International Journal of Computer Science and Information Technology Research (IJCSITR)
Vol. 3, No. 1, January-December, 2022, pp. 145-153
Journal ID: 9471-1297
website: www.ijcsitr.com
DOI: https://doi.org/10.63530/IJCSITR_2022_03_01_015

Automated Incident Response: AWS Lambda Forensic Snapshots, Velociraptor, and Timesketch for Timeline Analysis

Sandhya Guduru
Masters in Information Systems Security,
Software Engineer - Technical Lead, USA.

Abstract

This paper proposes a serverless, automated incident response framework tailored for cloud-native environments. The workflow integrates AWS Lambda for EBS snapshot acquisition, Velociraptor for live memory collection, and Timesketch for timeline analysis, enhanced by YARA-based signature matching and VirusTotal API enrichment. It addresses the shortcomings of traditional forensic approaches—manual workflows, delayed evidence capture, and fragmented analysis pipelines—by offering a modular and scalable architecture. By automating key stages of forensic triage and integrating open-source tools with cloud-native triggers, this framework improves detection speed, operational efficiency, and investigative accuracy in elastic infrastructure environments.

Keywords

Automated Incident Response, AWS Lambda, Digital Forensics, Velociraptor, Timesketch, YARA, VirusTotal API, Memory Forensics, Serverless Computing, Cloud-Native Security

How to Cite: Sandhya Guduru. (2022). Automated Incident Response: AWS Lambda Forensic Snapshots, Velociraptor, and Timesketch for Timeline Analysis. *International Journal of Computer Science and Information Technology Research*, 3(1), 145–153.

DOI: https://doi.org/10.63530/IJCSITR_2022_03_01_015
Article ID: IJCSITR_2022_03_01_015

Introduction

Modern digital infrastructures are increasingly built on cloud-native principles, featuring on-demand scalability, ephemeral workloads, and distributed computing environments. These characteristics present new challenges for digital forensics and incident response (DFIR), which traditionally depend on manual intervention, persistent storage, and static systems for evidence collection and analysis.

In particular, serverless computing has emerged as a paradigm that allows code to run in response to events without provisioning infrastructure. AWS Lambda exemplifies this model, enabling on-demand execution of forensic tasks such as snapshot acquisition and malware scanning [1]. Similarly, Amazon Elastic Block Store (EBS) provides block-level storage volumes for EC2 instances, and its snapshots allow investigators to preserve disk state for forensic imaging [2].

Tools like Velociraptor allow for real-time memory acquisition and remote querying of endpoints to capture volatile memory artifacts. Its forensic capabilities make it suitable for large-scale memory triage in both cloud and on-premise systems [3]. Once data is collected, timeline analysis tools like Timesketch reconstruct the sequence of events, enabling analysts to visualize attacker behavior over time [4].

YARA rules are used to enhance detection precision to identify known malware signatures within disk and memory images, while the VirusTotal API facilitates real-time IOC (Indicator of Compromise) enrichment using aggregated threat intelligence from global sources [5].

Despite the availability of these tools, most organizations rely on fragmented, reactive forensic workflows that delay response, increase analyst burden, and risk the loss of ephemeral evidence [6]. This paper proposes an integrated and automated approach that unifies these components through serverless orchestration—addressing the limitations of traditional DFIR while aligning with the operational models of modern infrastructure.

Literature Review

Automated incident response combines digital forensics capabilities with the speed and scalability of cloud-native infrastructure. In this context, serverless computing, memory forensics, and timeline analysis become core components in building a modern threat detection and response framework. This review defines and discusses the key technologies used in such

frameworks: AWS Lambda, Amazon EBS snapshots, Velociraptor, Timesketch, YARA rules, and the VirusTotal API.

AWS Lambda is a serverless computing service provided by Amazon Web Services that allows functions to run in response to triggers without provisioning or managing servers. Lambda supports event-driven architectures and is ideal for automating cloud forensics tasks such as snapshot creation and artifact collection. Its utility in digital forensics has been explored in cloud-native investigations, especially due to its scalability and low operational overhead

Amazon Elastic Block Store (EBS) snapshots provide incremental backups of volumes attached to EC2 instances and are often used in forensic workflows to capture disk state at a specific time. Researchers have explored using EBS snapshots for post-incident data preservation, citing the method's minimal intrusion and consistency guarantees [7].

Velociraptor is an open-source endpoint visibility and response tool that enables live memory acquisition, forensic artifact collection, and scalable querying. It is designed for enterprise-grade digital forensics and incident response (DFIR). Velociraptor's query language (VQL) allows practitioners to retrieve memory-resident indicators of compromise (IOCs) across distributed environments in real time

The use of memory forensics in incident response has been long established, with tools like Volatility and Rekall setting the stage for advanced frameworks like Velociraptor. A comparative analysis of live response tools noted Velociraptor's unique blend of memory inspection and scalability for cloud-native infrastructures.

Timesketch is an open-source timeline analysis tool developed by Google, designed to help analysts explore forensic timelines collaboratively. It supports ingestion of events from multiple sources (e.g., Plaso, Syslog, JSON) and provides search, tagging, and visualisation capabilities Its timeline correlation abilities are essential in reconstructing attack paths and identifying patterns over time.

In large-scale investigations, timeline analysis enables analysts to correlate logs, artifacts, and alerts across disparate systems. As shown in prior case studies, the integration of timeline tools into SOC workflows increases the accuracy and speed of forensic analysis. At the National Center for Computer Crime Data (NCCC), investigators used timeline analysis to reconstruct user activity leading up to a triple homicide. By correlating digital traces with physical evidence, they established a clear timeline that was pivotal in solving the case [8].

YARA rules are pattern-matching rules used for identifying malware or suspicious

content in files, memory, or network traffic. They are especially valuable in forensic analysis and malware classification, allowing security professionals to detect specific traits associated with known threats. The power of custom YARA rules lies in their flexibility to define conditions based on strings, byte patterns, and logic, as demonstrated in malware detection efforts across enterprise systems [9].

VirusTotal, acquired by Google, is a free service that analyses files and URLs for malware using multiple antivirus engines and tools. It offers a public and private API, allowing integration into automated workflows for threat intelligence and IOC validation. Various studies focusing on cloud-assisted malware triage have evaluated its role in digital forensics [10].

When combined, YARA rules and the VirusTotal API allow incident response teams to scan snapshots or memory dumps against known IOCs, expediting triage decisions. Research has shown that this integration can reduce mean-time-to-detect (MTTD) by correlating local findings with global threat intelligence [9].

Problem Statement

The challenges of conducting digital forensics and incident response in modern cloud-native environments can be better understood by breaking them down into key problem areas. The following table summarises the major limitations currently faced by security teams operating in dynamic, distributed infrastructures.

Issue	Summary
Cloud Complexity	Cloud-native environments are dynamic and distributed, complicating forensic evidence collection.
Lack of Automation	Manual snapshot and memory dump processes are too slow for timely incident response.
Live Memory Acquisition	Difficult to perform at scale in Linux environments; workflows remain fragmented.
Timeline Fragmentation	Event data is scattered across tools and logs, delaying root cause analysis.
Tool Sprawl	SOCs rely on multiple, disconnected tools, increasing workload and response time.
Delayed IOC Matching	YARA and threat intelligence tools are not integrated into forensic pipelines.
No Unified Workflow	There's no seamless, automated system that ties together all stages of forensic response.

Traditional digital forensics and incident response (DFIR) workflows struggle to keep pace with the speed and complexity of modern cloud-native infrastructures. These environments are characterized by elastic compute resources, ephemeral workloads, and distributed systems, which pose unique challenges for evidence acquisition and incident triage. Manual forensic practices are often too slow and labor-intensive to respond effectively to real-time threats in dynamic environments.

One major challenge is the lack of automation in acquiring forensic snapshots and memory dumps in cloud contexts. Investigators traditionally rely on manual scripts or agent-based tools, which introduce latency and risk contaminating volatile data. In environments such as AWS, disk volumes (e.g., EBS) may be deleted or modified rapidly, making timely, automated capture critical for preserving evidence [11].

Another limitation lies in the difficulty of live memory acquisition at scale, particularly in Linux-based cloud instances. While Velociraptor and similar tools have improved memory collection, many enterprise environments still lack seamless workflows to integrate live forensics into broader incident response pipelines [12].

Moreover, timeline analysis remains fragmented across toolsets and logs, often requiring manual correlation by analysts. This slows down root cause analysis and allows attackers more time to persist in the environment. While tools like Timesketch provide visualization and search features, they require structured data inputs and integration with upstream forensic tools to be effective [13].

The absence of a unified, serverless workflow that integrates disk snapshots, live memory collection, and automated timeline analysis results in disjointed investigations. Many Security Operations Centers (SOCs) face tool sprawl, where forensic artifacts must be manually exported, scanned, and correlated—a process that increases response time and analyst fatigue

Additionally, although YARA rules and threat intelligence platforms like VirusTotal are widely used, they are rarely embedded directly into forensic pipelines. This lack of integration delays IOC matching and limits the ability to detect known threat signatures in real-time.

In sum, the current landscape lacks a cohesive, cloud-native approach to automated incident response that unifies forensic snapshotting, memory analysis, timeline correlation, and IOC matching. This gap results in delayed detection, fragmented analysis, and, ultimately, longer threat dwell times in critical systems.

Proposed Solutions

This paper proposes an integrated, serverless forensic workflow to address the shortcomings outlined above. The table below highlights each component of the proposed solution and how it contributes to faster, more effective incident response in cloud environments.

Solution	Summary
Serverless Snapshotting	Use AWS Lambda to automate real-time EBS snapshot creation during incidents.
Scalable Memory Collection	Deploy Velociraptor to automate RAM and process data acquisition across systems.
Timeline Automation	Use tools like Plaso and Timesketch in a serverless workflow to build forensic timelines.
IOC Matching	Automate YARA scanning and integrate with threat intelligence platforms for enrichment.
Workflow Orchestration	Connect all components using automation tools like AWS Step Functions or Terraform.

Serverless Snapshot Acquisition with AWS Lambda

AWS Lambda can be used to automate the acquisition of Amazon EBS snapshots during security incidents. By configuring CloudWatch Events or GuardDuty alerts as Lambda triggers, investigators can initiate snapshot creation in real-time when anomalies are detected. This eliminates the need for manual intervention while ensuring timely data preservation. Because EBS snapshots are incremental and immutable, they provide reliable artifacts for later forensic analysis without impacting instance performance [14].

Scalable Live Memory Collection with Velociraptor

Velociraptor can be deployed as a lightweight agent to collect memory snapshots from EC2 instances or on-prem systems for volatile data acquisition. Using Velociraptor Query Language (VQL), analysts can automate the retrieval of RAM contents, system call traces, and suspicious processes across thousands of endpoints.

Automated Timeline Analysis with Timesketch

Captured disk and memory, artifacts can be processed with Plaso to extract event data and ingested into Timesketch for timeline reconstruction. Lambda functions can be chained to

automatically process snapshot volumes using forensic tools in containers (e.g., Plaso, log2timeline), and then push results to Timesketch via its REST API. Timesketch supports tagging, collaborative annotation, and visualization, accelerating root cause analysis in SOCs.

IOC Matching via YARA and VirusTotal API Integration

YARA rules can be executed against snapshot data or memory images to enhance detection capabilities. Using Lambda, analysts can automate the scanning of forensic artifacts for known signatures of malware or advanced persistent threats (APTs) [14]. Matched artifacts can then be sent to the VirusTotal API for further enrichment, enabling real-time classification and contextualization using a wide range of antivirus engines and community IOCs

Workflow Orchestration and Automation

To link all stages, a combination of AWS Step Functions or Terraform can be used to orchestrate the end-to-end workflow. Triggered by a detection rule (e.g., GuardDuty or an SIEM alert), the workflow initiates Lambda-based EBS snapshots, schedules Velociraptor memory collection, processes artifacts with Plaso, updates Timesketch, and performs IOC matching. This pipeline ensures rapid, reproducible responses and reduces the cognitive load on analysts during incidents.

Conclusion

The growing complexity and dynamism of cloud-native environments demand a reimagining of digital forensic and incident response practices. Traditional manual approaches are no longer sufficient to meet the speed and scale of modern threats. This paper has proposed a serverless, modular framework that integrates AWS Lambda for EBS snapshots, Velociraptor for live memory acquisition, and Timesketch for timeline analysis—enhanced by automated YARA scanning and threat intelligence enrichment through the VirusTotal API.

This approach reduces response time, increases evidence reliability, and ensures timely threat detection by leveraging cloud-native automation and open-source forensic tooling. Each workflow component—snapshot acquisition, memory analysis, timeline correlation, and IOC matching—is orchestrated through lightweight, event-driven infrastructure, enabling rapid deployment and minimal analyst overhead.

The proposed solution aligns with best practices in cloud forensics and addresses key

limitations in scalability, automation, and integration found in existing workflows. Future work may include evaluating the performance of this architecture in live incident simulations, extending support for cross-cloud environments, and integrating machine learning techniques for anomaly detection within timeline data.

As organizations continue to migrate to elastic and ephemeral infrastructure models, adopting automated, cloud-native incident response frameworks will be essential to maintaining security resilience and forensic readiness.

Reference

[1] P. Castro, V. Ishakian, V. Muthusamy, and A. Slominski, "The rise of serverless computing," Communications of the ACM, vol. 62, no. 12, pp. 44–54, Nov. 2019, doi: https://doi.org/10.1145/3368454

[2] "AWS Security Incident Response Guide," 2019. Available: https://d1.awsstatic.com/whitepapers/aws_security_incident_response.pdf

[3] "Velociraptor Digging Deeper," 2020. Available: https://dfrws.org/wp-content/uploads/2021/03/DFRWS-EU-2021-Velociraptor-Digging-Deeper.pdf?

[4] H. Studiawan, F. Sohel, and C. Payne, "Sentiment Analysis in a Forensic Timeline With Deep Learning," IEEE Access, vol. 8, pp. 60664–60675, 2020, doi: https://doi.org/10.1109/access.2020.2983435

[5] E. Neto, "Finding a fileless malware with Yara rules - Euler Neto - Medium," Medium, Dec. 21, 2020. Available: https://enetolabs.medium.com/finding-a-fileless-malware-with-yara-rules-fdb7c39bf914?

[6] J. Berggren, "Timeline analysis from the future - Timesketch - Medium," Medium, Jul. 21, 2017. Available: https://medium.com/timesketch/timeline-analysis-from-the-future-59a7ad7da498?

[7] S. Zawoad and R. Hasan, "I Have the Proof: Providing Proofs of Past Data Possession in Cloud Forensics," arXiv.org, 2012. Available: https://arxiv.org/abs/1211.4328?

[8] Forensic Focus, "Timelines In Digital Forensic Investigation: From Investigation To

Court - Forensic Focus," Forensic Focus, Sep. 10, 2020. Available: https://www.forensicfocus.com/articles/timelines-in-digital-forensic-investigation-from-investigation-to-court/?.

[9] A. Lockett, "Assessing the Effectiveness of YARA Rules for Signature-Based Malware Detection and Classification," arXiv:2111.13910 [cs], Nov. 2021, Available: https://arxiv.org/abs/2111.13910

[10] F. Lardinois, "Google Acquires Online Virus, Malware and URL Scanner VirusTotal," TechCrunch, Sep. 07, 2012. Available: https://techcrunch.com/2012/09/07/google-acquires-online-virus-malware-and-url-scanner-virustotal/

[11] J. Dykstra and A. T. Sherman, "Design and implementation of FROST: Digital forensic tools for the OpenStack cloud computing platform," Digital Investigation, vol. 10, pp. S87–S95, Aug. 2013, doi: https://doi.org/10.1016/j.diin.2013.06.010. Available: https://www.dfrws.org/sites/default/files/session-files/paper-design_and_implementation_of_frost_-_digital_forensic_tools_for_the_openstack_cloud_computing_platform.pdf.

[12] L. memory, "Linux memory acquisition issues," Information Security Stack Exchange, Apr. 09, 2018. Available: https://security.stackexchange.com/questions/183292/linux-memory-acquisition-issues?

[13] E. E. Eiland, "Time Line Analysis in Digital Forensics," 2018. Available: https://citeseerx.ist.psu.edu/document?doi=d0e290455599cc01d282bf1448f40e2132f3103b&repid=rep1&type=pdf

[14] "How to automate forensic disk collection in AWS | Amazon Web Services," Amazon Web Services, Aug. 24, 2021. Available: https://aws.amazon.com/blogs/security/how-to-automate-forensic-disk-collection-in-aws/?