ABSTRACT

The rapid evolution of cloud-native architectures has led to the widespread adoption of microservices and serverless computing, significantly enhancing system scalability, modularity, and deployment agility. However, these distributed, sandboxed, and ephemeral environments introduce severe challenges in system observability, forensic investigation, and security monitoring. Traditional monitoring and audit mechanisms that have been designed primarily for monolithic applications are rendered inadequate in these contexts due to fragmented visibility, namespace isolation, asynchronous communication, and the absence of unified log semantics. As a result, detecting advanced persistent threats (APTs), tracing causality across services, or conducting root cause analysis becomes significantly hindered in practice.

This thesis addresses the core research problem of enabling scalable, non-intrusive, and causally-consistent observability in distributed microservice ecosystems. It proposes a comprehensive approach to bridge the semantic and operational gap between system-level telemetry and application-level behavior by constructing unified provenance graphs that encode fine-grained, temporally and causally ordered interactions across heterogeneous components. These provenance graphs serve as the foundational structure for understanding system dynamics, detecting anomalies, and investigating multi-stage attacks.

The first contribution of the thesis develops methods to statically extract control flow representations from microservice binaries and dynamically associate them with system call logs during runtime, thereby capturing semantically meaningful relationships between logs originating from different abstraction layers. This unified view enables accurate reconstruction of execution paths across serverless functions and facilitates effective correlation during attack analysis. To overcome the limita-

tions of intrusive logging mechanisms and version-dependent instrumentation, the thesis further introduces a runtime log collection and synchronization methodology that leverages modern in-kernel tracing capabilities. It achieves low-overhead, platform-agnostic logging while preserving causal consistency of log entries across distributed hosts using kernel-level vector clocks and synchronized buffer mechanisms.

Recognizing that real-world attack detection must operate under conditions of incomplete labeling and evolving threat behavior, the next contribution of the thesis advances the provenance analysis by enriching the provenance graph with semantic attributes, such as system call arguments, file/socket metadata, and process namespace features. It then formulates the anomaly detection problem as an unsupervised graph reconstruction task, wherein the model learns to represent normal system behavior through latent embeddings of heterogeneous entities and their interactions. Deviations from these representations are then interpreted as potential indicators of abnormal or malicious behavior. The approach is inherently robust to unseen or zero-day attacks and does not rely on prior knowledge of vulnerabilities or labeling of events.

The proposed methodology is rigorously evaluated through a series of real-world deployments on large-scale microservice benchmarks and controlled attack simulations using known CVEs. The results demonstrate marked improvements in attack detection accuracy, precision, and scalability over state-of-the-art baselines. Additionally, the thesis quantifies improvements in log fidelity, reduction in irrelevant log noise, and end-to-end detection latency, while maintaining a minimal resource footprint suitable for production-grade deployments.

In summary, this research contributes a principled and practical solution to the problem of secure observability and attack forensics in cloud-native environ-

ments. By integrating causality-aware logging, provenance modeling, and graph-based anomaly detection into a cohesive pipeline, it lays the groundwork for a new generation of security analytics systems tailored for the complexity and dynamism of distributed microservice architectures.

Keywords: System Observability, Runtime Provenance, Causality Tracking, eBPF-based Monitoring, Provenance Graph, Attack Detection, Dynamic Log Analysis, Log Correlation, Distributed Microservices