

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/301894670>

# A Tamper Proof Log Architecture for Cloud Forensics

Article · January 2015

---

CITATIONS

0

---

READS

19

1 author:



**Geetha Venkat**

Pondicherry Engineering College

22 PUBLICATIONS 69 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Facial emotion recognition system for mobile devices [View project](#)



emotion recognition [View project](#)

All content following this page was uploaded by [Geetha Venkat](#) on 06 May 2016.

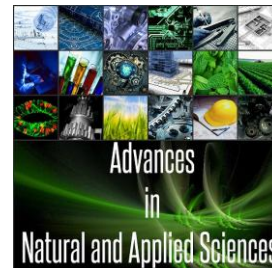
The user has requested enhancement of the downloaded file. All in-text references [underlined in blue](#) are added to the original document and are linked to publications on ResearchGate, letting you access and read them immediately.



AENSI Journals

## Advances in Natural and Applied Sciences

ISSN:1995-0772 EISSN: 1998-1090  
Journal home page: [www.aensiweb.com/ANAS](http://www.aensiweb.com/ANAS)



### A Tamper-Proof Log Architecture for Cloud Forensics

Geetha V., Udaya Ponni N.M., Devagi T.C. and Nandhini Priya M.

Pondicherry Engineering College, Department of Information Technology, Puducherry, India

#### ARTICLE INFO

##### Article history:

Received 12 October 2014

Received in revised form 26 December 2014

Accepted 1 January 2015

Available online 25 February 2015

##### Keywords:

Cloud Forensics

Forensic Investigation

Cloud Security, Logs.

#### ABSTRACT

Nowadays cloud computing is emerging as the best solution for small business owners to save money on infrastructure investment. Large organizations lent out their big data centers during slack time to generate more returns on the existing infrastructure. The small business owners avail the services from these datacenters. The trustworthiness of these clouds is very crucial in choosing a cloud service provider. The data provided by the cloud users or the cloud resources (Processor, storage and bandwidth) allocated to them should not be shared with other simultaneous users. If any security breach is suspected, evidence is needed to prove the cybernetic crime in the courts of law. Logs are used to record the activities taking place in a user session, which can be used as evidence later if crime is suspected. Then it becomes mandatory to provide secure and tamper proof log architecture so that the evidence can be preserved. This paper proposes a tamper proof architecture for securing logs in cloud architecture. The logs are created at IaaS layer of cloud as ultimately any user activity is mapped to machine operation. The performance of the proposed system is also analyzed and it works better than existing works.

© 2015 AENSI Publisher All rights reserved.

**To Cite This Article:** Geetha V., Udaya Ponni N.M., Devagi T.C. and Nandhini Priya M., A Tamper-Proof Log Architecture for Cloud Forensics. *Adv. in Nat. Appl. Sci.*, 9(6): 722-727, 2015

### INTRODUCTION

Security in public cloud is a very important research topic today. Cloud users share the public cloud resources spatially as well as temporally. This poses a security problem as the users may encroach on the other user spaces and embezzle their data and hardware resources. The cloud service provider may or may not be a part of this cyber crime. If the cloud service provider is reasonable, then he/she should provide some means of tracking the activities of the users to ensure whether they are working within their limit. Security comprises confidentiality and integrity of customer data and Cloud customers who process or store their data in the cloud want to have the same level of security as on a locally hosted system. This paper mainly relies on logs, as logging information becomes the fundamental and the most important part of applications. Every application has a contrast air of logging mechanism. A log is a record, as of the performance of a machine or the progress of an action undertaking or simply it lists the actions that have occurred. A well proposed logging system is a very useful utility for system administrators and developers, especially the support team. Logs save the support team and developers from the tedious work of tracking the users activities.

In the information and communication age, log files can be used as evidence in courts of law to show events that occurred for a certain timeframe, as long as they are proven to be accurate and untouched. For security professionals a log is used to record data on who, what, when, where, and why (W5) an event occurred for a particular device or application. As events on a device or application are taking place, processes are running that generate responses based on those events, and the responses are output into a log of one form or another. In a local environment, log files are not shared, but this is not the case in cloud. The way in which logs are generated from one device, can be easily read and displayed on another device. A developer of the device or application can format the contents contained in the log that has been generated. This in turn becomes a challenge in understanding the log output format for the users. Log has the capability to record all the events namely the W5 events, so, the log provides the security professionals the ability to monitor the activities of the application or device to ensure that all operations are done in correct manner. If any malicious activity is suspected, the log output can be reviewed when it comes for investigation to an incident. Lack of security is still a degree of speculation in cloud environment even though it has many benefits to offer. More particularly, there

**Corresponding Author:** Geetha.V., Department of Information technology, Pondicherry Engineering College, Puducherry, India.

are still questions to be answered relating to its ability to support forensic investigations. Through this paper we intend to highlight the basic information about logs and its architecture which provides tamper proof logs as evidence to the investigators dealing with any kind of forensic investigation.

The remainder of this paper is organized as follows: **Section II** provides background of the cloud forensics and logging in cloud environment. **Section III** provides related works and their techniques that can be adopted in this paper. **Section IV** describes the proposed system with tamper proof logs for checking the reliability of the provider. In **Section V** we provide a theoretical analysis of related works mentioned in this paper to that of ours and assess the performance of proposed work. **Section VI** concludes the paper with future enhancements.

#### *Background:*

As a cross between cloud computing and digital forensics, the term "cloud forensics" refers to the gathering of digital forensic data from a cloud infrastructure. Incident response and digital forensics have long been critical components of computer crimes investigations. Yet with the rapid evolution of cloud computing, these tasks have become challenging. The most reasonable option for many organizations is to generate logs on systems under their control in Infrastructure as a Service (IaaS) environment. This process is usually straightforward and conforms to the same types of standard logging practices employed within modern enterprise environments.

A user logging to a certain cloud service usually needs to send his data as well as associated access control policies to the service provider. After the information is received by the cloud service provider, the service provider will have authorized access privileges, such as read, write, and copy, on the data. In order to track the actual usage of the data, we aim to develop innovative logging techniques in such a way that the acquired evidence can only be read by the particular stakeholder. The following are the requirements which satisfy the above case:

1. The logging should be decentralized in order to adapt to the dynamic nature of the cloud.
2. Every access to the user's data should be automatically logged correctly. This should contain the basic and necessary information about logs including the timestamp.
3. Log files should be reliable and tamper-proof to avoid illegal insertion, deletion, and modification by unauthenticated parties.
4. Log files should be sent back to their data owners periodically to inform them of the current usage of their data and this is achieved through mail management.
5. Requested users get the logs by going through certain authentication techniques that ensure integrity and freshness of data.

The main aim of the project is to save the logs securely in cloud so that tampering activities are avoided and can be used as a perfect evidence for forensics investigation purposes; this enables the user to check the reliability of logs as well as the cloud provider. There are many types of attacks in cloud and these attacks can use huge amounts of computing resources; disables their usage by consumer efficiently. Assurance to the users account and password can be breached by several means. As a result, the subsequent stealing of confidential data or even the destruction of data can hamper the storage integrity and security of the cloud.

The complete details about logging have been discussed and our proposed architecture helps an investigator in acquiring all evidences (logs) by ensuring the reliability of the provider.

#### *Related Work:*

The growing adoption of cloud computing has not only introduced new interests in technology but also has raised many security issues, and one among them is logging, as logging information is one of the primary evidence in forensic investigation. Several researchers have explored this problem across multiple dimensions. Accorsi (2013) method devise a logging protocol for the storage phase and this paper is considered to be an extension of [Accorsi \(2009\)](#) BBox technique providing security in the entry level also. In essence, it aims at realizing tamper-proof evidence and promotes integrity.

Although several proposals for secure logging services exist, none of them ensures sufficient authenticity and [Accorsi \(2009\)](#) guarantees for log data storage while allowing the selective disclosure of log records to auditors.

[Marty \(2011\)](#) proposed a log management solution, which can solve several challenges of logging. In his solution, after enabling logging on all machine operations, he propose a synchronized, reliable, bandwidth efficient, and encrypted transport layer to transfer log from the source to a central log collector. Final step deals with ensuring the presence of the desired information in the logs. The proposed guideline tells us to focus on when to log, what to log, and how to log. The answer of when to log depends on the use cases, such that business relevant logging, operations based logging, security (forensics) related logging, and regulatory and standards mandates.

Meng et. al., (2005) proposed a more cost-efficient solution to store log files on a separate server by using virtual machines. The nature of Xen monitor makes it very difficult for hackers to detect, thus keeping the log files safe.

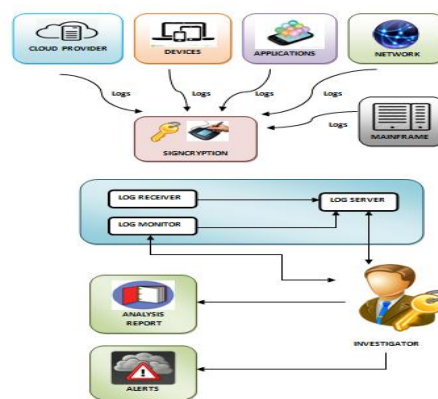
Ma and Tsudik (2009) propose a logging protocol for the storage phase that employs a novel authentication technique called “Forward-Secure Sequential Aggregate” which aims to address both key exposure and storage efficiency issues.

Waters, et. al., (2004) proposed encrypted and searchable audit log. This showed how identity-based encryption (IBE) can be used to make audit logs efficiently searchable. Keywords which relate to each log entry are used to form public keys in an IBE system. Administrators allow searching and retrieval of entries matching a given set of keywords by issuing clients the corresponding IBE private keys. They recommended the use of the Bellare and Yee’s technique (1997) as their authentication scheme.

All the above mentioned works provided support to secured log storage. Each of them employs one or more security techniques to secure log storage. But there is no security in log creation, modification and transmission. This paper aims to address this limitation by providing security for all possible log operations.

#### Proposed System:

Figure 1 shows the architecture of the proposed system. The architecture can be applied for forensic purpose in case of any criminal offence. A tamper evident storage is also being done by the use of hash chains to detect tampering attempts. Logs from various places are collected. We use a combination of digital signature and encryption techniques called *Signcryption* (Zheng,1997) which is done logically in a single step. Signcryption is more secure and it ensures that the message sent couldn’t be forged, the contents of which are confidential and ensures non-repudiation. Investigator uses special retrieving techniques like usage of queries and special keywords to acquire the logs for investigation purpose.



**Fig. 1:** Proposed Tamper-Proof Framework.

This tamper proof log architecture can be used to trace the hardware information. The cloud service provider, data owner, all other cloud users and investigator are the stakeholders. The logged on information is stored in cloud and can be viewed only by authorized people. Security is provided by using encoding and decoding techniques to ensure data integrity and freshness. Since the application runs on a cloud environment, information about users, their logs and other system and application information are maintained and stored in cloud. Stakeholders can retrieve log information from cloud only by using their own ID’s and they have to go through some security process for seeing the logs they have requested. This paper discusses a secure logging framework, rights of each stakeholders present and guidelines that provide a proactive approach to logging to ensure that the data needed for forensic investigations has been generated and collected in a secure way. It also ensures that the stakeholders cannot tamper with the logs present in the cloud.

The standardized framework eliminates the need for logging stakeholders to reinvent their own standards. In addition, they ensure that log consumers can effectively and easily analyze, process, and correlate the emitted log records. The devices generate log files that help information security professionals to research and analyze security incidents.

The functionalities provided by our framework are as follows:

1. **Reliable data (logging) origin:** Events from authorized Devices are recorded in log files for provenance information.
2. **Encrypted Records:** Log records are encrypted using unique evolving keys for forward secrecy.
3. **Keyword based retrieval of records:** Using special keywords and queries, Investigator can fetch Log records.

4. **Classifying Logs:** Generated logs are classified based on the importance level of logs. Encryption or Decryption techniques are possible and the logs can be checked and verified for identity and authentication purposes.

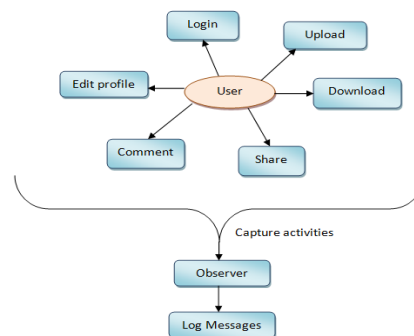
5. **Managing Logs:** An application needs to be deployed on each system to perform the requested service, collect logs, encrypt them and send them to log server.

6. **Processing and Storing Logs:** It should be able to obtain logs, store useful information to the database (for future use) and/or returns data reports to investigators.

Our proposed system helps to address the security issues by building a secure log architecture so that tampering of logs by the stakeholders during the retrieval of logs are avoided. Detailed explanation of each module is explained below:

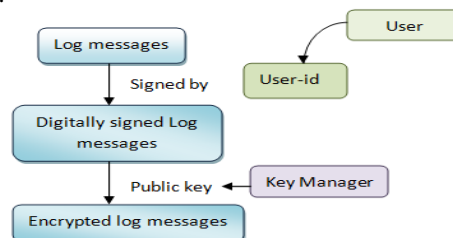
#### 1. Device logger/monitor module:

Device logger monitors users or devices activity by capturing every event from that device. The captured activity (log message) has to be transferred to the log collector for future investigation. In this module the user can login, upload, download, share, comment and edit profile. These activities are captured by the system and are generated as log messages.



**Fig. 2:** Device logger/monitor module diagram.

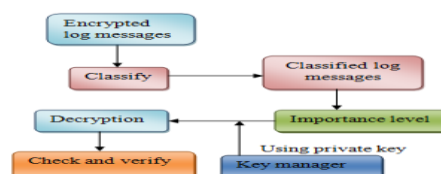
#### 2. Message Signcryption module:



**Fig. 3:** Signcryption Module diagram (Zheng,1997).

To maintain the identification and to avoid identity-based attacks, we make digital signature for the log messages and then encrypt them using a public key encryption methodology (Zheng,1997). The user with his ID signs log messages and digitally signed log messages are obtained. Public key encryption technique which is a cryptographic system that uses two keys - a public key known to everyone and a private or secret key known only to the recipient of the message. These keys are managed by the key manager and the public keys are used to encrypt the log messages.

#### 3. Log Appender module:

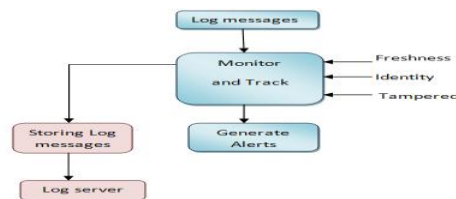


**Fig. 4:** Log appender module diagram.

This module receives log messages from the application running in different servers and then classifies and distinguishes them based on the category and their importance level after being decrypted and checked for identity. Encrypted log messages are being sent as input and these messages are being classified according to the importance level of the log messages. Decryption technique is being applied to these classified log messages and investigator can check and verify these messages later. Key manager manages the private key here.

#### 4. Tamper tracker module:

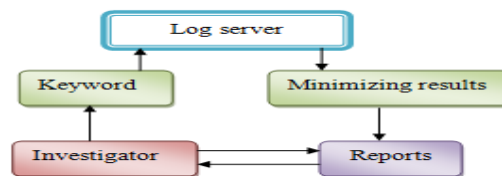
Tamper tracker in figure 5 continuously monitors and tracks the incoming log messages and verifies its integrity and freshness. If any tampering occurs at any of the levels it will be reported and recorded for further investigation. Log messages are being monitored and tracked based on its freshness and identity. The tracked messages are being stored in a log server.



**Fig. 5:** Tamper tracker module diagram.

#### 5. Keyword Log miner module:

A retrieving mechanism by providing appropriate keywords like device id, timestamp, priority, category etc., based on the criteria given. This log miner is only capable of retrieving log messages based on keyword mining of encrypted messages which is used by investigators on addressing various investigations and researches. In this module, the investigator may use special keywords to retrieve the log messages from the log server for his investigation purpose.



**Fig. 6:** Keyword Log miner module diagram.

**Table 1:** Comparison between existing methods and the proposed method.

Existing Methods							
Security Requirements							
Secure logging methods by	Transmission phase				Storage phase		
	confidentiality	authentication	integrity	uniqueness	accountability	integrity	confidentiality
Schneier/Kelsey	x	x	x	x	✓	x	✓
R. Accorsi	x	x	x	x	✓	✓	✓
Marty	✓	✓	✓	✓	x	x	x
Meng et. Al	x	x	x	x	x	✓	✓
Waters <i>et al.</i>	x	x	x	x	x	✓	✓
Ma/ Tsudik	x	x	x	x	✓	✓	✓
Proposed method							
Tamper Proof Log Architecture	✓	✓	✓	✓	x	✓	✓

#### Discussion:

Schneier–Kelsey(1999) logging protocol proposals focus on log data at rest and this scheme uses hash chains to create dependencies between log entries. Hence, removing one or more log entries from the “chain” makes tampering detectable to verifiers. The primary limitation of this work is that, an attacker can seize control of an insecure machine and simply continues creating log entries, without trying to delete or change any previous log entries.

R. Accorsi (2013) presents a simple extension of a scheme called BBox technique which ensures security in the storage phase, he fails to give security in the entry level. Accorsi (2009) have also created methodologies for Safekeeping Digital Evidence with Secure Logging Protocols which tells about the importance of log data as an evidence in court and his work explains the security issues relating to log which remains unclear. He also



elucidates a subset of the necessary secure requirements for digital evidence and extensively surveys the state of the art secure logging protocols, thereby demonstrating that none of the current protocols completely fulfills the elucidated requirements for admissible evidence. In analyzing the shortcoming of logging protocols, he also elaborates on the related research challenges. Marty's (2011) work tells about a logging framework and guidelines that provide a forehanded approach to logging to ensure that the data needed for forensic investigations has been generated and collected. The standardized framework in Marty's work eliminates the need for logging stakeholders to reinvent their own standards. These guidelines make sure that critical information associated with cloud infrastructure and software as a service (SaaS) use-cases are collected as part of a defense in depth strategy. In addition, they ensure that log consumers can effectively and easily analyze, process, and correlate the emitted log records. Ma and Tsudik (2009) propose a logging protocol for the storage phase that employs a novel authentication technique called "Forward-Secure Sequential Aggregate" which aims to address both key exposure and storage efficiency issues. Table 1 compares the existing methods with that of the concept which have been discussed in this paper which satisfies all security requirements ensuring the reliability of the provider also. This paper just gives a theoretical analysis of the current work by comparing it with the previous methodologies.

#### Conclusion:

The battle to improve our log deciphering skills will continue on well into the future of the information age as more and more devices get wired into the global networks. Logs are critical to Information Security professionals, as they are your forensic trail telling you the W5 of an event, managing and protecting the logs should be part of daily procedures as they are extremely valuable. Within the field of cloud forensics, more research and planning needs to be done, along with the implementation of industry standard law practices.

Currently, rules, regulations, guidelines, and standard practices vary greatly from provider to provider. This makes it increasingly difficult for forensic technicians to work. As further work, we intend to address and investigate the trustworthiness of logging architecture that is vulnerable to attacks by providing utmost security to the field of cloud forensics.

#### REFERENCES

- Accorsi, R., 2009. Safe-keeping digital evidence with secure logging protocols: State of the art and challenges, *Proceedings of the IEEE Conference on Incident Management and Forensics, IEEE Computer Society*, 94–110.
- Rafael Accorsi, 2013. A secure log architecture to support remote auditing, *Mathematical and Computer Modelling, Elsevier* 57(7-8): 1578-1591.
- Adelstein, F., 2006. Live forensics: Diagnosing your system without killing it first. *Communications of the ACM*, 49(2): 63-66.
- Bellare, M. and B. Yee, 1997. Forward integrity for secure audit logs, University of California, San Diego, Dept. of Computer Science & Engineering, Technical Report.
- Ma, D., G. Tsudik, 2009. A new approach to secure logging, *ACM Transactions on Storage*, 5 (1): 1–21.
- Marty, R., 2011. Cloud application logging for forensics. In *proceedings of the 2011 ACM Symposium on Applied Computing*, 178–184.
- Meng, J., X. Lu, G. Dong, 2005. A novel method for secure logging system call. *Proceedings of ISCIT2005*, 924-927.
- Schneier, B. and J. Kelsey, 1999. Secure audit logs to support computer forensics. *ACM Trans. Inf. Syst. Security*, 2(2): 159-176.
- Waters, B.R., D. Balfanz, G. Durfee, D.K. Smeters, 2004. Building an Encrypted and Searchable Audit Log. *ACM Annual Symposium on Network and Distributed System Security*.
- Zheng, Y., 1997. Digital Signcryption or How to Achieve Cost (Signature & Encryption) << Cost (Signature)+ Cost (Encryption). In *Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology*, 165-179. Springer-Verlag.