FaaSeC: Enabling Forensics-as-a-Service for Cloud Computing Systems

[1]Dipak Yadav, [2]Ritu Kumari, [3]Bhavya, [4]D.Vandana, [5]C.Yosepu

[1,2,3,4]B.Tech, [5]Associate Professor
Department of Computer Science and Engineering,
St. Martin's Engineering College, Hyderabad, India

Abstract: **Cloud Computing has challenges to digital *forensic* evidence in the cloud. Digital forensics is a critical technology for obtaining evidences in crime investigation. Nowadays, the overwhelming magnitude of data and the lack of easy-to-deploy software are among the major obstacles in the field of digital forensics. However, to support forensic examination efficiently using cloud, one has to overcome many challenges such as lack of understanding and experiences on configuring and using digital forensic analytic tools by the investigators, and lack of interoperability among the forensic data processing software. Performing investigation in the cloud environment is practically possible only if support from the Cloud Service Provider (CSP) is made available. Our proposed model-FaaSeC can extend the forensic support from CSP and makes to provide Forensics-as-a-Service (FaaS) to the investigator. The experimental results show that the proposed approaches can significantly reduce forensic data analysis time by parallelizing the workload. The overhead for the investigators to design and configure complex forensic workflows is greatly minimized. The proposed workflow management solution can save up to 87% of analysis time in the tested scenarios.**

***IndexTerms* –Cloud computing, digital forensics**

I.Introduction

Cloud computing has the potential to become one of the most transformative developments in the history of computing. The growth of cloud market has reached beyond the expected. It benefits the end users by providing uninterrupted services at lesser cost and with reduced maintenance overheads. But the recent attacks reported in the cloud raise several questions on its security. These security breaches caused trust deficit in the cloud. Two possible solutions exist in this context. One is to improve the security of the existing algorithms and the second solution is to perform forensic investigation in the cloud. In this paper, our interest is on the latter.

We found that till date, there is no vendor which facilitates the forensic investigation in the cloud environment. There are various legal and technical reasons behind cloud provider's unwillingness to provide FaaS to the third party personnel. The main cause is its multi-tenant nature as the third party investigator may have a chance to acquire other tenant's data during forensic investigation. This leads to privacy violation of the corresponding users and is treated as an offense. Our solution considers the above issue and increases the chance of facilitating FaaS to the third party personnel by the CSP.

The third party investigator may be trusted or untrusted. In this paper, we handle the worst case scenario i.e. when the investigator is not trusted and given access to the cloud infrastructure, there are high chances that he/she may perform suspicious activities. The untrusted investigator may be internal to the cloud organization as part of incident first responders team or can be an external entity. Once he/she is given access to the cloud infrastructure, there are high chances of evidence tampering. This indeed leads to generate a forensic report with misleading conclusions. So, we propose that CSP facilitating FaaS should know the events/activities being performed by the investigator at the cloud end.

II. Literature survey

Ensure monitoring and auditing is targeted to detect and deter major incidents

Aside from collecting data to support post incident investigations. It should be noted that monitoring plays a vital role in preventing or detecting security incidents that may be in progress. Establishing a monitoring and auditing facility such as an intrusion detection system would allow organizations to respond to and minimize the consequences of security threats and incidents.

Specify circumstances when escalation to a full formal investigation is required

If a suspicious event is triggered or manually detected, such as detected intrusion or failed access events, the event needs to be reviewed and a process has to be established to decide which of the detected events need to be followed up with formal investigations and escalated to management for further action. This will involve an impact assessment of the event and the cost of investigation.

A Study on Event Reconstruction Methodologies used for Forensic Analysis

Criminal investigations today can hardly be imagined without the forensic analysis of digital devices. This leads to an increasing number of cases with an ever-growing amount of data that exceeds the capacity of the forensic experts. To support investigators to work more efficiently, an approach to automatically reconstruct events that previously occurred on the examined system is required and hence reconstructing the sequence of computer events that led to a particular event is an essential part of the digital investigation process. Reconstructed events will assist in forensic inferences of evidence and traces caused by an action invocation in a system subject to forensics investigation. Currently, when a system is compromised, event reconstruction involves manual sifting for clues based on the current state of the system and the log files and so there is a demand for explaining the sequence of digital events, and a tool to automatically reconstruct the events and produce a timeline. This paper deals with the different event reconstruction methodologies for forensic analysis during investigation.

An automated timeline reconstruction approach for digital forensic investigations

Existing work on digital forensics timeline generation focuses on extracting times from a disk image into a timeline. Such an approach can produce several million ‘low-level’ events (e.g. a file modification or a Registry key update) for a single disk. This paper proposes a technique that can automatically reconstruct high-level events (e.g. connection of a USB stick) from this set of low-level events. The paper describes a framework that extracts low-level events to a *SQLite* backing store which is automatically analyzed for patterns. The provenance of any high-level events is also preserved, meaning that from a high-level event it is possible to determine the low-level events that caused its inference, and from those, the raw data that caused the low-level event to be initially created can also be viewed. The paper also shows how such high-level events can be visualized using existing tools.

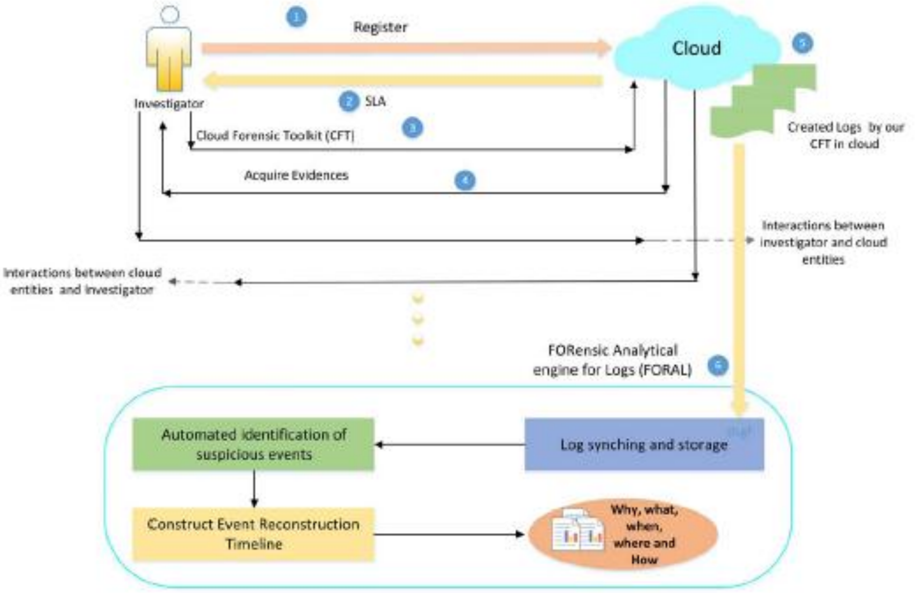

1
Register
Cloud
5
Investigator
2 SLA
3
Cloud Forensic Toolkit (CFT)
Created Logs by our CFT in cloud
Acquire Evidences
4
Interactions between investigator and cloud entities
Interactions between cloud entities and investigator
FORensic Analytical engine for Logs (FORAL)
6
Automated identification of suspicious events
Log synching and storage
Construct Event Reconstruction Timeline
Why, what, when, where and How

Figure 1: Proposed FaaS model for the cloud environment (FaaSeC)

III. MODULES

- **Data User**
- **Investigator Module**
- **CSP**
- **COPS and SEMS Model**

Data User:

In the first module, we develop the Data Owner Module. Owner Will Signup and upload data to cloud server with private key and encryption. After Getting key Owner can authenticate data using key, and upload any data related to users to cloud.

In this module, data owner will check the progress status of the file upload by him/her. It has large data needed to be stored and shared in cloud system. In our scheme, the entity is in charge of viewing his own data and executing Encrypt and key generation operation. After the completion, owner logout the session.

Investigator Module:

We develop Investigator Module. Investigator Will register with application and request for registration is sent to CSP for verification. After CSP accepts investigator request then only he can login in to application.

Investigator can view data of any owner which are uploaded by respective owner but he can't view data directly without verification from data owner.

Investigator will send data view request to data owner who will respond to request and authenticate by sending key which is used to decrypt data.

CSP:

We develop cloud service provider module which will handle authentication and verification of investigator . CSP can view all uploads of different users. CSP can't view user data as it is in encrypted state. He don't have permission to view user data.

CSP will view investigator log files with time and owner name. Taking this data as input for both algorithms (Sems and Cops) CSP will analyze which investigator is attacker . Based on the output of algorithms he will finalize attackers and normal investigator and unauthorized attackers.

COPS and SEMS Model:

We use TKS to initially get the top-k frequent item sequences. Then SEMS is applied to get the suspicious sequences. Say, the CSP is interested to know the suspicious sequences in CFI log, then each new sequence in the log during Time Window T is compared with the frequent item sequences (freqseq). If a mismatch occurs, the percentage of fraction left (per fractionLeft) will increase and if it is more than the user threshold (thseq) value then it is considered as suspicious sequence.

We can also get the suspicious sequences of a cloud forensic application using conditional probability. In COPS we use two method for conditional probability

1. Threshold value checks 2. Key mismanagement.

PROPOSED METHODOLOGY

We designed a comprehensive forensic process such that the chances of CSP providing forensic services to the investigator would increase (ii) The transparency in the cloud forensic process is improved by creating forensic logs at the cloud end. (iii) We propose two approaches namely SeMS and CoPS which can automate the detection of suspicious events/processes from forensic logs at the cloud end.

Using SEMS and Cops we can find suspicious sequences from cloud forensic application.

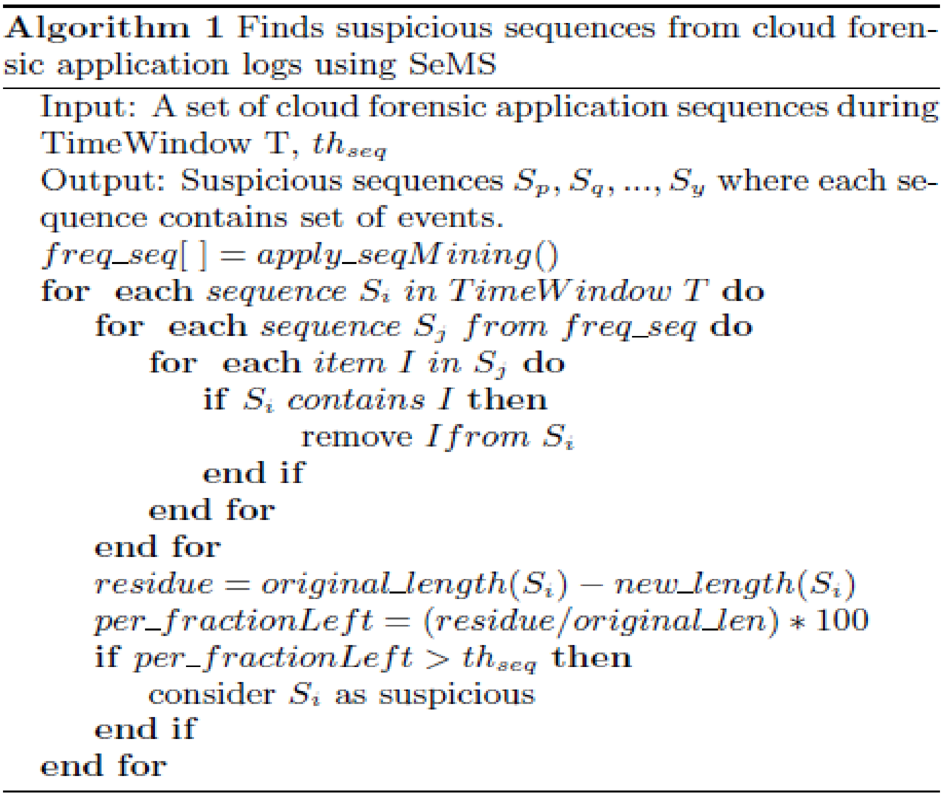
ADVANTAGES

SeMS and CoPS are designed in such a way that, they can be applied to detect suspicious sequences in any application log.

FaaSeC Model gives the complete process of providing forensic as a service starting from the investigator registration to report generation.

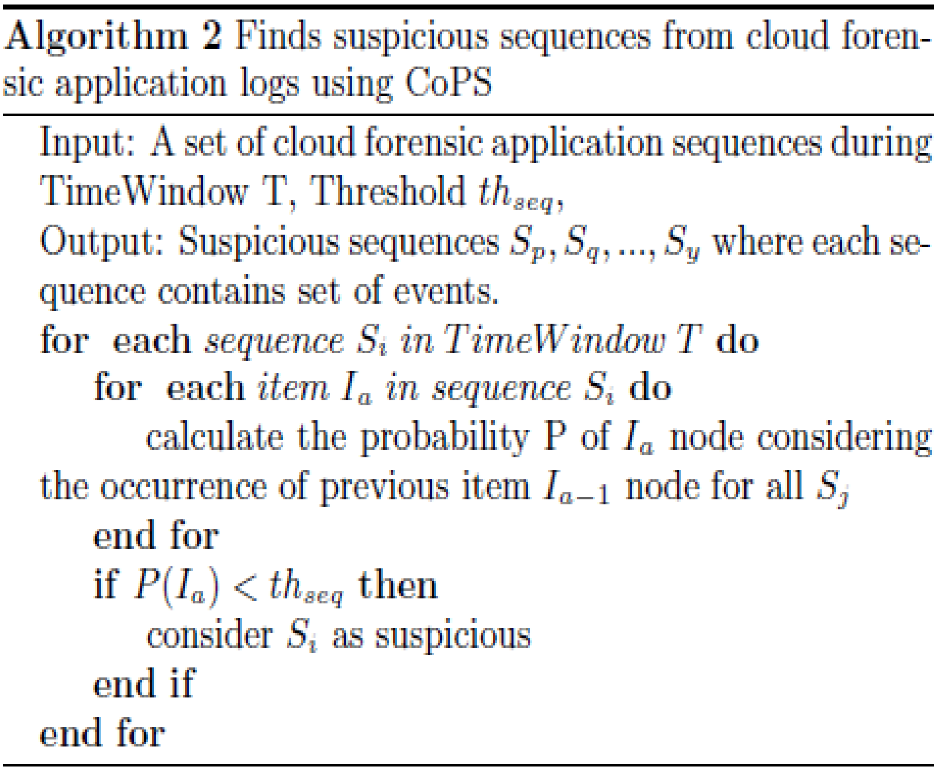
The suspicious events from the CFI log were identified automatically without much human effort.

IV. ALGORITHM



Algorithm 1 Finds suspicious sequences from cloud forensic application logs using SeMS

```
Input: A set of cloud forensic application sequences during
TimeWindow T, $th_{seq}$
Output: Suspicious sequences $S_p, S_q, ..., S_y$ where each sequence contains set of events.
$freq\_seq[\,] = apply\_seqMining()$
for each $sequence\ S_i\ in\ TimeWindow\ T$ do
    for each $sequence\ S_j\ from\ freq\_seq$ do
        for each $item\ I\ in\ S_j$ do
            if $S_i\ contains\ I$ then
                remove $I from\ S_i$
            end if
        end for
    end for
    $residue = original\_length(S_i) - new\_length(S_i)$
    $per\_fractionLeft = (residue/original\_len) * 100$
    if $per\_fractionLeft > th_{seq}$ then
        consider $S_i$ as suspicious
    end if
end for
```



Algorithm 2 Finds suspicious sequences from cloud forensic application logs using CoPS

```
Input: A set of cloud forensic application sequences during
TimeWindow T, Threshold $th_{seq}$,
Output: Suspicious sequences $S_p, S_q, ..., S_y$ where each sequence contains set of events.
for each $sequence\ S_i\ in\ TimeWindow\ T$ do
    for each $item\ I_a\ in\ sequence\ S_i$ do
        calculate the probability P of $I_a$ node considering
the occurrence of previous item $I_{a-1}$ node for all $S_j$
    end for
    if $P(I_a) < th_{seq}$ then
        consider $S_i$ as suspicious
    end if
end for
```

V. RESULTS

The evaluation of the proposed medical information retrieval system follows the standard TREC evaluation method for ad hoc retrieval tasks. Documents with high relevance were selected for experts to decide the relevance as "not relevant", "possible relevant" and "definitely relevant". These samples were used to evaluate the performance of the systems. The evaluation is done for the performance of three systems, the baseline of system, the PRF system that incorporated incremental PRF and the proposed system that has both incremental PRF and Knowledge based query expansion system. The results demonstrated that the proposed system is more efficient and accurate than the existing baseline system. The results demonstrated that the proposed system improves the performance of the system and outperforms the existing system. As an active research area a variety of areas are explored recently to optimize the performance of medical information retrieval system.

VI. CONCLUSION

Recent attacks in the cloud systems show the importance of performing forensic investigation in such environments. Forensics in the cloud environment is at a nascent stage and requires the cloud provider support for facilitating FaaS. We proposed a new Cloud Forensic Service model called FaaSeC. This model creates the forensic application log in the cloud from which the CSP can know the activities performed by the third party investigator. For forensic analysis, identifying the suspicious events plays a significant role and we find those events from the cloud forensic application log using SeMS and CoPS. We also compared both the approaches in terms of execution time and memory consumption.

REFERENCES

[1] A. of Chief Police Officers, "Good practice guide for computer based electronic evidence," ACPO, Tech. Rep.

[2] K. Kent, S. Chevalier, T. Grance, and H. Dang, "Guide to integrating forensic techniques into incident response," National Institute of Standards and Technology, Tech. Rep.

[3] J. Dykstra and A. T. Sherman, "Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques," Digital Investigation, vol. 9, 2012, pp. S90– S98.

[4] "Sleuth Hadoop," http://www.sleuthkit.org/tsk hadoop/, retrieved April 2013.

[5] P. Mell and T. Grance, "The NIST definition of cloud computing," http: //csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf.

[6] J. Erickson, M. Rhodes, S. Spence, D. Banks, J. Rutherford, E. Simpson, G. Belrose, and R. Perry, "Content-centered collaboration spaces in the cloud," IEEE Internet Computing, vol. 13, September 2009, pp. 34–42.

[7] D. D. Roure, C. Goble, and R. Stevens, "The design and realisation of the myexperiment virtual research environment for social sharing of workflows," Future Generation Computer Systems, vol. 25, no. 5, 2009, pp. 561 – 567.

[8] I. Foster, "Globus online: Accelerating and democratizing science through cloud-based services," Internet Computing, IEEE, vol. 15, no. 3, May-June 2011, pp. 70 –73.

[9] S. Caton and O. Rana, "Towards autonomic management for cloud services based upon volunteered resources," Concurrency and Computation: Practice and Experience, 2011.

[10] S. Distefano, V. D. Cunsolo, A. Puliafito, and M. Scarpa, "Cloud@home: A new enhanced computing paradigm," in Handbook of Cloud Computing, B. Furht and A. Escalante, Eds. Springer US, 2010, pp. 575–594.

[11] A. Chandra and J. Weissman, "Nebulas: using distributed voluntary resources to build clouds," in Proceedings of the 2009 conference on Hot topics in cloud computing. USENIX Association, 2009.

[12] S. Xu and M. Yung, "Socialclouds: Concept, security architecture and some mechanisms," in Trusted Systems, ser. Lecture Notes in Computer Science, L. Chen and M. Yung, Eds. Springer Berlin / Heidelberg, 2010, vol. 6163, pp. 104–128.

[13] "Amazon EC2," http://aws.amazon.com/ec2/, retrieved April 2013.

[14] Y. Song, H. Wang, Y. Li, B. Feng, and Y. Sun, "Multi-tiered on-demand resource scheduling for vm-based data center," in Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, ser. CCGRID '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 148–155.

[15] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," Commun. ACM, vol. 51, Jan. 2008, pp. 107–113.