Enhancing Cloud Audit based on Log Normalization

Yeeun Kim, Hyeon No, Min jeong Kim, So hui Kim, Wonhyung Park,
Il-Gu Lee, and Seongmin Kim*

Sungshin Women's University, Seoul, Republic of Korea
{220224005, 220224007, 220226033, 220226034, whpark, iglee, sm.kim}@sungshin.ac.kr

Keyword: Cloud Log, AWS, Azure, Normalization

1 Introduction and Background

The diversity and volume of logs generated in edge computing accentuate the significance of data collection and analysis. However, distributed log data generated in heterogeneous environments, such as Amazon Web Services (AWS) and Microsoft Azure (Azure) have different formats and fields. Moreover, the names of fields that mean the same value are also different, making integrated analysis between the two environments difficult. In particular, in edge computing, the frequency of log data generation increases significantly due to the frequent need for real-time data processing. This aspect highlights the necessity for an efficient framework to store and analyze logs effectively [1]. To achieve this, cloud service providers (CSPs) offer own risk auditing solutions to manage log data generated by instances and storage. AWS CloudTrail and Azure Activity Logs are representative examples. **CloudTrail** is a logging service that records all API calls and activities related to the overall infrastructure performed within the AWS account. It enables extracting records of significant events, such as changes in resource status, instance start or stop, and Identity and Access Management (IAM) user activities [2]. **Azure Activity Log** is a service that records and provides all management events and data events that occur in Microsoft Azure environments. It contains logs about Azure's various activities and events, as well as logs related to edge computing [3]. **ELK stack** is an open-source project composed of Elasticsearch, Logstash, and Kibana, serving as tools for collecting, storing, and analyzing data from various sources. For example, it is possible to perform integrated analysis of events and log data occurring in various cloud environments (e.g., CloudTrail). The integrated analysis results from various cloud platforms are typically presented in formats that are unique to each specific cloud environment, rather than adhering to a normalized log format. Consequently, even when setting up a unified log collection system, effectively analyzing these logs can be challenging due to the variations in log types and formats generated by each cloud platform. To address this, our study presents an efficient log analysis framework that is based on the normalization of log fields. As a demonstrating example, our analysis focuses on two state-of-the-art cloud platforms, AWS and Azure, both of which have a substantial market presence.

2 Cloud Log Normalization Framework

After identifying the limitations of integrated analysis tools with platform-specific log formats, we propose a field normalization-based log analysis framework for efficient log analysis. Figure 1 represents an overview and data flow of the proposed framework.

The 7th International Conference on Mobile Internet Security (MobiSec'23), Dec. 19-21, 2023, Okinawa, JAPAN, Article No.P-29
*Corresponding author: Sungshin Women's University, Seoul, 02844, Republic of Korea, Email: sm.kim@sungshin.ac.kr

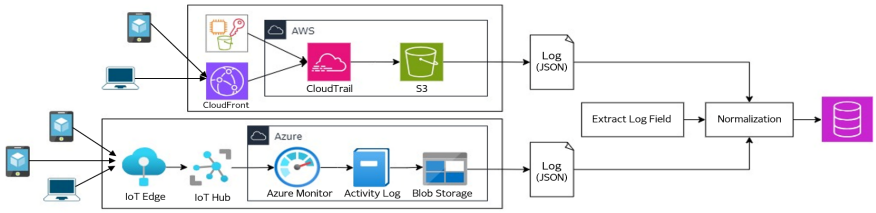

Figure 1: Framework Overview

To extract and consolidate log data from AWS CloudTrail and Azure Activity Log, our initial step involves discerning the meaning of log fields in each environment. Given that the significance of fields may vary depending on the type of incident, it is crucial to analyze them selectively in accordance with the intended use. Subsequently, the normalization of log field names across cloud platforms is imperative. Even when these fields convey the same information, different names are employed in each environment. Table 1 is a partial representation of a table matching log fields between AWS and Azure platforms. After achieving field normalization, we construct a database and retrieve log data from central storage (e.g., S3 bucket and Azure Blob Storage). This setup facilitates log analysis through queries based on normalized fields and enhances storage efficiency since only essential fields are extracted and loaded into the database.

Table 1: AWS-Azure log Fields Normalization Example

AWS	Azure	Description
eventTime	eventTimestamp	Time of Event or Operation Occurrence
eventName	operationName	Requested operation Name

3 Conclusion and Future Work

In this study, we propose a field-normalized log analysis framework to enhance the efficiency of the log analysis process in cloud environments. Furthermore, by adding the correlation table between EventName and MITRE ATT&CK Tactics, we identified and analyzed EventName related to the Credential Access tactic. Based on our analysis, the main actions of the tactic were correctly mapped to EventNames. This framework, utilizing an analysis approach where key fields are extracted, was able to enhance the storage efficiency of the data, enabling quick access and response to the data. In future research, we intend to investigate the integration rate of log fields and determine whether the efficiency of integrated log analysis can contribute to unified management. Additionally, we plan to quantitatively assess the improvement in storage efficiency resulting from the normalization process.

Acknowledgement: This work is partly supported by the NRF grant funded by the Korean government (No. 2021R1G1A1006326), the Korea Institute for Advancement of Technology (KIAT) grant funded by the Korean Government (MOTIE) (P0008703, The Competency Development Program for Industry Specialist), and the MSIT under the ICAN (ICT Challenge and Advanced Network of HRD) program (No. IITP-2022-RS-2022-00156310) supervised by the Institute of Information Communication Technology Planning Evaluation (IITP).

References

[1] Shilin He, Qingwei Lin, Jian-Guang Lou, Hongyu Zhang, Michael R. Lyu, and Dongmei Zhang. Identifying impactful service system problems via log analysis. In *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE 2018, page 60–70, New York, NY, USA, 2018. Association for Computing Machinery.

[2] Simeen Sheikh, G. Suganya, and M. Premalatha. *Automated Resource Management on AWS Cloud Platform*, pages 133–147. Springer Singapore, Singapore, 2020.

[3] Timothy L. Warner. *Monitoring Your Azure Environment*, pages 301–322. 2020.