International Journal of Research Studies in Computer Science and Engineering (IJRSCSE)
Volume 2, Issue 7, July 2015, PP 1-6
ISSN 2349-4840 (Print) & ISSN 2349-4859 (Online)
www.arcjournals.org

Secure Log Data Storage in Cloud by Using AES Algorithm

[1]G.Sindhuja, [2] G.Prabhakar, [3]G.Bhanu Prasad

[1]Final M.Tech Student, [2]Associate Professor, [3]Associate Professor
[1,2,3]Dept of Computer Science and Engineering
[1,2,3]Malla Reddy Engineering College For Women, Hyderabad, India

Abstract: *Cloud computing permits very ascendible services to be merely consumed over the online on associate as-needed basis. A serious feature of the cloud services is that users' information area unit typically processed remotely in unknown machines that users don't own or operate. whereas enjoying the convenience brought by this new rising technology, users' fears of losing management of their own information (particularly, monetary and health data) can become a giant barrier to the wide adoption of cloud services. To handle this drawback, during this paper, we have a tendency to tend to propose a very distinctive very suburbanized info answerableness framework to remain track of the actual usage of the users' knowledge among the cloud. Significantly, we have a tendency to tend to propose associate object-centered approach that allows insertion our work mechanism in conjunction with users' info and policies. We tend to leverage the JAR programmable capabilities to every produce a dynamic and traveling object, and to create certain that any access to users' info will trigger authentication and automatic work native to the JARs. To strengthen user's management, we have a tendency to conjointly give distributed auditing mechanisms. We provide full experimental studies that demonstrate the efficiency and effectiveness of the planned approaches. We implement how to store secure log file in cloud and that file we can change read, write, delete, upload and download. We can implement AES algorithm that uses for log monitor and log generator.*

Keywords: *Authentication, JAR, cloud, AES.*

1. INTRODUCTION

Cloud computing presents a brand new thanks to supplement the current consumption and delivery model for IT services supported the net, by providing for dynamically scalable and infrequently virtualized resources as a service over the Internet. To date, there are variety of notable business and individual cloud computing services, together with Amazon, Google, Microsoft, Yahoo, and Sales force. Details of the services provided are abstracted from the users who no longer got to be specialists of Technology infrastructure. Moreover, users might not apprehend the machines that truly process and host their information. Whereas enjoying the convenience brought by this new technology, users additionally begin worrying about losing management of their own information. The info processed on clouds are usually outsourced, resulting in variety of issues associated with answerableness, together with the handling of personally recognizable data. Such fears are getting a significant barrier to the wide adoption of cloud services. To allay users' issues, it's essential to produce associate degree effective mechanism for users to observe the usage of their data within the cloud. For instance, users got to be able to ensure that their information are handled in keeping with the service level agreements created at the time they register for services in the cloud. Standard access management approaches developed for closed domains like databases and operating systems, or approaches employing a centralized server in distributed environments, aren't appropriate, due to the following options characterizing cloud environments. First, data handling are often outsourced by the direct cloud service provider (CSP) to different entities within the cloud and theses entities can even delegate the tasks to others, and so on. Second, entities are allowed to hitch and leave the cloud in a very flexible manner. As a result, information handling within the cloud goes through a posh and dynamic graded service chain which doesn't exist in typical environments. To overcome the on top of issues, we have a tendency to propose a unique approach, particularly Cloud data answerableness (CIA) framework, supported the notion of data answerableness not like privacy protection technologies that are engineered on the hide-it-or-lose-it perspective, data answerableness focuses on keeping the information usage clear and track able. Our planned United States intelligence agency framework provides end-to end accountability in a very extremely distributed fashion. One of the

main innovative options of the United States intelligence agency framework lies in its ability of maintaining light-weight and powerful answerableness that combines aspects of access management, usage management and authentication. By suggests that of the United States intelligence agency, information house owners will track not solely whether or not or not the service-level agreements are being honored, however additionally enforce access and usage management rules as required. Related to the answerableness feature, we additionally develop 2 distinct modes for auditing: push mode and pull mode. The push mode refers to logs being periodically sent to the information owner or neutral whereas the pull mode refers to another approach whereby the user (or another approved party) will retrieve the logs as required. The design of the United States intelligence agency framework presents substantial challenges, together with un ambiguously distinguishing CSPs, ensuring the dependability of the log, adapting to an extremely localized infrastructure, etc. Our basic approach toward addressing these problems is to leverage and extend the programmable capability of JAR (JAR) files to mechanically log the usage of the users' information by any entity within the cloud. Users will send their information at the side of any policies like access control policies and work policies that they require to enforce, clathrate in JAR files, to cloud service suppliers. Any access to the information can trigger an automatic and authenticated work mechanism native to the JARs. We refer to this kind of social control as "strong binding" since the policies and also the work mechanism travel with the information. This sturdy binding exists even once copies of the JARs are created therefore, the user can have management over his knowledge at any location. Such suburbanized work mechanism meets the dynamic nature of the cloud however conjointly imposes challenges on ensuring the integrity of the work .To address this issue, we provide the JARs with a central purpose of contact that forms a link between them and also the user. It records the error correction info sent by the JARs, that permits it to monitor the loss of any logs from any of the JARs. Moreover, if a JAR isn't able to contact its central purpose, any access to its clathrate knowledge are going to be denied. Currently, we tend to specialize in image files since pictures represent a very common content kind for finish users and organizations and are increasingly hosted within the cloud as a part of the storage services offered by the utility computing paradigm featured by cloud computing. Further, pictures typically reveal social and personal habits of users, or are used for archiving vital files from organizations. Additionally, our approach will handle personal recognizable info provided they're stored as image files (they contain a picture of any matter content, for instance, the SSN hold on as a .jpg file). We tested our Central Intelligence Agency framework during a cloud tested, the Emulab tested, with Eucalyptus as middleware. Our experiments demonstrate the potency, measurability and granularity of our approach. Additionally, we tend to conjointly give a detailed security analysis and discuss the responsibility and strength of our design within the face of assorted nontrivial attacks, launched by malicious users or thanks to compromised Java Running setting (JRE).

2. LITERATURE SURVEY

2.1. Reliable Delivery and Filtering for Syslog

The Reliable Delivery and Filtering for Syslog feature allows a device to be customized for receipt of syslog messages. This feature provides reliable and secure delivery for syslog messages using Blocks Extensible Exchange Protocol (BEEP). Additionally, it allows multiple sessions to a single logging host, independent of the underlying transport method, and provides a filtering mechanism called a message discriminator. This module describes the functions of the Reliable Delivery and Filtering for Syslog feature and how to configure them in a network.

2.2. Guide to Computer Security Log Management

It provides practical, real-world guidance on developing, implementing, and maintaining effective log management practices throughout an enterprise. The guidance in this publication covers several topics, including establishing log management infrastructures, and developing and performing robust log management processes throughout an organization. The publication presents log management technologies from a high-level viewpoint, and it is not a step-by-step guide to implementing or using log management technologies.

2.3. Explorative Visualization of Log Data to support Forensic Analysis and Signature Development

In this paper, we propose an approach for log resp. audit data representation, which aims at simplifying the analysis process for the security officer. For this purpose audit data and existing

relations between audit events are represented graphically in a three dimensional space. We describe a general approach for analyzing and exploring audit or log data in the context of this presentation paradigm. Further, we introduce our tool, which implements this approach and demonstrate the strengths and benefits of this presentation and exploration form.

3. EXISTING SYSTEM

Data handling in the cloud goes through a complex and dynamic hierarchical service chain. This does not exist in conventional environments. Ordinary web framework Uses web services for request and responses.

3.1. Limitations

No security for user's data. No authentication or security provided High resource costs needed for the implementation. Not suitable for small and medium level storage users.

4. PROPOSED SYSTEM

In this paper, we propose a comprehensive solution for storing and maintaining log records in a server operating in a cloud-based environment. We address security and integrity issues not only just during the log generation phase, but also during other stages in the log management process, including log collection, transmission, storage, and retrieval. The major contributions of this paper are as follows. We propose architecture for the various components of the system and develop cryptographic protocols to address integrity and confidentiality issues with storing, maintaining, and querying log records at the honest but curious cloud provider and in transit.

5. ADVANTAGES

One of the main innovative features of the CIA framework lies in its ability of maintaining lightweight and powerful accountability that combines aspects of access control, usage control and authentication. Providing defenses against man in middle attack, dictionary attack, Disassembling Attack, Compromise JVM Attack.It's Suitable for limited and large number of storages.

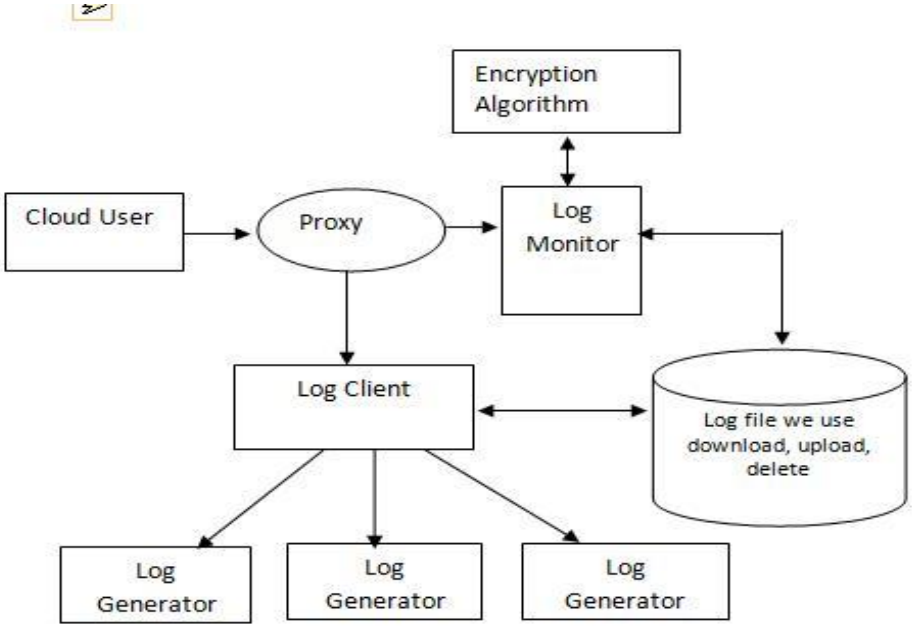

Fig1. *System Architecture*

Algorithm

Sing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

1. **Key Expansion -** round keys are derived from the cipher key using Rijndael's key schedule. AES requires a separate 128-bit round key block for each round plus one more.

2. **Initial Round**

   2.1. Add Round Key—each byte of the state is combined with a block of the round key using bitwise xor.

3. **Rounds**

   3.1. Sub Bytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.

   3.2. Shift Rows—a transpositi on step where the last three rows of the state are shifted cyclically a certain number of steps.

   3.3. Mix Columns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.

4. **Add Round Key 4.Final Round (no Mix Columns)**

   4.1. Sub Bytes

   4.2. Shift Rows

   4.3. Add Round Key

6. RELATED WORK

6.1. Log Generators

These are the computing devices that generate log data. Each organization hat adopts the cloud-based log management service has a number of log generators. Each of these generators is up to with logging capability. The log files generated by these hosts are not stored locally except temporarily till such time as they are pushed to the logging client.

6.2. Logging Client or Logging Relay

The logging client Isa collector that receives groups of log records generated by one or more log generators, and prepares the log data so that it can be pushed to the cloud for long term storage. The log data is transferred from the generators to the client in batches, either on a schedule, or as and when needed depending on the amount of log data waiting to be transferred. The logging client incorporates security protection on batches of accumulated log data and pushes each batch to the logging cloud. When the logging client pushes log data to the cloud it acts as a logging relay. We use the terms logging client and logging relay interchangeably. The logging client or relay can be implemented as a group of collaborating hosts. For simplicity however, we assume that there is a single logging client.

6.3. Logging Cloud

The logging cloud provides long term storage and maintenance service to log data received from different logging clients belonging to different organizations. The logging cloud is maintained by a cloud service provider. Only those organizations that have subscribed to the logging cloud's services can upload data to the cloud. The cloud, on request from an organization can also delete log data and perform log rotation. Before the logging cloud will delete or rotate log data it needs a proof from the requester that the latter is authorized to make such a request. The logging client generates such a proof. However, the proof can be given by the logging client to any entity that it wants to authorize.

6.4. Log Monitor

These are hosts that are used to monitor and review log data. They can generate queries to retrieve log data from the cloud. Based on the log data retrieved, these monitors will perform further analysis as needed. They can also ask the log cloud to delete log data permanently, or rotate logs.

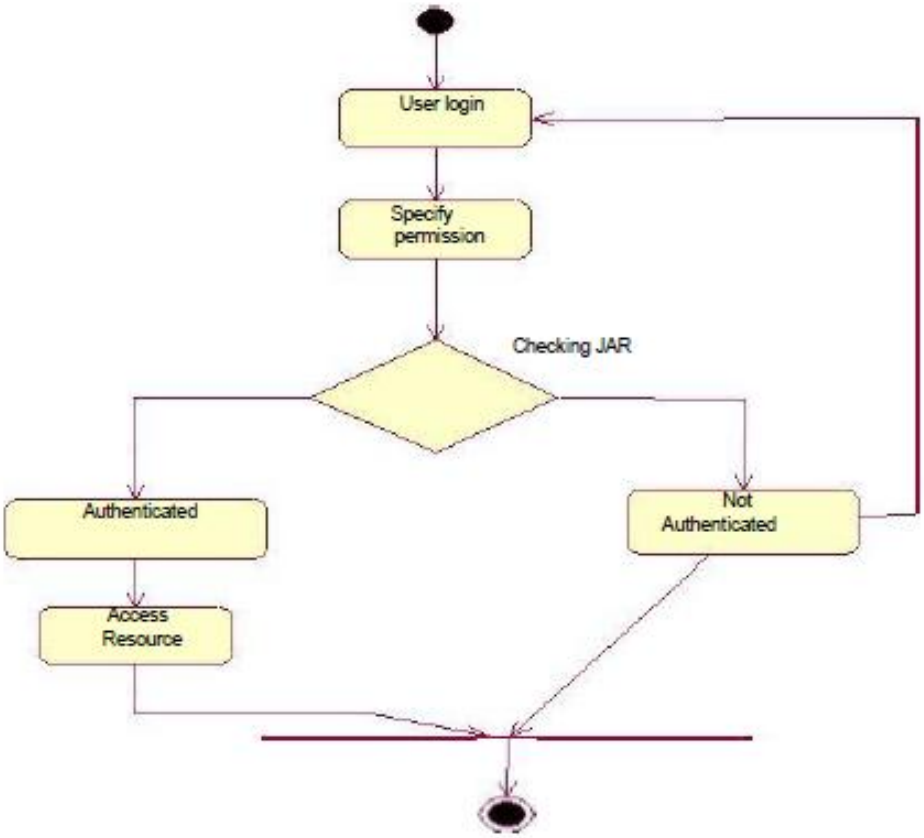

Fig2. *Flow Diagram*

7. CONCLUSION

We proposed a complete system to securely outsource log records to a cloud provider. We reviewed existing solutions and identified problems in the current operating system based logging services such as syslog and practical difficulties in some of the existing secure logging Techniques. In this work, find out the challenges for a secure cloud based log management service. The attackers use below three steps to hack. First, the attacker can intercept any message sent over the Internet. Second, the attacker can synthesize, replicate, and replay messages in his possession. And Last The attacker can be a legitimate participant of the network or can try to impersonate legitimate hosts. We implement how to store secure log file in cloud and that file we can change read, write, delete, upload and download. We can implement AES algorithm that uses for log monitor and log generator. We then proposed a comprehensive scheme that addresses security and integrity issues not just during the log generation phase, but also during other stages in the log management process, including log collection, transmission, storage and retrieval. One of the unique challenges is the problem of log privacy that arises when we outsourced log management to the cloud. Log information in this case should not be casually linkable or traceable to their sources during storage, retrieval and deletion. We provided anonymous upload, retrieve and delete protocols on log records in the cloud using the Tor network. The protocols that we developed for this purpose have potential for usage in many different areas including anonymous publish-subscribe.

FUTURE WORK

In the future, we plan to refine the log client implementation so that it is tightly integrated with the OS to replace current log process. In addition, to address privacy concerns current implementation allows access to log records that are indirectly identified by upload-tag values. We plan to investigate practical homomorphic encryption schemes that will allow encryption of log records in such a way that the logging cloud can execute some queries on the encrypted logs without breaching confidentiality or privacy.

REFERENCES

[1] 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.

[2] U.S. Department of Health and Human Services. (2011, Sep.).HIPAA—General Information [Online]. Available: https://www.cms.gov/hipaageninfo

[3] BalaBit IT Security (2011, Sep.). Syslog-ng—Multiplatform Syslog Server and Logging Daemon [Online]. Available: http://www.balabit.com/network-security/syslog-ng

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

[5] J. Kelsey, J. Callas, and A. Clemm, Signed Syslog Messages, Request for Comment RFC 5848, Internet Engineering Task Force, Network Working Group, May 2010.

[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. ACM Conf. Computer and Comm. Security, pp. 598-609, 2007.

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

[8] 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.

[9] B.Chun and A.C. Bavier, "Decentralized Trust Management and Accountability inderated Systems," Proc. Ann. Hawaii Int'l Conf. System Sciences (HICSS), 2004.

[10] Y. Chen et al., "Oblivious Hashing: A Stealthy Software Integrity Verification Primitive," Proc. Int'l Workshop Information Hiding, F. Petitcolas, ed., pp. 400-414, 2003.

[11] 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.

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

[13] D. New and M. Rose, Reliable Delivery for Syslog, Request for Comment RFC 3195, Internet Engineering Task Force, Network Working Group, Nov. 2001.

[14] M. Bellare and B. S. Yee, "Forward integrity for secure audit logs," Dept. Comput. Sci., Univ. California, San Diego, Tech. Rep., Nov. 1997.