International Journal of Science and Research Archive

eISSN: 2582-8185

Cross Ref DOI: 10.30574/ijsra

Journal homepage: https://ijsra.net/

(REVIEW ARTICLE)

Check for updates

# The cloud forensics frameworks and tools: A brief review

Sheena Mohammed [1,*] and Sridevi Rangu [2]

[1] *IT Department, Chaitanya Bharathi Institute of Technology, Hyderabad, Telangana, India.*
[2] *CSE Department, Jawaharlal Nehru Technological University Hyderabad, Hyderabad, India.*

International Journal of Science and Research Archive, 2023, 08(01), 173–181

Publication history: Received on 01 December 2022; revised on 08 January 2023; accepted on 10 January 2023

Article DOI: https://doi.org/10.30574/ijsra.2023.8.1.0023

## Abstract

A Cloud is a platform that allows for quick application deployment and dynamic scaling. The cloud differs from on-premise software and data storage in terms of cost, security, scalability, recovery, and mobility which makes more businesses are switching from on-premise to cloud solutions every year. Although cloud computing models have several benefits over on-site models, they are nonetheless vulnerable to both internal and external threats. Even malicious operations can be carried out on the cloud with ease because of the flexible environment. Forensic investigations require the extraction of evidence, and analysis of a cloud system after an intrusion or break-in. It enables investigators to find and retrieve data from a variety of sources in the cloud environment. It is very challenging to find proof of a crime since the distributed nature of the cloud prevents evidence from being retained on a specific physical machine and instead disperses the data over various regions. This paper focuses on current forensic investigation tools used in the Cloud environment and highlights the need for the development of efficient cloud forensic tools.

**Keywords:** CSP; DDOS; MITC; FROST; UFED

## 1. Introduction

The term "cloud forensics" describes investigations that are concentrated on crimes that primarily involve the cloud. Cloud forensics combines conventional computer forensics, small-scale digital device forensics, and network forensics [1]. The Cloud forensics process [2] involves the following phases 1. Event identification, 2. Evidence identification, 3. Collection of evidence, 4. Analysis of evidence, 5. Interpret evidence, 6. Present in the court of law as shown in Fig 1.

As Cloud has three deployment models [3] as defined by NIST. In each deployment model SaaS, PaaS, and IaaS the investigator has to follow different data acquisition techniques due to the dependencies on CSPs. This paper focuses on various attacks on Cloud environments, the tools currently in use, and the need for the development of efficient tools in the cloud environment. Section 2 covers various attacks on the cloud. Section 3 focuses on log-based analysis. Section 4 covers the forensic tools currently in use in the cloud environment. Section 5 highlights forensic frameworks and Section 6 briefly discusses machine learning methods used in intrusion detection in the cloud and fog environments.

* Corresponding author: Sheena Mohammed

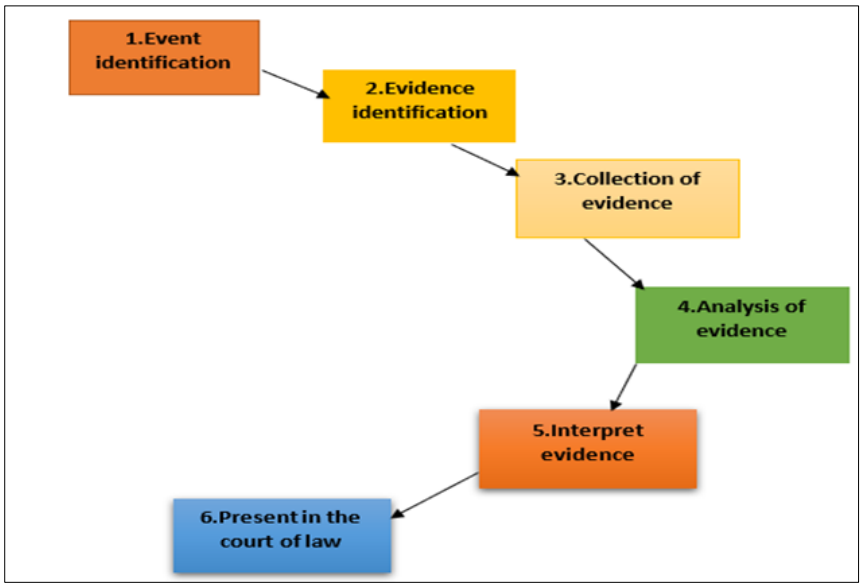


**Figure 1** Cloud Forensic Phases

## 2. Attacks on the cloud

The cloud is a global platform that allows sharing and distributing of digital information quickly and at little cost. Because cloud computing continues to be aggressively developed, there are several loopholes that hackers or nefarious insiders can take advantage of [4]. The cloud environment vulnerabilities are listed below.

- The management interface is being accessed without authorization.
- IP (Internet Protocol) vulnerabilities that enable Man-in-the-Middle attacks.
- Data recovery vulnerability due to reallocation of resources from one user to another.
- Metering and billing data manipulation as well as billing evasion
- Data threats
- Cloud API vulnerabilities impact the administration, provisioning, and monitoring of the cloud's security
- Malicious insiders disclose data in the cloud environment
- Shared technology vulnerabilities can cause significant damage to many cloud users.
- Weak cryptography

Numerous methods exist for attacking cloud computing systems, some are discussed below.

- Malware Injection Attacks:
- DDOS attack
- Abuse of cloud services
- Side-channel attacks
- Man-in-the-cloud attack

### 2.1. Malware Injection Attacks

Instances of virtual machines or explicit service implementation modules need to be managed and coordinated, which is a key duty of a cloud computing system. Any service that a client requests through cloud computing are determined and created by the cloud computing system in an instance of the specified service implementation type. Following that, those incidents are reported and sent to the required client. Some metadata is kept for identification purposes. All web service description documents (WSDL) associated to specific service implementations are covered by this metadata for the PaaS case of web services provided by the cloud. Any user who wants to utilize a cloud system service first checks the service's metadata description to see if it's appropriate for their intended usage.

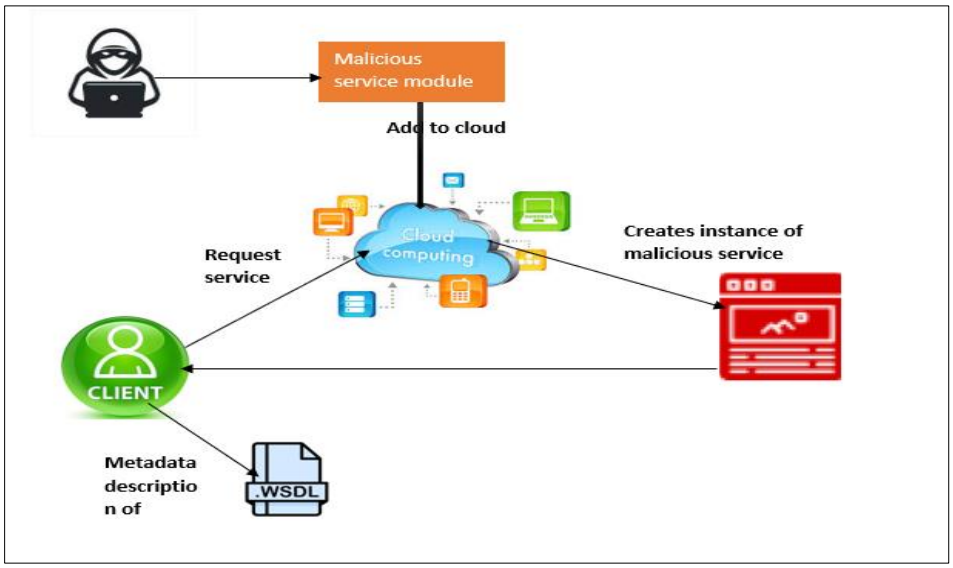


**Figure 2** Malware Injection Attack

A virtual machine running malicious code is introduced into the cloud system via a malware injection attack [5]. In order to carry out this attack, the adversary must develop its own malicious SaaS or PaaS service or IaaS virtual machine instance and then, must deceive the cloud system such that it handles the introduction of new services as one of the acceptable instances for the specific service by the enemy, who was attacked. The attack can be shown as shown in Fig 2 below. If the adversary's code is implemented successfully, legitimate user requests are automatically forwarded to the malicious service implementation by the cloud system.

### 2.2. DDOS attack

A DDoS attack tries to block access to resources or services by flooding system resources with a lot of false traffic [6]. This traffic would heavily levy the target server's resources in a cloud environment as shown in Fig 3. This overload situation is taken as feedback input; a cloud auto-scaling function would increase the pool of resources available for this VM.

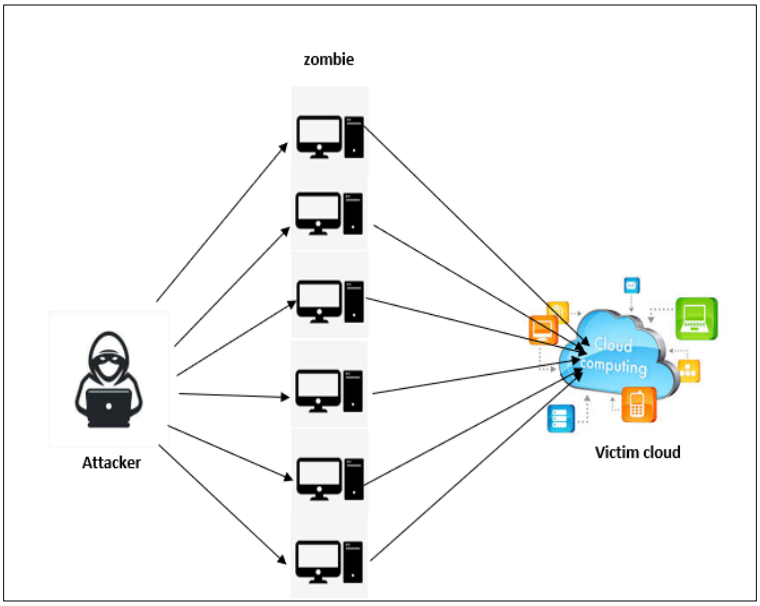


**Figure 3** DDOS Attack on Cloud

### 2.3. Abuse of cloud services

As shown in Fig 4, diverse abuses may affect cloud services [7]. If the cloud provider does not provide strict mechanisms to segregate common resources like memory, storage, and reputation of various customers or hosts, hackers may conduct "Host hopping attacks". In a "Malicious Insider attack" users in high-privilege roles, such as system administrators and information security managers, may abuse their privileged access to clients' sensitive data and risk disclosing or selling that information to rivals or other parties of interest by hosting sensitive information from multiple clients on the same physical machine. Without any limitations or constraints on workloads or resource consumption from cloud suppliers, malicious hackers can easily create accounts with cloud providers to consume cloud resources by paying for the usage called "Identity Theft Attacks". Attackers may take advantage of this benefit to compromise and use sensitive consumer information and then sell it for a fee.

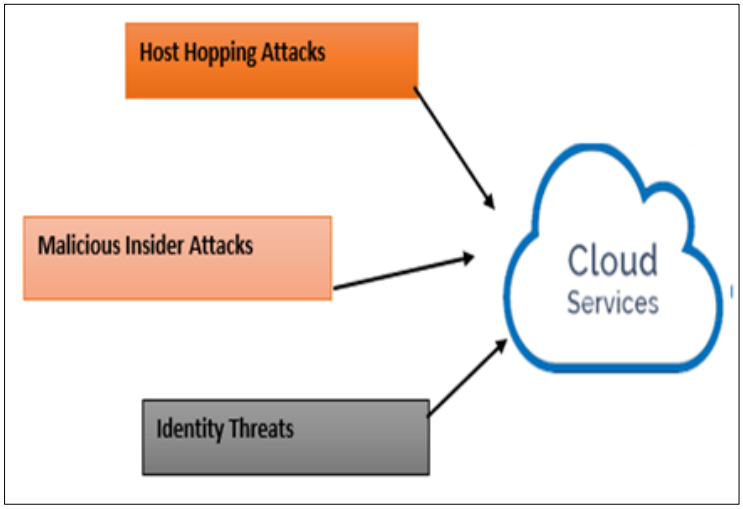


**Figure 4** Abuse of Cloud Services

### 2.4. Side-channel attacks

In contrast to direct attacks that target the programs or their code, side-channel attacks [8] target the system's hardware or indirect effects in order to gain information from or modify the program execution (Fig 5). Placement and Extraction are the two fundamental components of a side-channel attack. Placement refers to the arrangement made by the adversary or attacker to install their malicious VM on the same physical computer as the legitimate one. After the malicious VM has been successfully installed on the target VM, extract the sensitive data, files, and documents from the target VM.

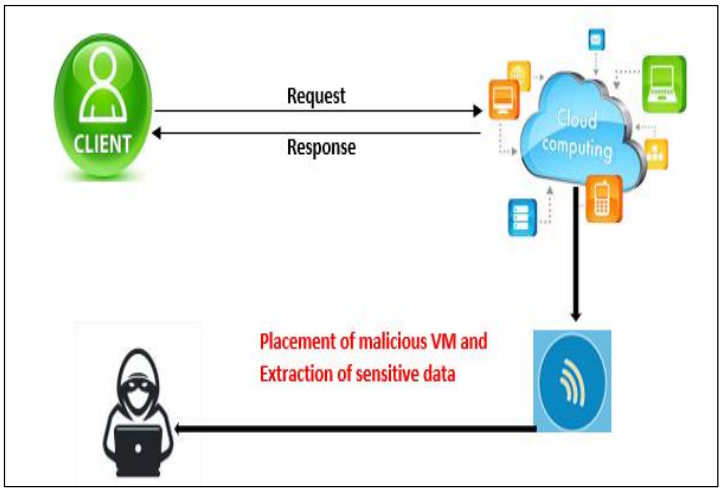


**Figure 5** Side-channel attack

### 2.5. Man-in-the-cloud attack

A type of online eavesdropping is the "Man in the Middle Attack"(MITC) (Fig 6). The goal of hackers is to intercept communications between the source and the destination [9]. The same token might permit access from any device due to the anytime, everywhere feature of cloud services. Consequently, if a hacker has access to and can copy a token, they can remotely access the victim's cloud in a way that seems legitimate and gets over security precautions.

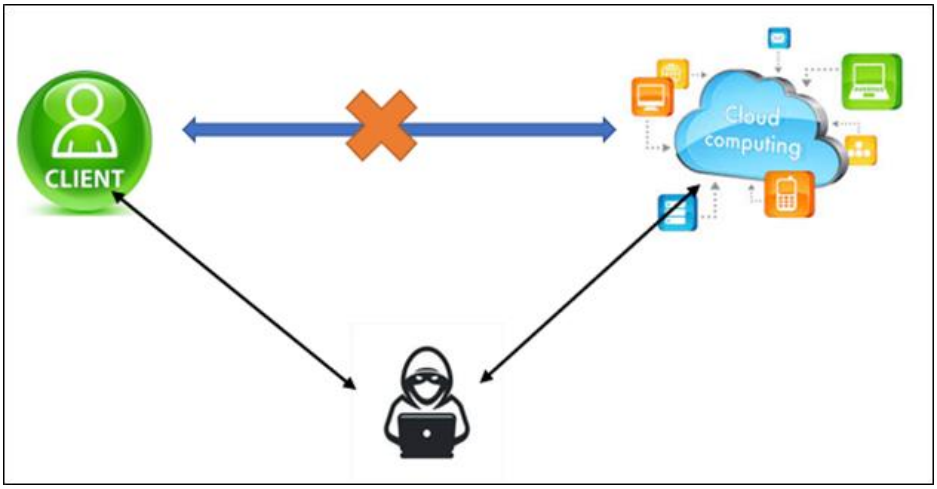


**Figure 6** Man-In-The-Middle Attack

## 3. Log-based investigation

Cloud forensics investigation mainly depends on logs [10]. This session focuses on the types of logs, and formats of logs in a cloud environment.

### 3.1. Types of Logs

#### *3.1.1. Admin Activity*

Log entries for API calls and other operations that change the configuration or metadata of resources can be seen in the audit logs for admin activity. These logs, for instance, keep track of when users create virtual machine instances or modify Identity and Access Management permissions.

#### *3.1.2. Data Access*

Data Access audit logs include API calls that generate, edit or read user-provided resource data and user-driven API calls that read resource configuration or information.

#### *3.1.3. System Event*

Log entries for activities that change resource settings are found in System Event audit logs.

#### *3.1.4. Policy Denied*

A denied audit log is recorded when a Cloud service refuses access to a user or service account due to a security policy breach.

### 3.2. Log formats

The log formats represent the structure of the log file, including the fields and their corresponding data types. These may be structured, unstructured, or semi-structured log formats.[11]. The commonly used log formats are JSON, CFF, Windows Event logs, CLF, ELF, W3C, etc. The three-tier architecture of cloud forensics for Forensics as a Service proposed by Nanda et al [12] is shown in Fig 7.

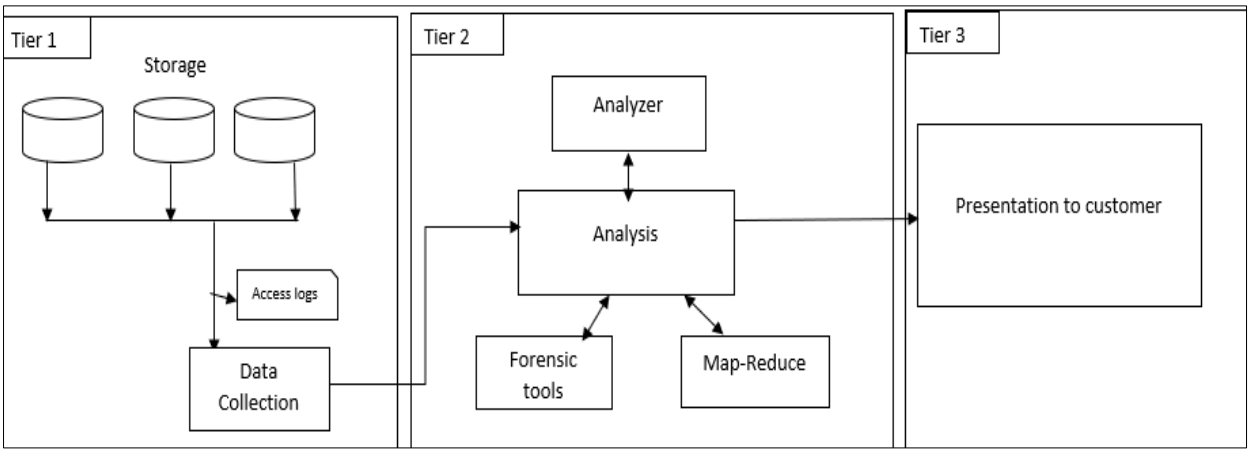


**Figure 7** Forensics as a Service- Three-Tier Architecture

## 4. Forensic tools in the cloud environment

### 4.1. FROST

In the implementation of Forensic Tools for OpenStack cloud computing [13], OpenStack has integrated FROST on Nova, the compute service, and Horizon the web-based user interface. There are three main parts to FROST.

- The virtual discs image linked to any of the user's virtual machines can be retrieved, and to verify the image's integrity cryptographic checksums are applied. To get the disk image for a single virtual disk one can use,

$ Nova get-disk <volume-name> <xmlfile>

The "report.xml" file is created for integrity validations.

- The logs of all API can be retrieved by sending requests to the cloud provider. A user must have successfully signed in to OpenStack using their private key or login information before using FROST. The Nova logs are maintained on the host OS in /var/log/nova/ and the command line to retrieve these logs for a single virtual machine is

$ nova get-nova-logs <UUID> <xmlfile>

It creates "report.xml", a DFXML file, and returns the Nova entries of UUID. DFXML file includes a hash of the log data, for verification of integrity and provenance details about the execution of FROST.

iii. The OpenStack firewall logs for any virtual machines owned by the cloud user can be retrieved, and the logs' accuracy can be checked. To retrieve the firewall logs t from the command line, one can use the following command that returns the UUID's firewall logs and generates a DFXML "report.xml" file for integrity validations.

$ nova get-firewall-logs <UUID> xmlfile>

### 4.2. UFED

The Windows-based extraction and analysis tool UFED Cloud Analyzer [14] can import a file containing login information from various cloud services. The tool is able to collect, store, and analyze data from accounts such as Facebook, Instagram, and Twitter. Additionally, it incorporates techniques for accelerating exploration, such as file storage. The following steps in Table 1 outline how to use the UFED Cloud Analyzer for any forensic investigation:

**Table 1** UFED cloud analyzer for cloud forensics

| SL.no | STEP |
|---|---|
| 1. | A mobile device belonging to the potential criminal is seized. |
| 2. | Decode the login information for cloud services, using the UFED Physical Analyzer. |
| 3. | To harvest private user data, login credentials or information are used. |
| 4. | A uniform format is used for data analysis and reporting. |
| 5. | UFED link Analyzer or any other analysing tools are used to share data and advance the inquiry. |

## 5. Forensic frameworks

Predicting threat behavior in a digital world is challenging since different harmful actions exhibit a variety of different behaviors. Nighat Usman et al. [15] constructed a hybrid prediction system based on dynamic behavior. The decision tree model has been used in this instance to model the dynamic prediction It performed with mediocre forecast accuracy and took longer to execute.

G. Nandita and T. Munesh Chandra developed a deep neural network with the frog leap algorithm to anticipate harmful behaviors [16]. The server host's malicious behavior has been recognized in this instance. The cloud's performance may suffer as a result of malicious server activity. Data theft occurs in cloud storage as a result, thus deep networks that were optimized for the situation were put in place to keep an eye on the server's activities. The best privacy rating was given to it. But the pattern is intricate.

P. Mohamed Shakeel et al [17] have described linear analysis with the blockchain mechanism for foretelling the threat in forensic data. Internet forensic data was used to evaluate the model that was put into use. Here, linear regression was applied to analyze user behavior and forecast anomalous behavior. Blockchain technology has been used to provide privacy. However, more resources were needed to put this strategy into action.

Deevi Radha Rani and G. Geethakumari [18] established the three-layer prediction framework for predicting threat behavior. Further, an intrusion detection database was used for the testing process. As a result, the designed three-stage approach has outperformed other existing models in terms of malicious prediction exactness. However, this three-stage prediction method has reported a high execution time when compared to the traditional incursion forecasting approach.

Table 2 shows the observation on the different frameworks discussed above

**Table 2** Observations on existing frameworks

| Sl. No | Framework | Prediction system | Observation |
|---|---|---|---|
| 1 | Usman et al [15] | hybrid prediction system based on the dynamic behavior | Longer to execute. |
| 2 | Nandita et al [16] | frog leap algorithm | The pattern is complex. |
| 3 | Mohamed Shakeel et al [17] | linear analysis with the blockchain mechanism | More resources |
| 4 | Geethakumari et al [18] | the three-layer prediction framework | High execution time |

## 6. Machine learning based intrusion detection

### 6.1. Autoencoder (AE)

A deep learning-based method (Auto-IF) for intrusion detection that makes use of Autoencoder (AE) and Isolation Forest (IF) for a fog environment [19]. This method focuses on the binary categorization of the incoming packets because real-time attack detection is what fog devices are most concerned with.

### 6.2. Stacked CAE with Support Vector Machine (SCAE-SVM):

Deep and shallow learning approaches are used in the Stacked contractive auto-encoder and support vector machine strategy [20], which fully utilizes their benefits to minimize the analytical overhead.

### 6.3. CNN with LSTM

Convolutional neural networks and long short-term memories are combined in this technique. As a result, all attacks detected by the cloud's network layer are efficiently categorized.[21].

These approaches used the standard NSL-KDD dataset to analyze the performance. The methods are compared on the various parameters including Accuracy, Precision, Recall, and f-Score. Table 3 shows the comparative analysis on these existing frameworks.

A cloud forensic tool is the most important component for a cyber application to forecast threat activities in a given data set. To address these difficulties, there are currently a number of intelligent networks. Threat prediction is a big challenging issue due to the distinctive behavior of the various hazardous occurrences. Various frameworks are proposed but still, there is a need for efficient tools which can predict the attacks on the cloud and classify them accordingly.

**Table 3** Comparative analysis of existing methods

| Sl. No | Method | Accuracy (%) | Precision (%) | Recall (%) | F-score (%) |
|---|---|---|---|---|---|
| 1 | AE-IF | 94.81 | 97.25 | 96.01 | 95.4 |
| 2 | SCAE-SVM | - | 91 | 90 | 87 |
| 3 | CNN-LSTM | 63.01 | 63 | - | 77.19 |

## 7. Conclusion

This paper has focused on the forensic investigation process in a cloud environment for intrusion detection, and common possible attacks on the cloud environment. This paper also gives a brief review of existing forensic tools and frameworks in the cloud environment and highlights the need for efficient tools in this field.

## Compliance with ethical standards

*Acknowledgments*

This paper and the research behind it would not have been possible without the exceptional support of my supervisor Dr. Sridevi Rangu, Professor, JNTUH. I want to express my gratitude to the management of Chaitanya Bharathi Institute of Technology who has been supportive of my career goals and who worked actively to provide me with the protected academic time to pursue those goals. I would like to thank my parents, whose love and guidance are with me always. I wish to thank my loving and supportive husband, Mohammad Riyaz Pasha, and my only loving kid, Minha, who provides unending inspiration.

*Disclosure of conflict of interest*

Sheena Mohammed Declares no conflicts of Interest. She is a Research Scholar at JNTUH under the supervision of Dr. Sridevi Rangu. This paper is a part of Sheena Mohammed's literature survey. The article is not under consideration for publication elsewhere. On behalf of the Co-Author, the corresponding author bears full responsibility for submission.

## References

[1] Keyun Ruan, Joe Carthy, Tahar Kechadi, Ibrahim Baggili, Cloud forensics definitions and critical criteria for cloud forensic capability: An overview of survey results, 2013 Elsevier Ltd, http://dx.doi.org/10.1016/j.diin.2013.02.004

[2] Purnaye, P., Kulkarni, V. A Comprehensive Study of Cloud Forensics. Arch Comp utat Methods Eng 29, 33–46 (2022). https://doi.org/10.1007/s11831-021-09575-w

[3] P. Mell and T. Grance. “NIST definition of cloud computing”. National Institute of Standards and Technology. October 7, 2009

[4] B. Grobauer, T. Walloschek and E. Stocker, "Understanding Cloud Computing Vulnerabilities," in IEEE Security & Privacy, vol. 9, no. 2, pp. 50-57, March-April 2011, DOI: 10.1109/MSP.2010.115.

[5] Mrs. Asma A. Shaikh, “Attacks on Cloud Computing and its Countermeasures” International conference on Signal Processing, Communication, Power and Embedded System (SCOPES)-2016, 978-1-5090-4620-1/16/$31.00 ©2016 IEEE.

[6] Gaurav Somani , Manoj Singh Gaur , Dheeraj Sanghi , Mauro Conti, Rajkumar Buyya, “DDoS attacks in cloud computing: Issues, taxonomy, and future directions”, Computer Communications Volume 107, 15 July 2017, Pages 30-48, http://dx.doi.org/10.1016/j.comcom.2017.03.010 0140-3664/© 2017 Elsevier B.V.

[7] Yasir Ahmed Hamza, Marwan Dahar Omar “Cloud Computing Security: Abuse and Nefarious Use of Cloud Computing”, International Journal of Computational Engineering Research, Vol 03 Issue 6, June 2013

[8] Bhrugu Sevak , “Security against Side Channel Attack in Cloud Computing” International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-2, December 2012

[9] Zimba, Aaron & Wang, Zhaoshun. (2017). On Man-In-The-Cloud (MITC) attacks: The analytical case of Linux. 170-172. 10.1109/ISI.2017.8004901.

[10] Ghosh, A., De, D., Majumder, K. (2021). A Systematic Review of Log-Based Cloud Forensics. In: Smys, S., Balas, V.E., Kamel, K.A., Lafata, P. (eds) Inventive Computation and Information Technologies. Lecture Notes in Networks and Systems, vol 173. Springer, Singapore

[11] Arfan Sharif, 6 Common Log File Formats- December 21, 2022. https://www.crowdstrike.com/cybersecurity-101/observability/log-file-formats/

[12] Saurav Nanda, Raymond A Hansen, and Forensics as a Service: Three-tier Architecture for Cloud based Forensic Analysis, 2016 15th International Symposium on Parallel and Distributed Computing.

[13] Josiah Dykstra Alan T.Sherman, “Design and implementation of FROST: Digital forensic tools for the OpenStack cloud computing platform” Volume 10, Supplement, 2013, Pages S87-S95, ISSN 1742-2876,http://dx.doi.org/10.1016/j.diin.2013.06.010, Elsevier.

[14] Cellebrite, “Extracting Legally Defensible Evidence From The Cloud,” Explaining UFED Cloud Analyzer Extraction and Analysis Processes, 2015.

[15] Usman, Nighat, et al. "Intelligent dynamic malware detection using machine learning in IP reputation for forensics data analytics." Future Generation Computer Systems 118 (2021): 124-141

[16] Nandita, G., and T. Munesh Chandra. "Malicious host detection and classification in cloud forensics with DNN and SFLO approaches." International Journal of System Assurance Engineering and Management (2021): 1-13

[17] Shakeel, P. Mohamed, et al. "Internet of things forensic data analysis using machine learning to identify roots of data scavenging." Future Generation Computer Systems 115 (2021): 756-768.

[18] Rani, Deevi Radha, and G. Geethakumari. "A framework for the identification of suspicious packets to detect anti-forensic attacks in the cloud environment." Peer-to-Peer Networking and Applications 14.4 (2021): 2385-2398.

[19] Sadaf, Kishwar, and Jabeen Sultana. "Intrusion detection based on autoencoder and isolation forest in fog computing." IEEE Access 8 (2020): 167059-167068.

[20] Wang, Wenjuan, et al. "Cloud intrusion detection method based on stacked contractive auto-encoder and support vector machine." IEEE transactions on cloud computing “(2020).

[21] Thilagam, T., and R. Aruna. "Intrusion detection for network based cloud computing by custom RC-NN and optimization." ICT Express 7.4 (2021): 512-520