ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 3, Special Issue 5, July 2014

International Conference On Innovations & Advances In Science, Engineering And Technology [IC - IASET 2014]

Organized by

TocH Institute of Science & Technology, Arakunnam, Kerala, India during 16th -18th July -2014

Distributed Information Accountability Using Jar File and Log File Concepts

Mahima K G[1], Sivadasan E T[2]

Mtech , Dept of Computer Science, Vidya Academy of Science and Technology, Thalakkottukara, Thrissur, Kerala, India[1]

Asst Professor, Dept of Computer Science, Vidya Academy of Science and Technology, Thalakkottukara, Thrissur, Kerala, India[2]

ABSTRACT: With the increased reliance on distributed computing platforms to perform computing, the accountability of the data stored on these platforms is becoming a grave issue. In all the current cloud computing platforms, the SLA's(Service Level Agreement) with service providers is the only means of ensuring accountability. But the fact remains that the data owner don't have any direct control over their precious data stored in a cloud service or on a distributed platform. They don't have a mechanism to track where their data is, or whether its usage is according to the agreed SLA. Hence SLA's alone are completely irrelevant for ensuring accountability. Since the conveniences brought by the cloud is impossible to ignore, there is need to address this accountability issue. So this DIA framework is the result of a study to introduce a mechanism to track the files in a cloud for their usage. It provide a JAR based architecture, which stores the files inside special JAR files in a cloud environment. There's inbuilt automatic logging mechanism as well as transparent usage tracking. This makes sure that the owner of the data can be aware of the whereabouts of his data in the cloud. It make use of a specific cloud platform to demonstrate this idea, but it can be as effectively used on any cloud environment or any distributed computing platform for ensuring accountability of one's data.

KEYWORDS: Cloud computing, accountability, Privacy, auditing, data sharing, security.

I. INTRODUCTION

The increasing demand for flexibility in obtaining and releasing computing resources in a cost-effective manner has resulted in a wide adoption of the Cloud computing paradigm. The availability of an extensible pool of resources for the user provides an effective alternative to deploy applications with high scalability and processing requirements. In general, a Cloud computing infrastructure is built by interconnecting large-scale virtualized data centres, and computing resources are delivered to the user over the Internet in the form of an on-demand service by using virtual machines. While the benefits are immense, this computing paradigm has significantly changed the dimension of risks on user's applications. Numerous surveys report that Cloud Computing will be a top 10 technology that enterprise business managers need to be aware of. Cloud Computing is a lower cost delivery model for IT services. Cloud computing promises to increase the velocity with which applications are deployed, increase innovation, and lower costs, all while increasing business agility. Cloud computing that allows it to support every facet, including the server, storage, network, and virtualization technology that drives cloud computing environments to the software that runs in virtual appliances that can be used to assemble applications in minimal time.

A .Motivation

Cloud computing is a technology which uses internet and remote servers to store data and application. In cloud there is no

need to install particular hardware, software on user machine, so user can get the required infrastructure on his machine in cheap charges/rates. Cloud computing is an infrastructure which provides useful on-demand network services to use various resources with less effort. Features of Cloud computing are, huge access of data, application, resources and hardware without installation of any software, user can access the data from any machine or anywhere in the world, business can get resource in one place, that's means cloud computing provides scalability in on demand
services to the business users. Everyone kept their data in cloud, as everyone kept their data in cloud so it becomes public so security issue increases towards private data. Data usage in cloud is very large by users and businesses, so data security in cloud is very important issue to solve. Many users want to do business of his data through cloud, but users may not know the machines which actually process and host their data. While enjoying the convenience brought by this new technology, users also start worrying about losing control of their own data. The data processed on clouds are often outsourced, leading to a number of issues related to accountability, including the handling of personally identifiable information. Such fears are becoming a significant barrier to the wide adoption of cloud services.

To solve the security issues in cloud; other user can't read the respective users data without having access. Data owner should not bother about his data, and should not get fear about damage of his data by hacker; there is need of security mechanism which will track usage of data in the cloud. Accountability is necessary for monitoring data usage. To allay users concerns, it is essential to provide an effective mechanism for users to monitor the usage of their data in the cloud. For example, users need to be able to ensure that their data are handled according to the service level agreements made at the time they sign on for services in the cloud. Conventional access control approaches developed for closed domains such as databases and operating systems, or approaches using a centralized server in distributed environments, are not suitable, due to the following features characterizing cloud environments. First, data handling can be outsourced by the direct cloud service provider (CSP) to other entities in the cloud and theses entities can also delegate the tasks to others, and so on. Second, entities are allowed to join and leave the cloud in a flexible manner. As a result, data handling in the cloud goes through a complex and dynamic hierarchical service chain which does not exist in conventional environments.
Objective of this work is to provide a novel highly decentralized information accountability framework to keep track of the actual usage of the user's data in the cloud. Unlike privacy protection technologies which are built on the hide-it-or-lose-it perspective, this information accountability should focuses on keeping the data usage transparent and track able. And it should provide end-to-end accountability in a highly distributed fashion.

B. Current Scenario

Cloud computing has raised a range of privacy and security issues. The user data or application resides in cloud at least for a certain time in that time period those users don't know who is actually handling his/her data or to whom it is passing to control. Till date very few works has been done on this particular area. Pearson et al. have proposed accountability mechanisms to address privacy concerns of end users and then develop a privacy manager. Their basic idea is that the users private data are sent to the Cloud storage in an encrypted form, and the processing is done on the encrypted data. The output of the processing is decrypted by the privacy manager to reveal the correct result. The main issue with the privacy manager is it only gives minimum security to the user's data. Once it is decrypted it does not guarantee the safety of the data. A significant work is done by Smitha Sundareswaran et al. have illustrated the method of automatic and enforceable logging mechanism in the cloud. Using object oriented approach (SDO). They also have illustrated the mechanism of pull mode and push mode. In that paper they have used object oriented technology to ensure transparency in users data (using JAR).Another work is by Mont et al. who proposed an approach for strongly coupling content with access control, using Identity-Based Encryption(IBE).

Presently many researcher's are continuing research in this area within the A4cloud project. The cloud accountability project(A4cloud for short) focuses on the Accountability for Cloud and Other Future Internet Services as the most critical prerequisite for effective governance and control of corporate and private data processed by cloud-based IT services.

C. Problem Statement

The problem can be stated using an illustrative example which serves as the basis of the problem statement and will be used throughout the project to demonstrate the main features of this system. A professional photographer, plans to sell her photographs by using some Cloud Services. For her business in the cloud, she has the following requirements:

1. Her photographs are downloaded only by users who have paid for her services.
2. Potential buyers are allowed to view her pictures first before they make the payment to obtain the download right.
3. In case any dispute arises with a client, she wants to have all the access information of that client.

With the above scenario in mind, one can identify the common requirements and develop several guidelines to achieve data accountability in the cloud. A user who subscribed to a certain cloud service, usually needs to send his/her data as well as associated access control policies (if any) to the service provider. After the data are received by the cloud service provider, the service provider will have granted access rights, such as read, write, and copy, on the data. Using conventional access control mechanisms, once the access rights are granted, the data will be fully available at the service provider. In order to track the actual usage of the data, this framework aim to develop novel logging and auditing techniques which satisfy the following requirements:

1. 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.
2. 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.
3. 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.

II. OVERVIEW OF PROPOSED METHOD

Cloud computing is a large infrastructure which provide many services to user without installation of resources on their own machine. This is the pay as you use model. Examples of the cloud services are Yahoo email, Google, Gmail and Hotmail. There are many users, businesses, government uses cloud, so data usage in cloud is large. So data maintenance in cloud is complex. Many Artists wants to do business of their art using cloud. For example one of the artist want to sell his painting using cloud then he want that his paintings must be safe on cloud no one can misuse his paintings.

There is need to provide technique which will audit data in cloud. On the basis of accountability, we proposed one mechanism which keeps use of data transparent means data owner should get information about usage of his data. This mechanism support accountability in distributed environment Data owner should not bother about his data, he may know his data is handled according to service level agreement and his data is safe on cloud. Data owner will decide the access rules and policies and user will handle data using this rule and logs of each data access have been created. In this mechanism there are two main components i.e. logger and log harmonizer. The overall CIA framework, combining data, users, logger and harmonizer is sketched in Figure. 1.

The logger is with the data owner's data, it provides logging access to data and encrypts log record by using public key which is given by data owner and send it to log harmonizer. The log harmonizer is performing the monitoring and rectifying, it generates the master key it holds decryption key decrypting the logs, and at the client side decryption it sends key to client. In this mechanism data owner will create private key and public key, using generated key owner will create logger which is a JAR file (JAVA Archives), it includes his policies like access policies and logging policies with data send to cloud service provider.

Authentication of cloud service provider has been done using open SSL based certificates after authentication of cloud service provider user can be able to access data in JAR, log of each data usage has been created and encrypted using public key and it automatically send to log harmonizer for integrity log records are signed by entity which is using the data and log records are decrypted and access by owner. In push mode logs are automatically send to data owner and in pull mode owner can demand logs, so he can see access of his data at anytime, anywhere and he can do monitoring of his data. In Figure 1working of accountability mechanism in cloud is given in this when user will access data then log of each access is created by logger and periodically sent to log harmonizer, log harmonizer send this logs to data owner and data owner can see logs and take appropriate action if he wants.

Figure 1: Accountability Mechanism in cloud

In accountability mechanisms the log records are generated as access of data in jar happened then it create log record log rec (Lr). Lr = r1, r2, r3, r4...rk. Parameters used for log record are

rk = (id, action, T, loc) Where,
rk = log record
id = user identification
action = perform on users data
T = Time at location loc
loc = Location

III. DESIGN DESCRIPTION

This paper presents effective mechanism, which performs automatic authentication of users and create log records of each data access by the user. There are four completely different modules during this system. They are shown below.

1. JAR Generation
2. Log Record Generation
3. Mode Setting
4. Steganography
5. One Time Password

Figure 2: Architecture

A. Logger Generation

The JAR file contains the data and a set of access control rules specifying whether and how the cloud servers and possibly other data interested party (users, companies) are authorized to access the content itself. Depending on the configuration settings defined at the time of creation, the JAR will provide usage control associated with logging, or will provide only logging functionality. We leverage the programmable capability of JARs to conduct automated logging. A logger component is a nested Java JAR file which stores a user's data items and corresponding log files. As shown in Fig 5.3, the proposed JAR file consists of one outer JAR enclosing one or more inner JARs.

The main responsibility of the outer JAR is to handle authentication of entities which want to access the data stored in the JAR file. In our context, the data owners may not know the exact CSPs that are going to handle the data. Hence,authentication is specified according to the servers functionality (which assume to be known through a lookup service), rather than the servers URL or identity. For example, a policy may state that Server X is allowed to download the data if it is a storage server. As discussed below, the outer JAR may also have the access control functionality to enforce the data owners requirements, specified as Java policies, on the usage of the data. A Java policy specifies which permissions are available for a particular piece of code in a Java application environment. The permissions expressed in the Java policy are in terms of File System Permissions. However, the data owner can specify the permissions in user-centric terms as opposed to the usual code-centric security offered by Java, using Java Authentication and Authorization Services. Moreover, the outer JAR is also in charge of selecting the correct inner JAR according to the identity of the entity who requests the data. Each inner JAR contains the encrypted data, class files to facilitate retrieval of log files and display enclosed data in a suitable format, and a log file for each encrypted item.

To carry out these functions, the inner JAR contains a class file for writing the log records, another class file which corresponds with the log harmonizer, the encrypted data, a third class file for displaying or downloading the data (based on whether we have a PureLog, or an AccessLog), and the public key of the IBE key pair that is necessary for encrypting the log records. No secret keys are ever stored in the system. The outer JAR may contain one or more inner JARs, in addition to a class file for authenticating the servers or the users, another class file finding the correct inner JAR, a third class file which checks the JVMs validity using oblivious hashing. Further, a class file is used for managing the GUI for user authentication and the Java Policy.

B. Log Record Generation

Log records are generated by the logger component. Logging occurs at any access to the data in the JAR, and new log entries are appended sequentially, in order of creation LR hr1; . . . ; rki. Each record ri is encrypted individually and appended to the log file. In particular, a log record takes the following form: rk = (id, action, T, loc) Where,
rk = log record
id = user identification
action = perform on user’s data
T = Time at location loc
loc = Location

Anexample of log record for a single file is shown below. Suppose that a cloud service provider with ID Kronos, located in USA, read the image in a JAR file (but did not download it) at 4:52 pm on May 20, 2011. The corresponding log record is hKronos, View, 2011-05-29 16:52:30,USA, The location is converted from the IP address for improved readability. To ensure the correctness of the log records, it verify the access time, locations as well as actions. In particular, the time of access is determined using the Network Time Protocol (NTP) to avoid suppression of the correct time by a malicious entity. The location of the cloud service provider can be determined using IP address. The JAR can perform an IP lookup and use the range of the IP address to find the most probable location of the CSP. The most critical part is to log the actions on the user’s data. In the current system, it support four types of actions, i.e., Act has one of the following four values: view, download, timed-access, and Location-based-access. For each action, it propose a specific method to correctly record or enforce it depending on the type of the logging module.

C. Mode Setting

To allow users to be timely and accurately informed about their data usage, this distributed logging mechanism is complemented by an innovative auditing mechanism. It support two complementary auditing modes: push mode and pull

mode.
1. Push mode:In this mode, the logs are periodically pushed to the data owner (or auditor) by the harmonizer. The push action will be triggered by either type of the following two events: one is that the time elapses for a certain period according to the temporal timer inserted as part of the JAR file; the other is that the JAR file exceeds the size stipulated by the content owner at the time of creation. After the logs are sent to the data owner, the log files will be dumped, so as to free the space for future access logs. Along with the log files, the error correcting information for those logs is also dumped. This push mode is the basic mode which can be adopted by both the PureLog and the AccessLog, regardless of whether there is a request from the data owner for the log files. This mode serves two essential functions in the logging architecture: it ensures that the size of the log files does not explode and it enables timely detection and correction of any loss or damage to the log files. Concerning the latter function, notice that the auditor, upon receiving the log file, will verify its cryptographic guarantees, by checking the records integrity and authenticity. By construction of the records,

the auditor, will be able to quickly detect forgery of entries, using the checksum added to each and every record.
2. Pull mode: This mode allows auditors to retrieve the logs anytime when they want to check the recent access to their own data. The pull message consists simply of an FTP pull command, which can be issues from the command line. For naive users, a wizard comprising a batch file can be easily built. The request will be sent to the harmonizer, and the user will be informed of the data's locations and obtain an integrated copy of the authentic and sealed log file.

D. Steganography

Steganography comes from the Greek words Steganos (Covered) and Graptos (Writing). The term Steganography came into use in 1500s after the appearance of Trithemius book on the subject Steganographia. The word Steganography technically means covered or hidden writing. Its ancient origins can be traced back to 440 BC. In ancient times, messages were hidden on the back of wax writing tables, written on the stomachs of rabbits, or tattooed on the scalp of slaves. Invisible ink has been in use for centuries for fun by children and students and for serious espio-nage by spies and terrorists The majority of today's steganographic systems uses multimedia objects like image, audio, video etc as cover media because people often transmit digital pictures over email and other Internet communication In modern approach, depending on the nature of cover object, steganography can be divided into five types: Text Steganography, Image Steganography, Audio Steganography, Video Steganography and Protocol Steganography. So, in the modern age so many steganographic techniques have been designed which works with the above concerned objects. With respect to
Steganography there is a problem of unauthorized data access.
 Image steganography: To hide information, straight message insertion may encode every bit of information in the image or selectively embed the message in noisy areas that draw less attentionthose areas where there is a great deal of natural color variation. The message may also be scattered randomly throughout the image. A number of ways exist to hide information in digital media.

In this framework Steganography is used for ensuring more data security. Here whenever a data is downloaded by any user, that user details will be hidden inside the data using Steganography. Its main use is to trace the user who using the data in unauthorized places. For Example if one user downloaded an image from a photo selling website and suppose that he try to sell that image in another website, then the data owner can trace that user details from that image. For this the data owner has to download that image and then extract the user details using watermark Extractor after that the owner can take actions against that user.

E. One Time Password Schemes

They are used by almost all business applications for authentication. However static passwords have lots of limitations e.g. passwords can get hacked; careless employee may write down passwords somewhere; system with saved passwords

may be used by various users or a malicious user may reset all passwords just to create havoc. Hence it is advisable to move to a more dynamic password scheme like one time passwords or OTP.OTP generation can be done by various OTP generation algorithms for generating strings of passwords. In this DIA framework it provide a one time pass word for the uploading purpose where it can have give some more accountability. The purpose of a one-time password (OTP) is to make it more difficult to gain unauthorized access to restricted resources, like a computer account. Traditionally static passwords can more easily be accessed by an unauthorized intruder given enough attempts and time. By constantly altering the password, as is done with a one-time password, this risk can be greatly reduced. The proposed system attempts to alleviate the problem of shoulder surfing or eves dropping by making the replay of a password
useless. Every time a user is authenticated by totally different password[15].At the time of uploading or deleting data the owner will get a one time password in his mail and then he has to enter it to do further actions.

IV. CONCLUSION

This DIA framework presents an effective mechanism, which performs automatic authentication of users and create log records of each data access by the user. Data owner can audit his content on cloud, and he can get the confirmation that his data is safe on the cloud. Data owner also able to know the duplication of data made without his knowledge. Data owner should not worry about his data on cloud using this mechanism and data usage is transparent, using this mechanism. One of the main features of this work is that it enables the data owner to audit even those copies of its data that were made without his knowledge. As an extension to the base paper, here steganography and One Time Password concepts are also added to improve security and to provide a higher level of accountability.

REFERENCES

[1] T. Mather, S. Kumaraswamy, and S. Latif,'Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance (Theory in Practice)",first ed. O Reilly,2009.

[2] 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.

[3] S. Pearson and A. Charlesworth,"Accountability as a Way Forward for Privacy Protection in the Cloud,",in ProcProc First Int'l conf. Cloud Computing,,2009.

[4] B. Chun and A. C. Bavier, "Decentralized Trust Management and Accountability in Federated System", ,in ProcProc. Ann. Hawaii Int'l Conf. System Science (HICSS),,2004.

[5] R. Corin, S. Etalle, J.I. den Hartog, G. Lenzini, and I. Staicu; "A Logic for Auditing Accountability in Decentralized Systems",in ProcProc. IFIP TC1 WG1.7 Workshop Formal Aspects in Security and Trust,,pp. 187-201, 2005.

[6] R. Jagadeesan, A. Jeffrey, C. Pitcher, and J. Riely; 'Towards a Theory of Accountability and Audit, in Proc.14th European Conf. Research in Computer Security (ESORICS),pp. 152-167, 2009.

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

[8] W. Lee, A. Cinzia Squicciarini, and E. Bertino,'The Design and Evaluation of Accountable Grid Computing System",Proc. 29th IEEE Intl Conf. Distributed Computing Systems (ICDCS 09),pp. 145-154, 2009.

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

[10] A. Squicciarini , S. Sundareswaran and D. Lin"Preventing Information Leakage from Indexing in the Cloud",in ProcProc. IEEE Int'l Conf. Cloud Computing,, 2010.

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

[12] M.C. Mont, S. Pearson, and P. Bramhall,'Towards Accountable Management of Identity and Privacy: Sticky Policies and Enforceable Tracing Services",Proc. Intl Workshop Database and Expert Systems Applications (DEXA),pp. 377-382, 2003.

[13] R. Bose and J. Frew, 'Lineage Retrieval for Scientific Data Processing: A Survey",ACM Computing Surveys,vol. 37, pp. 1- 28, Mar. 2005.

[14] P. Buneman, A. Chapman, and J. Cheney, 'Provenance Management in Curated Databases",Proc. ACM SIGMOD Intl Conf. Management of Data (SIGMOD 06),pp. 539-550, 2006.

[15] Richa Chowdhary,Satyakshma Rawat,'One Time Password for Multi-Cloud Environment",International Journal of Advanced Research in Computer

Science and Software Engineering.Volume 3, Issue 3, March 2013.
[16] WAWGE P.U. AND RATHOD A.R., ‘Cloud Computing Security With Steganography and Cryptography AES Algorithm Technology.”,World Research Journal of Computer Architecture,Volume 1, Issue 1, 2012, pp-11-15.
[17] Smitha Sundareswaran, 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, pg 556-568, 2012.
[18] S. Pearson , Y. Shen, and M. Mowbray,, A privacy Manager for Cloud Computing,”, in Proc3rd Int. Conf. Cloud Comput,,pp.90-106,2009.
[19] S. Sundareswaran, A. Squicciarini, D. Lin, and S. Huang, ‘Promoting Distributed Accountability in the Cloud”,Proc. IEEE Intl Conf. Cloud Computing, 2011.
[20] 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.
[21] B. Schneier, ‘Applied Cryptography: Protocols, Algorithms, and Source Code.”,,in C. John Wiley and Sons,1993
[22] Praveen Gauravaram, John Kelesy, Lars Knudsen, and Soren Thomsen, ‘On Hash function using Checksums”,
[23] 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.
[24] 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.
[25] A. Pretschner, F. Schuo tz, C. Schaefer, and T. Walter,‘Usage Control Enforcement: Present and Future”,IEEE Security & Privacy,vol. 6, no. 4, pp. 44-53, July/ Aug. 2008.
[26] A. Pretschner, F. Schuo tz, C. Schaefer, and T. Walter,‘Distributed Usage Control”,Comm. ACM,vol. 49, no. 9, pp. 39-44, Sept. 2006.
[27] J.H. Lin, R.L. Geiger, R.R. Smith, A.W. Chan, and S. Wanchoo,‘Method for Authenticating a Java Archive (jar) for Portable Devices”,US Patent,6,766,353, July 2004.