer. Then, the user will be informed of the data locations and obtain an integrated copy of the sealed and authentic and log file.

4.2.3 Push and Pull mode Algorithm

Require:

Size: log file maximum size is specified by the data owner,

time: maximum time allowed to pass before the log file is dumped,

tbeg: timestamp at which the last dump occurred, *log*: the current log file,

pull: indicates whether a command from the data owner is received.

Algorithm:

1. Let TS (NTP) // the network time protocol timestamp
2. pull : = 0
3. rec : = <UID,OID,AccessType,Result,Time,loc>
4. curtime : =TS(NTP)
5. lsize : = sizeof(log)
6. **if**((cutime-tbeg)<time)&&(lsize<size)&&(pull==0) **then**
7. log:=log+ENCRYPT(rec)
8. PING to CJAR
9. **if** PING - CJAR **then**
10. PUSH RS(rec)
11. **else**
12. EXIT(1)
13. **end if**

14 **end if**
15 **if** ((cutime – tbeg) > time) || (lsize>= size)||(pull≠0) **then**
16 **if** PING - CJAR **then**
17 PUSH log
20. RS(log) := NULL
21. tbeg := TS(NTP)
22. pull := 0
23. **else**
24. EXIT (1) **end if**
25. **end if**

Fig. 2 Push and pull PureLog mode

The log retrieval algorithm is shown in Fig.2 for the Push and Pull modes. This algorithm presents synchronization and logging steps with the log harmonizer in case of PureLog mode. Firstly, the algorithm checks whether the size of the JAR file is exceeding a predefined size or the normal time between two alternate dumps has elapsed. The size and time threshold for a dump are specified by the data owner at the time of creation of the JAR. This algorithm also checks whether the data owner has requested for a dump log file. If no one of these events have occurred, it proceeds to encrypt the record and write the error-correction information to the harmonizer.

5. SECURITY ATTACKS

Now, we analyze possible attacks to our framework. Analysis is based on assuming that a user does not lose his master keys to unauthorized parties, while the attacker may try to learn extra information from the log files. Suppose that the attackers may have enough Java programming skills to disassemble a JAR file and prior knowledge of CIA architecture. Firstly, we have to check the JVM file is not corrupted.

5.1 Copying Attack

The most powerful attack is that the attacker/ hacker copies hole/entire JAR files. The attacker may sure that, how accessing that data in the JAR file without access rights given by the data owner. This type attack will be very harmful to secure file. However, this attack will be checked by the auditing mechanism. Every JAR file is required to send log record to the log harmonizer.

5.2 Disassembling Attack

The second possible attack is to disassemble the JAR file of the logger and then attempt to extract useful information out of it or spoil the log records in it.

6. CONCLUSION

In this paper, We proposed original approaches for automatically logging access to the data in the cloud composed with an auditing mechanism. Our approach allows the data owner do not only edit his content, but also apply strong back-end protection if recommended. The main feature is that it enables the data owner to edit his content with those copies of that data were made without his knowledge.

ACKNOWLEGEMENT

The authors express gratitude to Principal, Head of Department (CSE) **Dr. Radhakrishna Naik**. Marathwada Institute of Technology, College of Engineering, Aurangabad, and Maharashtra India. They also express their sincere thanks all the faculty members of the CSE Department, MIT College of Engineering, Aurangabad, and Maharashtra, India for their constant support and enthusiasm.

REFERENCES

[1] SmithaSundareswaran, Anna C. Squicciarini and Dan Lin, "Ensuring Distributed Accountability for Data sharing in the Cloud,",IEEE Transaction on dependable a secure computing, VOL. 9, NO. 4, pp 556-568,August 2012.s

[2] X. Feng, Z. Ni, Z. Shao, and Y. Guo, "An Open Framework for Foundational Proof-Carrying Code," Proc. ACM SIGPLAN Int'l Workshop Types in Languages Design and Implementation,pp. 67-78, 2007

[3] S. Pearson, "Towards Accountability in the Cloud ," Proc. IEEE Internet Computing, pp. 64-69, 2011

[4] S..Pearson,Y. Shen, and M. Mowbray," A privacy Manager for Cloud Computing," Proc. Int'l Conf. Cloud Computing (cloudcom), pp.90-106, 2009.

[5] 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, pp. 187-201, 2005.

[6] D. Boneh and M.K. Franklin, "Identity-Based Encryption from the Weil Pairing," Proc. Int"l Cryptology Conf.

[7] 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. 269-383, 2009.

[8] S. Sundareswaran, A. Squicciarini, D. Lin, and S. Huang, —Promoting Distributed Accountability in the Cloud,‖ Proc. IEEE Int'l Conf. Cloud Computing, 2011.

[9] 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.

[10] B. Crispo and G. Ruffo, "Reasoning about Accountability within Delegation," Proc. Third Int'l Conf. Information and Comm. Security (ICICS), pp.251-260, 2001

[11] Ryan K L Ko, Peter Jagadpramana, Miranda Mowbray, Siani Pearson, Markus Kirchberg, Qianhui, TrustCloud: A Framework for Accountability and Trust in Cloud Computing‖ HP Laboratories, pp 1 – 7, HPL-2011-38

[12] J.W. Holford, W.J. Caelli, and A.W. Rhodes, "Using Self-Defending Objects to Develop Security Aware Applications in Java," Proc. 27th Australasian Conf. Computer Science,vol. 26, pp. 341-349, 2004.