Research Paper

Volume : 2 | Issue : 10 | October 2013 • ISSN No 2277 - 8179

A Framework for Cloud Information Accountability

Engineering

KEYWORDS : Cloud, Distributed, Data, Accountability, Sharing

Raj Kumar Patil — M.Tech Student, Mallareddy Institute of Engineering and Technology, Hyderabad

Rehaman Pasha — Assistant Professor, Mallareddy Institute of Engineering and Technology, Hyderabad

ABSTRACT

Distributed computing empowers quite versatile administrations to be effortlessly expended over the Internet on an as-required premise. A major characteristic of the cloud administrations is that clients' information are as a rule handled remotely in obscure machines that clients don't own or work. While reveling in the accommodation carried by this new developing innovation, clients' reasons for alarm of losing control of their own information (especially, fiscal and health information) can turn into a noteworthy restraint to the wide selection of cloud administrations. To address this issue, in this paper, we propose a novel exceptionally decentralized data responsibility structure to stay informed concerning the genuine utilization of the clients' information in the cloud. Specifically, we propose an item focused methodology that empowers encasing our logging instrument together with clients' information and approaches. We power the JAR programmable competencies to both make a dynamic and voyaging question, and to guarantee that any right to gain entrance to clients' information will trigger verification and robotized logging neighbourhood to the Jars. To fortify client's control, we additionally give appropriated reviewing instruments. We furnish far reaching exploratory studies that exhibit the effectiveness and viability of the proposed approaches.

INTRODUCTION

Distributed computing shows another approach to supplement the present utilization and conveyance display for IT administrations dependent upon the Internet, by accommodating alertly versatile and frequently virtualized assets as an administration over the Internet. To date, there is various outstanding businesses and singular distributed computing administrations, incorporating Amazon, Google, Microsoft, Yahoo, and Salesforce. Parts of the administrations furnished are absorbed from the clients who no more drawn out need to be specialists of innovation framework. Additionally, clients may not know the machines which really handle and host their information. While reveling in the comfort carried by this new innovation, clients additionally begin stressing over losing control of their own information. The information handled on mists is regularly outsourced, accelerating various issues identified with responsibility, incorporating the taking care of directly identifiable data. Such feelings of trepidation are turning into a critical boundary to the wide selection of cloud administrations.

The configuration of the CIA schema presents significant tests, incorporating extraordinarily distinguishing Csps, guaranteeing the dependability of the log, adjusting to a profoundly decentralized foundation, and so on. Our fundamental approach to tending to these issues is to influence and augment the programmable proficience of JAR (Java Archives) records to immediately log the use of the clients' information by any substance in the cloud. Clients will send their information as well as any strategies, for example access control approaches and logging approaches that they need to implement, encased in JAR documents, to cloud administration suppliers. Any right to gain entrance to the information will trigger a computerized and verified logging system nearby to the Jars. We propose a novel immediate and enforceable logging component in the cloud. As far as anyone is concerned, this is the first run through a precise approach to information responsibility through the novel use of JAR indexes is proposed. Our proposed construction modelling is stage autonomous and greatly decentralized, in that it doesn't require any devoted validation or space framework set up.

We go past universal access control in that we give a certain level of utilization control for the secured information after these are conveyed to the beneficiary. We direct investigate a genuine cloud testbed. The outcomes exhibit the productivity, versatility, and granularity of our methodology. We likewise give a point by point security dissection and talk over the unwavering quality and quality of our structural planning.

II. EXISTING SYSTEM

To relieve clients' concerns, it is key to furnish a viable component for clients to screen the utilization of their information in the cloud. Case in point, clients need to have the capacity to guarantee that their information is taken care of as per the administration level assentions set aside a few minutes they sign on for administrations in the cloud. Tried and true access control methodologies created for shut areas for example databases and working frameworks, or methodologies utilizing a concentrated server within conveyed situations, are not suitable, because of the accompanying characteristics describing cloud situations. In the first place, information taking care of could be outsourced by the immediate cloud administration supplier (CSP) to different substances in the cloud and propositions elements can additionally appoint the undertakings to others, et cetera. Second, elements are permitted to join and leave the cloud in an adaptable way. Thus, information taking care of in the cloud experiences a mind boggling and rapid progressive administration chain which does not exist in accepted situations. Scientists have examined responsibility generally as a provable property through cryptographic systems, especially in the connection of electronic trade. A delegate work here. The creators propose the use of approaches appended to the information and present a rationale for responsibility information in conveyed settings. As of late proposed a rationale for outlining responsibility based circulated frameworks. Appointment is reciprocal to our work, in that we don't point at regulating the data workflow in the mists. In a synopsis, all these works stay at a hypothetical level and don't incorporate any calculation for undertakings like required logging.

III. PROPOSED FRAMEWORK

Cloud Information Accountability

The Cloud Information Accountability skeleton proposed in this work behaviors robotized logging and disseminated reviewing of important access performed by any entity, carried out at any purpose of time at any cloud administration provider. It has two major parts: lumberjack and log harmonizer. The lumberjack is unequivocally coupled with client's information (either single or various information things). Its principle undertakings incorporate immediately logging access to information things that it holds, encoding the log record utilizing people in general key of the substance holder, and intermittently sending them to the log harmonizer. It might additionally be arranged to guarantee that right to gain entrance and use control strategies connected with the information are respected. At last, the lumberjack is likewise answerable for creating the failure rectification data

for every log record and sends the same to the log harmonizer.

The log harmonizer is answerable for auditing. it underpins two inspecting methodologies: push and pull. Under the push procedure, the log index is pushed again to the information holder intermittently in a mechanized manner. The force mode is an on-interest methodology, whereby the log record is gotten by the information possessor as regularly as asked. The JAR index incorporates a set of basic access control tenets determining if and how the cloud servers, and potentially other information stakeholders (clients, organizations) are commissioned to enter the substance itself.

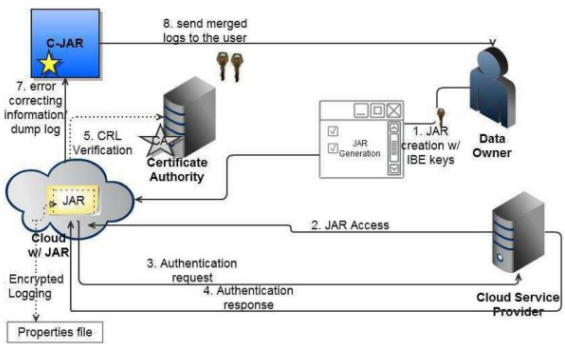


Fig .1: Framework of Overview of the cloud information accountability

At that point, he sends the JAR index to the cloud administration supplier that he subscribes to validate the CSP to the JAR, we utilize Open ssl based authentications, wherein a trusted declaration power ensures the CSP.

Automated Logging Mechanism
a) The Logger Structure

We power the programmable capability of Jars to control electronic logging. A logger portion is a settled Java JAR record which safeguards a customer's data things and looking at log files. as showed in Fig. 2, our proposed JAR record embodies one outside JAR walling one in or more internal Jars. Each interior JAR holds the encoded data, class records to assist recuperation of log lists and showcase encased data in a suitable plan, and a log record for every one mixed thing. We maintain two choices:

Pure log: Its primary undertaking is to record each right to gain entrance to the information. The log indexes are utilized for unadulterated evaluating reason.

Access log: It has two capacities: logging activities and implementing access control. On the off chance that a right to gain entrance solicit is denied, the Jar will record the time when the solicitation is made. Provided that the right to gain entrance solicit is truly, the JAR will also record the right to gain entrance data along with the term for which the right to gain entrance is permitted.

In the current system, we support four types of actions, i.e., Act has one of the following four values: view, download, timed_access, and Location-based access.

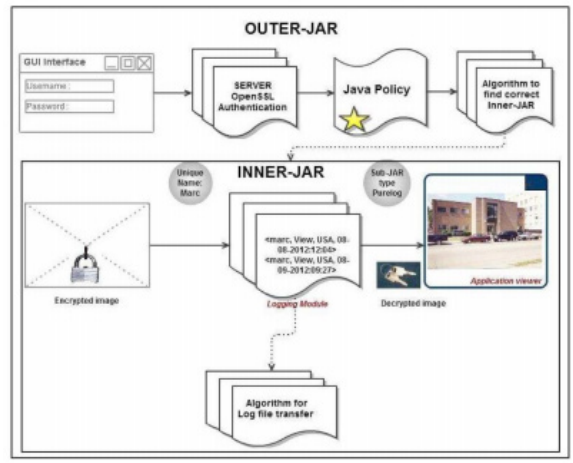


Fig .2: JAR File structure

b) Dependability of Logs
In this section, we discuss how we ensure the dependability of logs. In particular, we aim to prevent the following two types of attacks. First, an attacker may try to evade the auditing mechanism by storing the JARs remotely, corrupting the JAR, or trying to prevent them from communicating with the user. Second, the attacker may try to compromise the JRE used to run the JAR files.

c) Push and Pull Mode
To permit clients to be convenient and correctly educated about their information utilization, our dispersed logging instrument is supplemented by an inventive reviewing component. We back two reciprocal examining modes:

1) Push mode.
2) Pull mode.

Push mode: In this mode, the logs are occasionally pushed to the information possessor (or reviewer) by the harmonizer. The push movement will be triggered by either sort of the accompanying two occasions: one is that the time passes for a certain period as per the worldly timer embedded as a component of the JAR document; the different is that the JAR record surpasses the size stipulated by the substance holder around then of creation. After the logs are sent to the information manager, the log documents will be dumped, in order to free the space for future access logs. Plus the log indexes, the lapse adjusting data for those logs are additionally dumped.

Pull mode: This mode permits examiners to recover the logs whenever when they need to check the later access to their own particular information. The force message comprises essentially of a FTP pull summon, which might be issues from the charge line. For gullible clients, a wizard containing a cluster index could be effectively manufactured. The appeal will be sent to the harmonizer, and the client will be educated of the information's areas and get a joined duplicate of the legitimate and fixed log document.

Require: $size$: maximum size of the log file specified by the data owner, $time$: maximum time allowed to elapse before the log file is dumped, $tbeg$: timestamp at which the last dump occurred, log: current log file, $pull$: indicates whether a command from the data owner is received.

Volume : 2 | Issue : 10 | October 2013 • ISSN No 2277 - 8179

Let TS(NTP) be the network time protocol timestamp
$pull = 0$
$rec := \langle UID, OID, AccessType, Result, Time, Loc \rangle$
$curtime := TS(NTP)$
$lsize := sizeof(log)$ //current size of the log
if $((cutime$ -$tbeg) < time)\&\&$
$(lsize < size)\&\&(pull == 0)$ **then**
$log := log + ENCRYPT(rec)$ // ENCRYPT is the encryption function used to encrypt the record
$PING$ to CJAR //send a PING to the harmonizer to check if it is alive
if $PING$-CJAR **then**
PUSH $RS(rec)$ // write the error correcting bits
else
EXIT(1) // error if no PING is received
end if
end if
if $((cutime - tbeg) > time)||(lsize >= size)$ $||(pull \neq 0)$ **then**
// Check if PING is received

if $PING$-CJAR **then**
PUSH log //write the log file to the harmonizer
$RS(log) := NULL$ // reset the error correction records
$tbeg := TS(NTP)$ // reset the $tbeg$ variable
$pull := 0$
else
EXIT(1) // error if no PING is received
end if
end if

IV. RESULT ANALYSIS

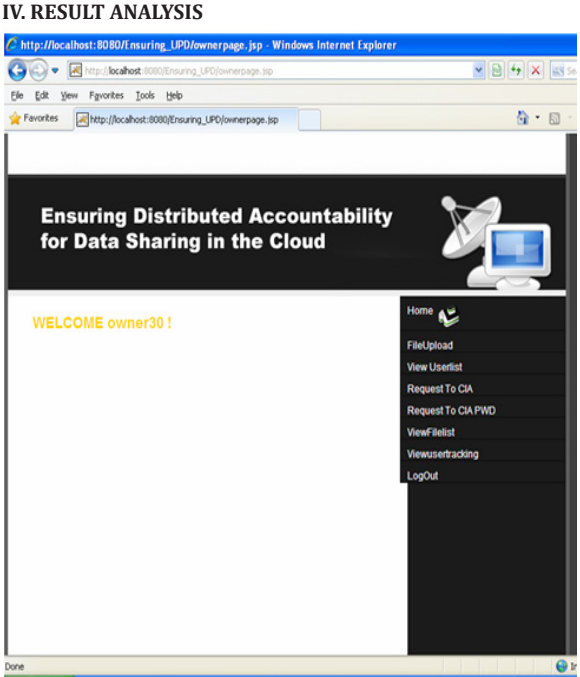


Fig .3: Owner Home page

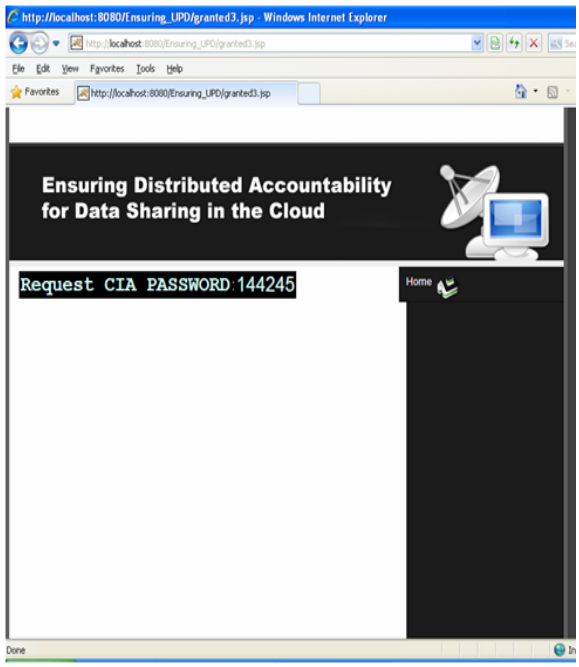
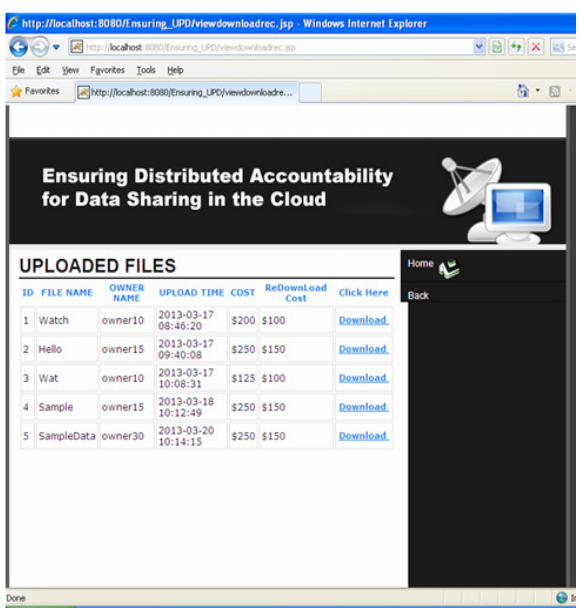


Fig .4: Owner CIA request password page



ID	FILE NAME	OWNER NAME	UPLOAD TIME	COST	ReDownLoad Cost	Click Here
1	Watch	owner10	2013-03-17 08:46:20	$200	$100	Download
2	Hello	owner15	2013-03-17 09:40:08	$250	$150	Download
3	Wat	owner10	2013-03-17 10:08:31	$125	$100	Download
4	Sample	owner15	2013-03-18 10:12:49	$250	$150	Download
5	SampleData	owner30	2013-03-20 10:14:15	$250	$150	Download

Fig .5: File Details page

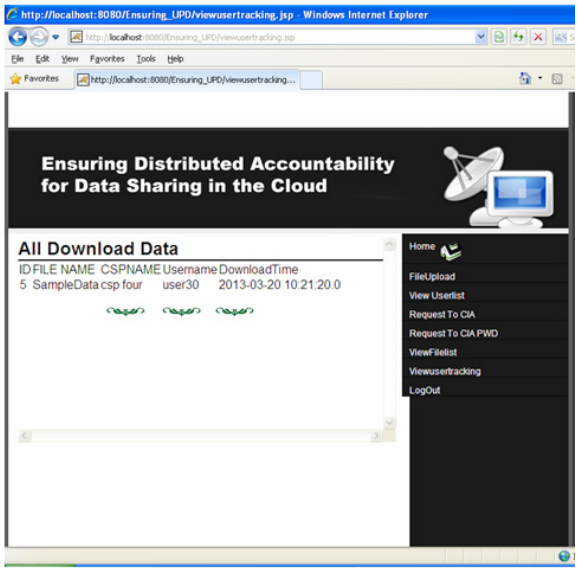


Fig .6: View User Data Tracking

CONCLUSION

We proposed inventive methodologies for immediately logging any right to gain entrance to the information in the cloud together with an examining instrument. Our methodology permits the information holder to review his substance as well as implement solid back-closure assurance if required. Besides, one of the fundamental characteristics of our work is that it empowers the information holder to review even those duplicates of its information that were made without his learning. Sometime later, we want to refine our methodology to confirm the respectability of the JRE and the verification of Jars. For example, we will explore if it is conceivable to power the idea of a protected JVM being produced by IBM. This research is pointed at furnishing programming alter imperviousness to Java requisitions. In the enduring, we want to plan a complete and more nonexclusive article arranged approach to expedite independent assurance of voyaging substance. We might want to back a mixture of security approaches, for instance indexing arrangements for content documents, utilization control for executable, and nonexclusive responsibility and provenance controls.

REFERENCE

[1] P. Buneman, A. Chapman, and J. Cheney, "Provenance Management in Curated Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '06), pp. 539-550, 2006. | [2] B. Chun and A.C. Bavier, "Decentralized Trust Management and Accountability in Federated Systems," Proc. Ann. Hawaii Int'l Conf.System Sciences (HICSS), 2004. | [3] OASIS Security Services Technical Committee, "Security Assertion Markup Language (saml) 2.0," http://www.oasis-open.org/committees/tc home.php?wg abbrev=security, 2012. | [4] R. Corin, S. Etalle, J.I. den Hartog, G. Lenzini, and I. Staicu, "A Logic for Auditing Accountability in Decentralized Systems," Proc. IFIP TC1WG1.7 Workshop Formal Aspects in Security and Trust,pp. 187-201, 2005. | [5] B. Crispo and G. Ruffo, "Reasoning about Accountability within Delegation," Proc. Third Int'l Conf. Information and Comm. Security (ICICS), pp. 251-260, 2001. | [6] J.W. Holford, W.J. Caelli, and A.W. Rhodes, "Using SelfDefending Objects to Develop Security Aware Applications in Java," Proc. 27th Australasian Conf. Computer Science, vol. 26, pp. 341-349, 2004. | [7] Trusted Java Virtual Machine IBM, http://www.almaden.ibm. com/cs/projects/jvm/, 2012. | [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] R. Jagadeesan, A. Jeffrey, C. Pitcher, and J. Riely, "Towards a Theory of Accountability and Audit," Proc. 14th European Conf. Research in Computer Security (ESORICS), pp. 152-167, 2009. | [10] M.C. Mont, S. Pearson, and P. Bramhall, "Towards Accountable Management of Identity and Privacy: Sticky Policies and Enforceable Tracing Services," Proc. Int'l Workshop Database and Expert Systems Applications (DEXA), pp. 377-382, 2003. | [11] S. Oaks, Java Security. O'Really, 2001. | [12] J. Park and R. Sandhu, "Towards Usage Control Models: Beyond Traditional Access Control," SACMAT '02: Proc. Seventh ACM Symp. Access Control Models and Technologies, pp. 57-64, 2002. |