IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15338

REAL-TIME LOG ANALYSIS WITH AWS OPENSEARCH AND DOCKER

Abinaya G[1], Mr. T. Pradeep[2]

III BCA, Department of Computer Applications, Sri Ramakrishna College of Arts and Science (Autonomous), Coimbatore[1]

Assistant Professor, Department of Computer Applications, Sri Ramakrishna College of Arts and Science (Autonomous), Coimbatore[2]

Abstract**:** Cloud computing environments generate a massive amount of operational and security logs due to continuous API calls, user authentication activities, and service interactions. Monitoring these logs manually is inefficient and cannot provide real-time insights for security and operational management. This research presents a centralized real-time log monitoring system using AWS CloudTrail, Docker, and Amazon OpenSearch. The proposed system automatically collects CloudTrail logs stored in Amazon S3, processes them through a Dockerbased log ingestion engine, and indexes them in Amazon OpenSearch for fast search and analysis.

OpenSearch Dashboards provide visualization tools that help administrators detect suspicious activities, analyze system usage patterns, and improve cloud security monitoring. The system ensures automation, scalability, centralized visibility, and efficient log analytics in AWS environments. By integrating AWS services with containerization technologies, the system simplifies log monitoring and enables faster incident detection. The solution provides an effective framework for real-time monitoring of cloud infrastructure activities.

Keywords: CloudTrail, OpenSearch, Docker, Log Analysis, Cloud Security, AWS

I. INTRODUCTION

Cloud computing has transformed the way organizations deploy, manage, and scale applications. Instead of relying on traditional on-premises infrastructure, organizations now use cloud platforms that provide flexible computing resources, high availability, and scalable storage systems. Among the leading cloud providers, Amazon Web Services (AWS) is widely used by businesses, startups, and government organizations.

As cloud adoption continues to increase, managing and monitoring system activities becomes critical for maintaining security and operational efficiency. Every action performed in an AWS environment generates a log entry. These logs record various system activities such as user logins, API calls, service interactions, configuration changes, and resource creation or deletion.

Logs play a crucial role in cloud security and monitoring. They help administrators identify unauthorized access attempts, detect unusual system behavior, investigate incidents, and ensure compliance with security policies. However, as cloud infrastructure grows, the volume of logs generated also increases significantly.

In large-scale cloud environments, thousands of API calls and user actions occur every minute. Manually analyzing such large volumes of logs is inefficient and time-consuming. Traditional log monitoring methods require administrators to download raw log files, inspect them manually, and search for specific events. This process is slow and does not support real-time monitoring.

AWS provides a service called CloudTrail, which records account activities and API calls within an AWS account. CloudTrail logs are stored in Amazon S3 in JSON format. Although CloudTrail ensures that all activities are recorded, the stored logs are not easily searchable without additional processing.

To address these limitations, automated log analytics systems are required. Real-time log analysis systems collect logs continuously, process them, and store them in searchable databases where administrators can perform queries and visualize system activities.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15338

This project proposes a Real-Time Log Analysis System using AWS OpenSearch and Docker. The system integrates CloudTrail, Amazon S3, Docker-based log processing tools, and Amazon OpenSearch to create a centralized and automated log monitoring pipeline.

II. LITERATURE REVIEW

Log management systems have become essential components of modern cloud infrastructures. Several studies highlight the importance of centralized logging systems in maintaining security and operational visibility. Traditional logging systems mainly focus on storing log files for auditing purposes, but they often lack efficient search and analysis capabilities.

Centralized log management platforms such as the ELK Stack (Elasticsearch, Logstash, and Kibana) have been widely used to collect and analyze logs from distributed systems. These systems provide indexing and visualization capabilities that enable administrators to search and analyze logs efficiently.

However, deploying and managing self-hosted log analytics platforms requires significant infrastructure management. Cloud-based logging solutions have emerged as an alternative approach to simplify log management.

AWS OpenSearch Service provides a managed search and analytics platform based on Elasticsearch technology. It enables users to index, search, and analyze large datasets with high performance and scalability. The integration of OpenSearch with other AWS services simplifies log analytics workflows.

Recent research also emphasizes the use of containerization technologies such as Docker to deploy log processing tools. Docker containers provide isolated environments that simplify application deployment and ensure consistent configurations across different systems.

The integration of cloud-native logging services with containerized log processors and search engines provides an efficient architecture for real-time log monitoring. This approach improves security monitoring, reduces operational complexity, and enables faster incident detection.

III. SYSTEM ARCHITECTURE

The proposed system architecture is designed to automate the collection, processing, and analysis of AWS logs. The architecture consists of several interconnected components that work together to process and analyze log data.

Main Components

- AWS CloudTrail
- Amazon S3
- Docker Container (Fluentd Log Processor)
- Amazon OpenSearch Service
- OpenSearch Dashboards

Architecture Workflow

1. AWS CloudTrail continuously captures account activities and API calls.
2. CloudTrail delivers log files to an Amazon S3 bucket.
3. A Docker container running Fluentd retrieves log files from S3.
4. The logs are parsed and transformed into structured records.
5. Processed logs are forwarded to Amazon OpenSearch.
6. OpenSearch indexes the logs for fast search operations.
7. OpenSearch Dashboards provide visualization and monitoring tools.

This architecture ensures automated log collection and centralized monitoring of AWS infrastructure activities.

IV. SYSTEM DESIGN

The system design focuses on building a scalable pipeline that processes log data efficiently. The design follows a layered architecture where each component performs a specific role in the log analytics workflow.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15338

Log Generation Layer
AWS CloudTrail acts as the primary log generation service. It records management events and API calls performed within the AWS account. These events include user authentication, resource provisioning, service interactions, and configuration changes.

Storage Layer
CloudTrail logs are stored in Amazon S3. The logs are stored in compressed JSON format to reduce storage consumption. Amazon S3 provides high durability and ensures that log data is securely stored.

Processing Layer
A Docker container running Fluentd reads log files from the S3 bucket. Fluentd is responsible for parsing log files, extracting important fields, and converting the data into structured records.

Indexing Layer
Processed log records are sent to Amazon OpenSearch Service. OpenSearch indexes the log data and enables fast search operations across large datasets.

Visualization Layer
OpenSearch Dashboards provide visualization tools that allow administrators to monitor log data through charts, graphs, and tables.

V. IMPLEMENTATION

The system implementation involves several steps for configuring AWS services and deploying the log processing environment.

Step 1: Configure AWS CloudTrail
CloudTrail is enabled to record management events across the AWS account. The trail is configured to deliver log files to a designated S3 bucket.

Step 2: Create Amazon S3 Bucket
A dedicated S3 bucket is created to store CloudTrail log files. The bucket stores logs in a structured folder hierarchy based on account ID, region, and date.

Step 3: Launch EC2 Instance
An Amazon EC2 instance is launched to host the Docker environment. The EC2 instance acts as the log processing server.

Step 4: Install Docker
Docker is installed on the EC2 instance to run containerized log processing tools.

Step 5: Deploy Fluentd Container
A Fluentd Docker container is deployed to read CloudTrail logs from the S3 bucket. Fluentd parses the logs and extracts important fields such as event name, user identity, and timestamp.

Step 6: Configure OpenSearch
An OpenSearch domain is created to store indexed log data.

Step 7: Connect Fluentd to OpenSearch
Fluentd is configured to send processed logs to the OpenSearch domain using secure HTTPS communication.

Step 8: Create Dashboards
OpenSearch Dashboards are configured to visualize log data and display activity trends.

VI. RESULTS AND ANALYSIS

The implemented system successfully collects AWS CloudTrail logs and processes them in near real time. The logs are indexed in OpenSearch and displayed through OpenSearch Dashboards.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15338

The dashboard visualizations include:

- Login activity trends
- API usage statistics
- Service activity distribution
- Error event monitoring
- Failed authentication attempts

Administrators can filter logs using multiple parameters such as:

- Username
- IP address
- Event type
- AWS region • Time range

This improves log investigation efficiency and enables faster detection of suspicious activities.

VII. ADVANTAGES OF THE SYSTEM

The proposed system offers several advantages compared to traditional log monitoring methods.

- Real-time log monitoring
- Automated log collection and processing
- Centralized log management platform
- Fast search and filtering capabilities
- Interactive dashboards and visualizations
- Improved security monitoring
- Scalable architecture for large cloud environments

VIII. CONCLUSION

Cloud infrastructure environments generate large volumes of logs that must be monitored for security and operational management. Traditional log analysis methods are inefficient and cannot handle the scale of modern cloud systems.

This project demonstrates an automated real-time log analysis system using AWS CloudTrail, Docker, and Amazon OpenSearch. The system collects logs automatically, processes them through containerized log processors, and indexes them in OpenSearch for fast searching and visualization.

The proposed system provides centralized monitoring, faster incident detection, and improved operational visibility. By integrating cloud-native services with containerization technology, the solution enhances cloud security and simplifies log management.

IX. FUTURE ENHANCEMENTS

Future improvements to the system may include:

•Automated alert notifications using AWS SNS

- Machine learning-based anomaly detection
- Multi-account log aggregation
- Integration with SIEM security platforms
- Real-time automated security incident response

These enhancements will further strengthen the monitoring capabilities of the system.

REFERENCES

[1]. Amazon Web Services Documentation – AWS CloudTrail
[2]. AWS OpenSearch Service Documentation
[3]. Docker Containerization Documentation
[4]. Fluentd Logging Framework Documentation
[5]. Distributed Logging and Monitoring Systems Research Papers

