International Research Journal of Engineering and Management Studies (IRJEMS)
Volume: 03 Issue: 05 | May -2019 ISSN: 2395-0126

Cloud Forensic Ensuring Cloud Log Soundness and Confidentiality

Kavyashree S, Dr Raghavendra Rao , Neha Najem , Ranjith J

Mtech Information Technology Department of Computer Science and Engineering,The National Institute of Engineering(NIE)
Professor Department of Computer Science And Engineering ,The National Institute of Engineering(NIE)
Mtech Information Technology Department of Computer Science and Engineering,The National Institute of Engineering(NIE)
Mtech Information Technology Department of Computer Science and Engineering,The National Institute of Engineering(NIE)

Abstract - *Client movement logs can be an important wellspring of data in cloud forensic examinations; consequently, guaranteeing the unwavering quality furthermore, trustworthiness of such logs is essential. Most existing answers for secure logging are intended for traditional frameworks instead of the intricacy of a cloud domain. In this paper, we propose the Cloud Log Assuring Soundness and Secrecy (CLASS) process as an elective plan for the verifying of logs in a cloud domain. In CLASS, logs are encoded utilizing the singular client's open key with the goal that just the client can unscramble the substance. So as to avoid unapproved alteration of the log, we create verification of past log (VPL) utilizing Rabin's unique mark and Bloom channel. Such a methodology diminishes check time fundamentally. Discoveries from our tests sending CLASS in OpenStack show the utility of CLASS in a genuine world setting.*

Key Words: Cloud log, Cloud log assuring soundness and secrecy scheme for cloud forensics, Verification of past log, Cloud forensic, openStack.

1. INTRODUCTION

Distributed storage, security and protection are reasonably built up research regions , which isn't astounding thinking about the across the board reception of cloud administrations and the potential for criminal misuse (for example trading off cloud records and servers for the taking of delicate information). Strangely however, cloud crime scene investigation [8-10] is a moderately less gotten subject. If a cloud administration, cloud server, or customer gadget has been bargained or engaged with noxious digital action (for example used to have unlawful substance, for example, radicalization materials, or lead disseminated refusal of administration (DDoS) assaults) [11, 12], agents should most likely direct criminological examination so as to "answer the six key inquiries of an episode – what, why, how, who, when, and where". Because of the inborn idea of cloud advances, regular computerized measurable methods and instruments need to be refreshed to hold a similar convenience and materialness in a cloud domain [14]. In contrast to an ordinary customer gadget, cloud virtual machines (VMs) can be bolstered by equipment that may be found remotely and along these lines would not be physically open (for example out of the jurisdictional domain) to a specialist.

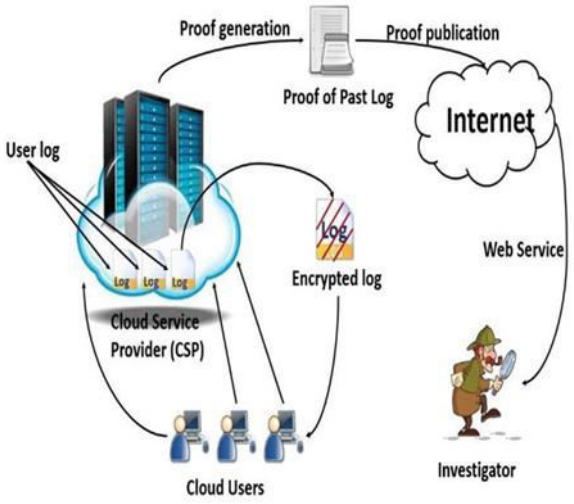

Fig. 1. Overview of CLASS scheme process

What's more, VMs can be disseminated over various physical gadgets in a grouped domain or they can exist inside a pool of VMs on the equivalent physical parts. Hence, catching the machine for criminological examination isn't suitable in many examinations. Moreover, information living in a VM might be unstable and could be lost when the power is off or the VM ends. Thus, the cloud specialist co-op (CSP) assumes a urgent job in the accumulation of evidential information (for example cloud

client's movement log from the log. For instance, the CSP composes the movement log (cloud log) for every client. Subsequently, counteracting alteration of the logs, keeping up a legitimate chain of guardianship and guaranteeing information security is urgent . This exploration considers "movement log information" as any recorded PC occasion that relates to a particular client. Such information must be kept up privately to preserver client protection and to encourage potential insightful exercises.

1.1 Threat model and security properties

In this section, we will describe some definitions required to understand our scheme, the threat model, an attacker's capability, possible attacks [29] , and the standard security properties that a secure cloud logging system must possess.

SUMMARY OF NOTATIONS

a) Log: Log can be network log, process log, registry log, application log or any customized text that meets the requirement of being stored for investigation purpose.

b) Log Chain (LC): LC is a small piece of information that coexists with its corresponding log in order to maintain the integrity and to prevent any modification of the log (such as addition, modification, deletion, and reordering).

c) Proof of Past Log (PPL): PPL is a signature or information about the actual log that will be available publicly for forwarding secrecy . That means if the system is compromised, an attacker cannot change the log without detection. PPL can be used to establish log veracity.

d)Cloud Service Provider (CSP): CSP is a cloud service provider in which a user can rent and use computing and storage resources. We assume that a CSP is honest but curious . That means it will serve according to contract agreement but has a curiosity about client activity. We design our (CLASS) scheme to include features to prevent a dishonest CSP.

e) User: User is a CSP client.

f) Investigator : An investigator is an individual or entity with legal authority to conduct investigative activities in response to some event. These activities include accessing and assessing the contents of log files supplied by a CSP. It is possible for an investigator to collude with a malicious user or CSP to manipulate the perception of an event.

g) Auditor: An auditor is an individual or entity who is authorized to verify the integrity of log entries, typically through techniques such as PPL. It is assumed that the auditor is always fully trusted.

f) Content Concealing (CC): CC is a strategy that helps to withstand against privacy breaches that are the result of collusion between a malicious cloud employee and an investigator.

g) Content Concealing Key (CC key): CC key is a pair of the private-public key that is used for concealing log content.CC key (the private key not the public key) should be shared by Shamir's or Blakley's secret sharing approach among some trusted entities.

1.2 Security Properties

CLASS seeks to achieve three properties of cryptography, namely: confidentiality, integrity, and authenticity, in terms of the following criteria.

Correctness: Cloud logs should reflect the correct history of a system's event with the occurring time. Any distortion to it is considered a violation of the correctness property.

Tamper Resistance: No one except real logger can introduce an invalid log entry as

a valid one. Any sort of contamination such as the addition of new log entries, modification or deletion of existing log entries or even reordering of log entries requires prevention. At a minimum a tamper resistant scheme prevents an attacker from modification of logs without detection.

Verifiability: Verification should be possible by both the user whose activity is represented in the log and the investigating entity. The auditor or any other party involved in the related litigation need to be able to establish log veracity.

Confidentiality: Log data contains sensitive user information and requires privacy protection. For example, if a user mistakenly puts their password into the username field, the system will record this as a failed sign-in attempt and store the password as username in the log. This illustrates the need for confidentiality for all logged data in addition to data more traditionally viewed as requiring privacy protections.

Admissibility: A secure cloud log should be maintained in a way that allows it to be admissible in a court of law for criminal prosecution. The features of log integrity (correctness and tamper resistance), a chain of custody, and forward secrecy all help to achieve such admissibility.

1.2 Threat Model

Our scheme is designed based on the "trust no one" policy. Any party among the CSP, investigator, and user, should be capable of protecting its own security and privacy against another party or collusion between other parties. Potential challenges to designing forensic enabled cloud logging have been discussed in a number of previous studies. For example, in an insecure cloud logging model, only the CSP can write to a log. An investigator or user can collude with the CSP to modify a log before or after publishing PPL. Thus, if a CSP falsely alters a log, whether in collusion with a malicious user or investigator or not, it can hinder the investigative process and conceal the truth of an event. This could result in an attacker failing to be identified or, more dangerously, attributing the attack to the wrong entity. Conversely, as the cloud is the host of multiple users, a malicious user can repudiate the log under investigation as his/her own log which can lead the criminal lawsuit to be dismissed. On the other hand, the log contains secretive data of the user and the user's privacy may be vulnerable due to this fact. A malicious investigator can alter the log before presenting to court authorities. Moreover, in collaboration with dishonest CSP or CSP employee(s), the investigator can violate the privacy of the user. Based on the above discussion, possible attacks on secure cloud log are given below:

Modification of Log: A dishonest CSP can modify the log before or after publishing its proof (PPL) upon or beyond collusion with the user or investigator. A malicious investigator may alter the log before presenting to court to save a dishonest user or to frame an honest user.

Modification of a log: can be of many forms, such as insertion of invalid entries, removal of the crucial entries, changing existing entries, reordering log entries to mislead the investigation and to hide malicious activities.

Privacy Violation: Leakage of a log file can reveal information that is able to be directly linked to a users' identity or is able to aggregate in such a way as to create such a link. Even with cryptographic security, cloud employees can transfer the log to an entity that has the key to decrypt (i.e. an investigator) and thus privacy violation may take place.

Repudiation of Ownership of Log: Cloud servers host many users. This presents the

possibility for a malicious cloud user to repudiate that the log files under investigation represent the activity of another user. On the other hand, a CSP can repudiate that it did not write the log under investigation. Likewise in SecLaaS, the CSP writes a log for every user and the user has no visibility regarding log entries. This may raise user suspicions regarding log veracity and credibility.

2. PROPOSED SCHEME: CLASS

In this section, we improve on SecLaaS and present CLASS scheme. We are assuming that in a cloud infrastructure, no party is trusted, that means an attack can come from any party: a CSP, user, or investigator. We are also assuming that cryptographic primitives work properly (i.e. if someone encrypts a message, then nobody can decrypt it without knowing the secret key).

2.1 System Overview

A dishonest cloud user can attack a system outside the cloud. They can also attack any application deployed in the same cloud or an attack can be launched against a node controller which controls all the cloud activities. For a virtual machine (VM), CLASS scheme takes the log from the node controller (NC), hides its content, and stores it in a database. This allows logs to become available for further investigation despite VM shutdown. Moreover, CLASS publishes its proof so that log integrity can be protected and admissibility ensured.

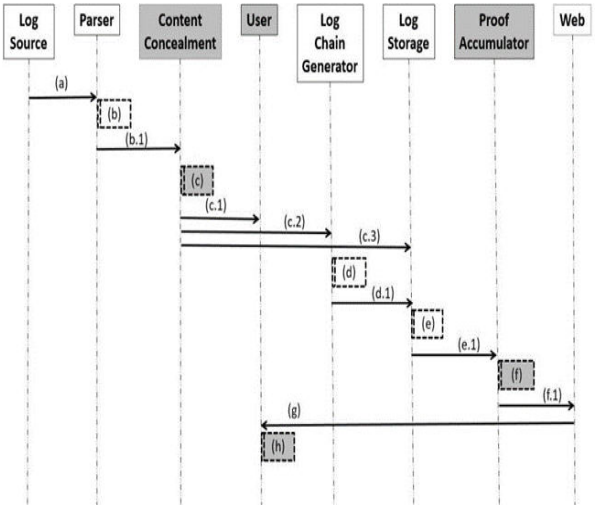

Fig. 2. Proposed CLASS scheme (new algorithms shaded)

3. CONCLUSIONS

In this paper, we proposed a secure logging scheme (CLASS) for cloud computing with features that facilitate the preservation of user privacy and that mitigate the damaging effects of collusion among other parties. CLASS preserves the privacy of cloud users by encrypting cloud logs with a public key of the respective user while also facilitating log retrieval in the event of an investigation. Moreover, it ensures accountability of the cloud server by allowing the user to identify any log modification. This has the additional effect of preventing a user from repudiating entries in his own log once the log has had its PPL established. Our implementation on OpenStack demonstrates the feasibility and practicality of the proposed scheme. The experimental results show an improvement in efficiency thanks to the features of the CLASS scheme, particularly in verification phase. Potential future extensions include the following:

1. Normally logs are low-level data and hard for the common user to understand what exactly those logs signify. Thus, we will explore leveraging big data techniques to facilitate user retrieval and visualization of information from log data.

Standardization of log format is also an associated research area.

2. To ease searching, we kept some crucial and sensitive information in plaintext format. This makes them vulnerable to be exposure. Thus, designing secure and efficient searchable encryption would extend this work.

3. There is also the need for an online credibility system designed to develop trust and credibility of a cloud user so that the CSP can enable stricter auditing policies for low-trust users in comparison to high-trust users.

4. Designing and implementing a prototype of the proposed scheme in collaboration with a real-world CSP, with the aim of evaluating its utility (e.g. performance and scalability) in a real-world environment.

REFERENCES

[1] X. Liu, R. H. Deng, K.-K. R. Choo, and J. Weng, "An efficient privacy-preserving outsourced calculation toolkit with multiple keys," IEEE Transactions on Information Forensics and Security, vol. 11, pp. 2401-2414, 2016.

[2] Y. Mansouri, A. N. Toosi, and R. Buyya, "Data storage management in cloud environments: Taxonomy, survey, and future directions," ACM Computing Surveys (CSUR), vol. 50, p. 91, 2017.

[3] M. Tao, J. Zuo, Z. Liu, A. Castiglione, and F. Palmieri, "Multi-layer cloud architectural model and ontology-based security service framework for IoT-based smart homes," Future Generation Computer Systems, vol. 78, pp. 1040-1051, 2018.

[4] Z. Xia, Y. Zhu, X. Sun, Z. Qin, and K. Ren, "Towards privacy-preserving content-based image retrieval in cloud computing," IEEE Transactions on Cloud Computing, pp. 276-286, 2018.

[5] L. Zhou, Y. Zhu, and A. Castiglione, "Efficient k-NN query over encrypted data in cloud with limited key-disclosure and offline data owner," Computers & Security, vol. 69, pp. 84-96, 2017.

[6] Q. Alam, S. U. Malik, A. Akhunzada, K.-K. R. Choo, S. Tabbasum, and M. Alam, "A Cross Tenant Access Control (CTAC) Model for Cloud Computing: Formal Specification and Verification," IEEE Transactions on Information Forensics and Security, vol. 12, pp. 1259-1268, 2017.

[7] L. Li, R. Lu, K.-K. R. Choo, A. Datta, and J. Shao, "Privacy-preserving-outsourced association rule mining on vertically partitioned databases," IEEE Transactions on Information Forensics and Security, vol. 11, pp. 1847-1861, 2016.

[8] K.-K. R. Choo, M. Herman, M. Iorga, and B. Martini, "Cloud forensics: State-of-the-art and future directions," Digital Investigation, pp. 77- 78, 2016.

[9] C. Esposito, A. Castiglione, F. Pop, and K.-K. R. Choo, "Challenges of Connecting Edge and Cloud Computing: A Security and Forensic Perspective," IEEE Cloud Computing, vol. 4, pp. 13-17, 2017.