CERIAS

The Center for Education and Research in Information Assurance and Security

Unleashing Insights from Terabytes: Microservices Architecture for Digital Intelligence and Evidence

Akif Ozer and Umit Karabiyik

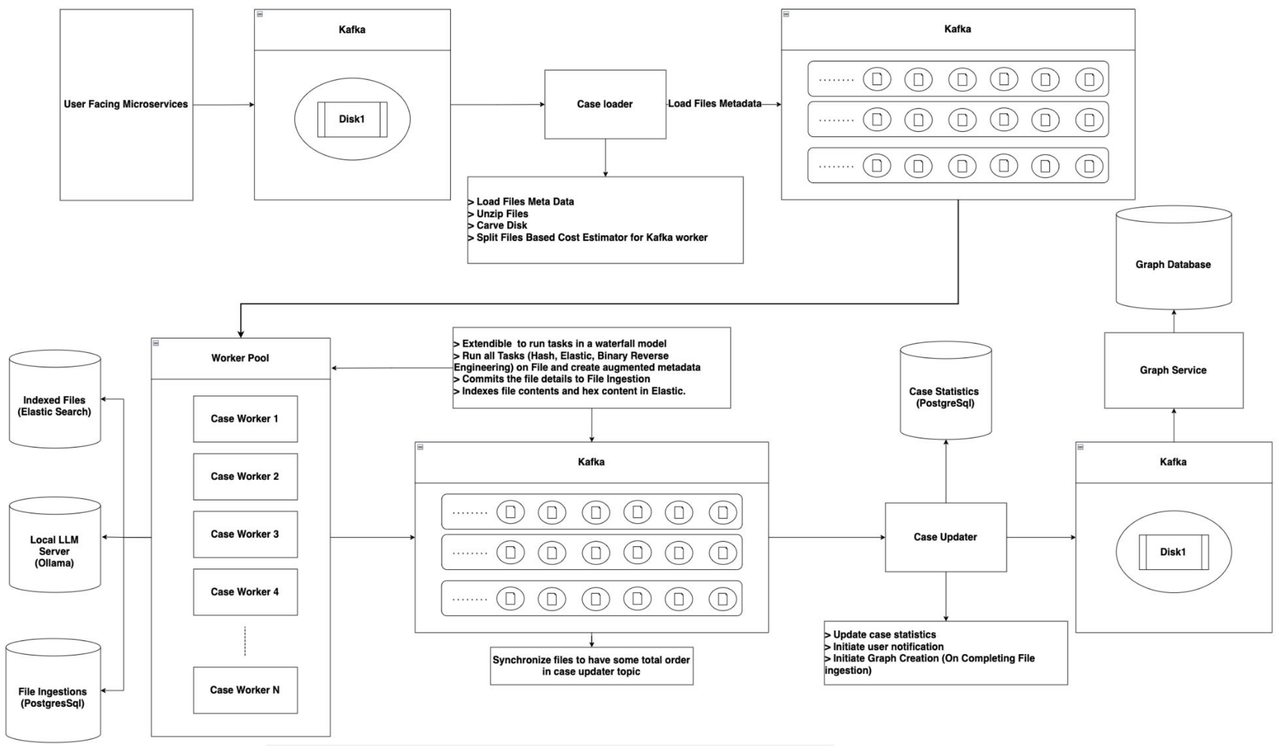
OVERVIEW

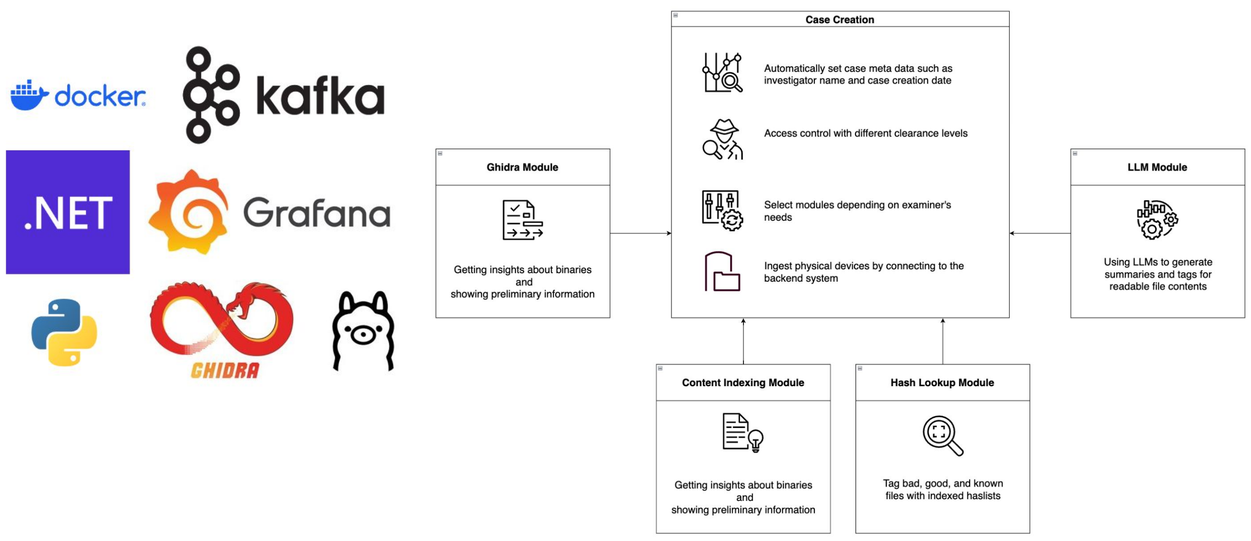
FOREST is a distributed forensic search tool built on an event-driven microservices architecture. It uses Apache Kafka for parallel processing and relies on ElasticSearch and PostgreSQL for fast data handling. Integrated AI modules and Ghidra automate artifact extraction, while a graph service links data across cases. This design offers a scalable and efficient solution for modern forensic investigations.

This research project was funded by AFLR and NASIC over 4 years and 5 phases

METHODOLOGY

FOREST uses an event-driven microservices architecture where each component is responsible for a distinct stage of forensic data processing. User-facing services send tasks to **Kafka**, which coordinates file ingestion through a **Case Loader** that unzips files, extracts metadata, and carves disk images. The **Worker Pool** then processes files, performing tasks like hashing, reverse engineering (via Ghidra), and AI-based summarization and tag generation (using a local LLM). Processed data is stored in **ElasticSearch** for rapid querying and in **PostgreSQL** for structured metadata. The **Case Updater** updates case statistics, and initiates graph creation in a **Graph Database**, allowing investigators to uncover cross-case correlations. This design promotes scalability, reliability, and efficiency in handling large volumes of forensic data.





GOALS

The main goals of our project included:

- Examining the potential of a distributed microservices architecture to enhance the processing and analysis of large-scale forensic data.
- Developing a tool that integrates AI-driven artifact extraction, graph-based indexing, and reverse engineering to improve the speed and accuracy of digital forensic investigations.

TOOL EVALUATION

Our performance benchmarks indicate that FOREST is highly effective at processing large forensic datasets. The table below summarizes the processing times for different dataset sizes:

Test Size	File Count	Time to Complete
500 GB	~485,000 files	8 hours
1.5 TB	~1,000,000 files	23 hours
4.5 TB	~2,000,000 files	41 hours

Additionally, scalability tests demonstrate that doubling the number of worker nodes significantly reduces processing time by half. For a dataset of 10,000 files, the results are as follows:

Test Size	File Count	Time to Complete
10K Files	1	**27 minutes and 37.97 seconds**
10K Files	2	**14 minutes and 0.21 seconds**
10K Files	4	**7 minutes and 25.05 seconds**
10K Files	8	**3 minutes and 56.17 seconds**

These evaluations show that FOREST's distributed microservices architecture efficiently handles large volumes of forensic data. Importantly, the system is designed to manage a high number of files, meaning that processing time is more directly influenced by the file count rather than the overall dataset size. As additional worker nodes are deployed, processing times decrease significantly, ensuring rapid and reliable digital forensic investigations even when faced with millions of individual files.

PURDUE UNIVERSITY

{ozer, umit}@purdue.edu