Authentication Framework in Forensic Science with Cloud Computing

Ankit J Faldu, Parag C Shukla

Atmiya Institute of Technology and Science- Department of MCA- Rajkot, Gujarat India

ankitfaldu@yahoo.com
paragshukla007@gmail.com

Abstract - **Cloud computing has emerged pattern that pulls peoples. On this paper, the necessity for understanding where, how and when knowledge is either processed or saved in database, becomes a chief interest because of the continually constructing discipline of cloud computing forensics. In this paper we describe in a distinctive manner a predominant part of our cloud forensic framework that may be built on high of both new and current knowledge - the logging phase. We discuss the problems that have got to be dealt with in such architectures and we detail our proposed options to them. We explain how our architecture and findings can support forensics investigators that habits investigations in a cloud environment.**

***Keyword*: Cloud Computing, Forensic Science, Logging Framework, Data Science**

I. INTRODUCTION

In view that its construction, cloud computing technology offered itself to the customers as a way in which they would employ quite a lot of quantities of computing vigor beneath the type of virtual machines, intermediate platform detailed to developers or equipped to make use of functions for mass utilization. The technologies surrounding it have evolved with quality %, however, we can find a customary main issue inside all of them - cloud computing safety.

However this is not ample in our current digital world, as we retailer increasingly information remotely, in cloud techniques. Hackers, malware and all other web threats are actual menaces for our data. For this reason, authorized investigators need to have a method in which they can screen the endeavor of a detailed virtual computer and the program related [5]. The obstacle that they face on this case is typically concerning jurisdiction because the cloud data is most of the time break up across multiple knowledge facilities, over multiple countries or even continents. On 2d case, current cloud infrastructures are inclined to depart this clever part away and best monitor virtual machines for efficiency instead than what's going down inside of them.

On this paper we describe a new manner of monitoring exercise in cloud environments and information centers utilizing a secure cloud forensic framework. We speak about the architecture of this type of framework and how can or not it's utilized on top of new or current cloud computing deployments. Also, for testing and gathering results we implemented this answer on our previous developed cloud computing procedure.

The rest of the paper is structured as follows. In section 2 we gift some of the associated work in this field, linked to our subject and in section three we element our proposed cloud forensics logging framework. Part four is committed to presenting our results from our present implementation, and in part 5 we conclude.

II. RELATED WORK

The subject of cloud logging, as a support for forensics, can also be beginning to emerge along with the ones offered before. On this guidelines, we find theses, such because the one among Zawoad et al which reward in [1] an architecture for a relaxed cloud logging service. They speak about the necessity of log gathering from various sources around the datacenter or hypervisors with a purpose to create a everlasting photo of the operations achieved in a datacenter and so they present an structure that can be used for serving to strengthen this discipline.

The equal challenges are evidenced by using Marty [2] and Sibiya et al [3]. The paper discusses a logging framework and offers a sequence of instructional materials that can furnish assurance to forensics investigators that the information has been reliably generated and picked up and suggest a standardized approach to do logging, to be able to have a single, centralized logging collector and processor, as a consequence saving time and money for each businesses and customers.

Throughout our study we all for making a choice on an intermediate illustration of the data that's dispatched between the local and vital forensic modules. Now we have analyzed specific current met languages for logging.

The primary one is the “management met language” [10] proposed by the UnixWare neighborhood. Its expertise is that it may be used as a transparent API in the kernel modules as it provides an interface for an external host. The drawback is that it wishes a variety of auxiliary binary data to be sent as a way to re-create the complete image on the other end, and utilizing it we get speedily a traffic better than the one that can be obtained with the aid of sending only the elemental snapshots. That is because

that this met language is designed for use best in the community over a process.

On the opposite facet, the CEE (normal occasion Expression) group [9] proposes a suite of requisites utilizing the JSON and XML markup languages for event going surfing disk or in transit over a network. These requisites are designed for maximum interoperability with current event and interchange necessities to reduce adoption bills. The talents of this procedure is that CEE expresses its interfaces and does now not promote an precise implementation. Utilizing the understanding gathered from these met languages, and due to the fact that that our work is novel within this field, we provide our possess logging met language, custom tailor-made for our wishes. We depend our implementation on the CEE requirements and take additionally into consideration the strategies made by way of prior research on this discipline [8]. As met language layout we use the JSON markup because of its well-known utilization in contemporary internet functions or distributed techniques. It's viewed a lightweight substitute to XML markup and can be used as a backend in our cloud forensic modules, which might be going to be provided later on this paper.

III. LOGGING FRAMEWORK ARCHITECTURE

In this section we describe the highest view architecture of a cloud enabled forensics approach, commencing with the general inspiration that our system implements after which focal point on the logging section. We are able to detect a modular structure and each and every of the modules is awarded in detail and it is convenient to peer that the whole framework can also be multiplied with different modules or plugins.

A) With the intention to have a superior working platform, we need to first introduce the inspiration of a cloud computing forensics framework. The architecture described in this section is detailed in our earlier work [4], [6] and here we in short reward the foremost ingredients. As will also be obvious in figure 1, the top view of a cloud forensic framework comprises two primary layers: the virtualization layer and the administration layer.

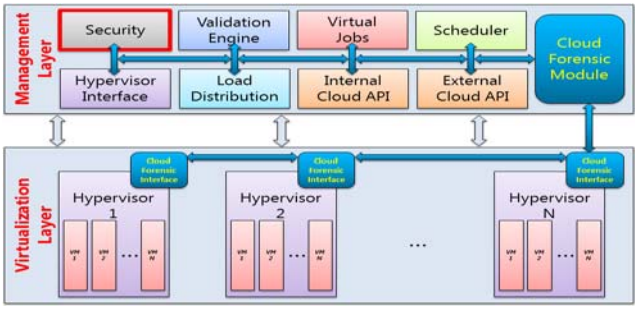

Figure 1. Forensic Enabled Cloud Computing Architecture

In the virtualization layer we discover the genuine platforms and servers that host the virtual machines and have virtualization enabled hardware. In the administration layer we discover the modules liable for enabling the entire operations detailed to the cloud. These modules are, in order: protection (responsible with all security concerns related to the cloud system - intrusion detection and alarming module), Validation engine (receives requests to add new jobs to be processed), virtual jobs (creates an abstraction between the data requested through the consumer and the payload that have to be delivered to the cloud system), Scheduler (schedules the jobs to the virtualization layer), Hypervisor interface (acts like a translation layer that's particular to a virtualization program seller), Load distribution (accountable with horizontal and vertical scaling of the requests bought from the scheduler), inside cloud API (meant as a hyperlink between the virtualization layer and the cloud procedure) and external cloud API (offers a strategy to the user to engage with the method).

For the forensics materials mainly we implemented a specific module, the Cloud Forensic Module. Its predominant intention is to acquire all forensic and log data from the virtual machines which are strolling throughout the virtualization layer. Additionally, we must attribute to the protection module better responsibilities and allow it to be in contact with all the other modules within the management layer.

B) Now that the inspiration of a cloud forensics procedure as a whole has been presented, on this part we discuss the important aspect within the Cloud Forensic Module - the logging framework.

Our cloud method makes use of the suggestion of leases [6], in which we can specify the period of time the job have to run, or specify between which hours of a day it runs. To achieve our goal, we brought a new power rent, accountable solely with the logging phase, that in the course of the day, when the datacenter is typically occupied by means of the customers, makes use of a minimum number of nodes and for the period of the night time, when the datacenter is close to fully free, it mechanically scales up to use a highest number of nodes.

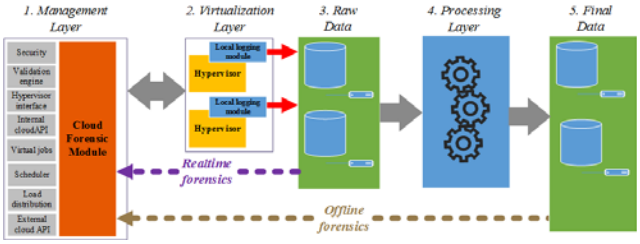

Figure 2. Cloud Forensics Logging Framework

The structure is a layered one, containing five layers, each with its own rationale. We additionally represented this in a graphical means, in determine 2 the layers and the relationship between them. The layers are all carried out using the disbursed computing paradigm and truely, they're jobs in our cloud computing environment. We favored this architecture because it is natively scalable on top of existing data centers or laptop networks and it is competent to handle large quantity of data and connected consumers.

In addition, with a view to be certain that the info is kept safely and no one can tamper it, every operation made by means of the approach goes through a hashing algorithm, both long-established, resulting and diff documents. These hashes are saved encrypted using a passphrase furnished by way of the investigator. The first layer, as provided in our earlier work [4], represents the management layer in a cloud computing deployment. It comprises the modules in charge with enabling the entire operations specified to the cloud. We are able to additionally see the previously stated Cloud Forensic Module.

The 2nd layer represents the virtualization layer in a cloud computing deployment. Its motive is to contain the genuine workstations that host the digital machines and have virtualization enabled hardware. A committed "local logging module" have got to be installed into the existing physical laptop. It is responsible with the raw information gathering from the monitored virtual machines. The information quantity can be adjusted by using the investigator and he can prefer to monitor a specified digital computer or screen the complete activity present inside that machine.

In order to collect knowledge reliably from the digital machines the neighborhood logging module need to be built-in fully with the running hypervisor within the bodily computing device. In this paper we center of attention on the combination with the "KVM" virtualization science that exists in brand new Linux kernel releases. We have chosen it due to the fact that it's a full open-supply virtualization answer, built-in with the Linux kernel on the grounds that 2007 and it's actively utilized by many corporations the world over.

An major factor that we have to think about is which information are we intercepting from the digital laptop and then ship to additional processing. Considering that all this exercise can also be intercepted, there exists the hazard of severe time penalties and processing velocity. In order to resolve this trouble, at this point we present the likelihood for an investigator to decide upon the logging degree for a certain virtual computing device. That is beneficial when you consider that that, for instance, an investigator handiest wishes to analyse the digital reminiscence for its contents, and it is not occupied with virtual disk photographs or virtual network pastime. Additionally at this step we have to recall the situation of community transmission overhead.

The third layer represents a storage layer for the raw data despatched from the nearby logging modules current within the virtualization layer. The logging modules ship uncooked information, within the type they are gathered from the hypervisor. Accordingly, this deposit has the operate of a dispensed storage and it comprises a sequence of nodes, each running a database. We've got chosen this procedure in order to create a bendy and scalable layer architecture that may face the info site visitors coming from the upper layer.

For the reason that the data that's going to be sent from the physical
virtualization host to the primary forensic management unit can attain an essential dimension, we implement a mechanism of "diff" between two portions of data. For illustration, if an investigator wishes to investigate a digital desktop reminiscence over a period, the regional forensic module is going to ship just one initial reminiscence picture and after that simplest what has been converted will likely be sent. Of course we will use the entire talents of the host and provide a neighborhood aggregation module that pre-tactics the information accrued earlier than sending it to the vital forensic module. This procedure is new for the subject of cloud computing forensics and we bear in mind it as a excellent approach to slash the influence over the network.

The process runs in the following manner. Initially the logging modules send a reference file after which, at an user defined time interval, the modules respond with a delta file, that represents the difference between the prior reference file and the present state. Accordingly, it implements a photo mechanism on the hypervisor level. We now have chosen this strategy because we wish to present to the forensic investigator the likelihood to have an photo of what is going down within a digital machine between two snapshots. This feature is presently no longer to be had in other hypervisors, equivalent to VMware's; in their case we are able to have a picture at time t0 and one at time ti, however we are not able to comprehend the state of the digital machine between the 0 and that i step.

This residue has additionally a different rationale. In case of severe emergency, the forensic investigator can see an actual-time evolution of the monitored virtual computer via issuing an immediate connection to this sediment. This selection is made on hand by means of the Cloud Forensic Module, which has the capability to by means of-move usual uncooked knowledge processing.

The fourth layer has the purpose of analysing, ordering, processing and aggregating the information stored within the previous layer. Due to the fact all these steps are computing intensive, the whole analysis approach is made in an off-line manner and is available to the investigators as soon as the job is able. After this entire method the investigator has a full photograph of what occurred over the monitored far off digital machine in a manner such because the one encountered in application source code variation tools, hence enabling him to navigate backward and forward into the history of the virtual desktop.

This residue is carried out additionally as a allotted computing software. We've got chosen this strategy because of the processing power wants that our framework demands, more exactly it desires to do correlations between exclusive snapshots in a reasonable period of time.

The fifth layer represents the storage of the results published by using the previous layer. A forensic investigator interacts with the monitored virtual computer snapshots at this residue, via utilising the Cloud Forensic Module from the management layer.

IV. RESULTS

In this part we're going to present important points concerning the results we accumulated after imposing our Cloud Logging modules. For testing, the modules have

been split across multiple workstations, as confirmed in figure 3.

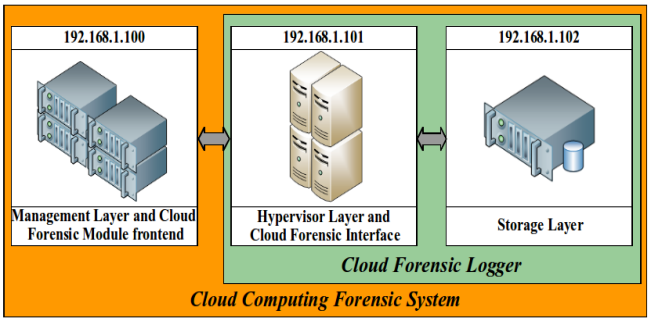

Figure 3. Mapping Modules to Workstations.

They are represented as a cluster of servers, each having the performance designated in the structure section. As it may be visible, the entire modules located within the perimeter, referred to as "Cloud Computing Forensic process", can also be ran together on one computer. Factors like network switches aren't represented so as not to burden the image, however the IP addresses of the hosts are saved. In our configuration, we used three precise workstations, each and every having the functionalities and network addresses provided within the determine.

The hardware platform that we used consists of an AMD Phenom II X6, 6 cores, 8GB RAM, RAID0 configured rough disks walking KVM as hypervisor and QEMU as a hypervisor interface, an Intel DualCore, 4GB RAM as the storage layer and an AMD C-60 DualCore, 4GB RAM as the management layer. The community we used is 10/a hundred MB.

The software platform used for conducting experiments contains regular Linux running procedure with KVM as a hypervisor, QEMU and libvirt as communique driver between the hypervisor and our measuring tools.

The tests had the intention of measuring the virtual laptop used reminiscence by the use of digital RAM snapshots, and the digital desktop storage by means of virtual DISK snapshots. The digital machines had been strolling more than a few program stacks, from bare kernel and just a FTP server, to a stack composed from Apache HTTP server, Hypertext Preprocessor environment and a PostgreSQL database strolling an on-line content management approach like Django, Drupal or Plone. The process of recording digital machines' activity used to be remodeled a period of several hours, at a time step of 10 minutes. The CPU load when conducting files making use of all the 6 cores used to be about 20%.

The outcome are fascinating, if we take into consideration the technologies utilized by KVM internally. For example, RAM snapshots are made thoroughly from host machine RAM and don't include necessarily consecutive bodily or virtual RAM areas; moreover they're made efficiently by means of KVM and every image has a few tens or hundred of megabytes in measurement.

However, inside our experiments, the DISK snapshots had been the biggest, attaining even gigabyte in dimension and those grew to be the main target for our logging process.

Bellow you will find the precise tests that have been made for the DISK snapshots. We used digital computing device photos starting from a 10MB size and as much as 6GB. Table I and determine 4 gift the info collected from our modules and the tine needed to method the complete knowledge. Also, in table I we will in finding small print regarding the digital machine operating system and the application stack being in use on the second of trying out. The switch time between the Cloud Forensic Interface module and the Storage module shouldn't be viewed, as being a consistent time, of about eighty two seconds for a 800 MB file.

We are able to see that the time needed for our process at this second is proportional with the size of the digital computer DISK picture and it grows exponentially together with it. This is the reason, even though our carried out modules in charge with logging are performing in just right stipulations, the outcome at this step are usually not very enough. However, they are required so as to establish a base level for assessment. The important conclusion after checking out our implementation is that, to be able to fulfil the desired performance for our cloud forensic system, we'd like a dedicated software for computing variations, peculiarly tailor-made for the usage with digital machines snapshots [7].

V. CONCLUSION

On this paper we provided a novel resolution that presents the digital forensic investigators with a nontoxic and at ease procedure in which they may be able to screen user undertaking over a Cloud infrastructure.

Our work is inquisitive about growing reliability, defense, protection and availability of Cloud Computing methods. The characteristics of such systems reward problems when tackling with secure resource administration because of its heterogeneity and geographical distribution. We described the design of a hierarchical architectural model that allows for investigators to seamlessly analyse workloads and digital machines, while keeping scalability of massive scale allotted systems.

REFERENCES

[1] S. Zawoad, A.K. Dutta and R. Hasan, "SecLaaS: Secure Logging-as-a- Service for Cloud Forensics", Symposium on Information, Computer and Communications Security (ASIACCS), 2013

[2] R. Marty, "Cloud Application Logging for Forensics", Proceedings of the 2011 ACM Symposium on Applied Computing, 2011

[3] G. Sibiya, H. Venter and T. Fogwill, "Digital forensic framework for a cloud environment", Proceedings of the 2012 Africa Conference, 2012

[4] A. Patrascu and V. Patriciu, "Beyond Digital Forensics. A Cloud Computing Perspective Over Incident Response and Reporting", International Symposium on Applied Computational Intelligence and Informatics, 2013

[5] A. Amarilli, D. Naccache, P. Rauzy and E. Simion, "Can a program reverse-engineer itself?", Proceedings of the Thirteenth IMA International Conference on Cryptography and Coding, 2011

[6] A. P˘atra¸scu, C. Leordeanu, C. Dobre and V. Cristea, "ReC2S: Reliable Cloud Computing System", European Concurrent Engineering Conference, Bucharest, 2012.

[7] A. P˘atra¸scu, I. Bica and V. Patriciu, "Enhanced diff for high performance forensic enabled cloud infrastructures", The 13th International Conference on Informatics in Economy, Education, Research and Business Technologies, Bucharest, 2014.

[8] T.Sang, Y. Du, P. Qin and J. Du, "A Log Based Approach to Make Digital Forensics Easier on Cloud Computing", Third conference on Intelligent System Design and Engineering Applications, 2013

[9] Common Event Expression Log Syntax Specification (2014, January 28) [Online].
Available: http://cee.mitre.org/language/1.0-beta1/cls.html

[10] UnixWare Management Metalanguage (2014, January 28) [Online].
Available:
http://uw714doc.sco.com/en/UDI_spec/m_mgmt.html

[11] Logging Framework for Cloud Computing Forensic Environments, Alecsandru P˘atra¸scu1;2 and Victor-Valeriu Patriciu3 Advanced Technologies Institute, Bucharest, Romania