Simulating Microservice-based Architectures for Resilience Assessment Enriched by Authentic Container Orchestration

Sebastian Frank¹, Martin Straesser², Lion Wagner³, Patrick Haas², Alireza Hakamian¹, Samuel Kounev², André van Hoorn¹

Abstract: This summary outlines the resilience simulation approach MiSim [Fr22b] published at the International Conference on Software Quality, Reliability and Security (QRS 2022) and its extension for authentic container orchestration [St23] presented at the International Conference on Performance Evaluation Methodologies and Tools (VALUETOOLS 2023).

Keywords: Microservices; Resilience; Scenarios; Simulation; Kubernetes; Container Orchestration

Context. Increased resilience compared to monolithic architectures is both one of the key promises of microservice-based architectures and a big challenge, e.g., due to the systems’ distributed nature. Resilience assessment through simulation requires fewer resources than measurement-based techniques in production environments, as used in practice.

Objective. However, there is no existing simulation approach suitable for a holistic resilience assessment of microservices comprised of (i) representative fault injections, (ii) common resilience mechanisms, and (iii) time-varying workloads. The MiSim approach aims to overcome these limitations and fit resilience engineering practices by supporting scenario-based experiments. Also, MiSim focuses on low modeling effort by using lightweight input models and high flexibility by being extensible regarding resilience mechanisms.

Method. Figure 1 shows the workflow, inputs, and outputs of MiSim. Resilience engineers must provide experiments in the generic description format or, more intuitively, as scenarios. The experiment descriptions contain information about the workload and faultload. MiSim also requires a lightweight architecture description of the system under test, which contains information about the services, interdependencies, and the implemented resilience mechanisms. By default, MiSim can perform event-driven simulations of the common resilience mechanisms (i) circuit breaker, (ii) retry, (iii) autoscaling, (iv) load balancing, and (v) connection limiter. New mechanisms can be added by providing the implementation of these mechanisms or modifying the existing ones through a plugin system. As an extension, we complement the simulation with the implementation of Kubernetes artifacts by providing an adapter that translates events from the simulation to Kubernetes events and vice versa.

¹ University of Hamburg, Hamburg, Germany, {sebastian.frank, andre.van.hoorn}@uni-hamburg.de
² University of Würzburg, Würzburg, Germany, {martin.straesser, samuel.kounev}@uni-wuerzburg.de
³ DATEV eG, Nuremberg, Germany
**Results.** We demonstrate how MiSim simulates (1) its default resilience mechanisms and (2) fault injections — i.e., instance/service killing and latency injections. We also use TeaStore [Vo18], a reference microservice-based architecture, aiming to reproduce scaling behavior from an experiment using simulation. Our results show that MiSim allows for quantitative insights into microservice-based systems’ complex transient behavior by providing metrics, traces, and visualization scripts. Further, we show that a combination of simulation and real Kubernetes artifacts can correctly handle different configurations of orchestration mechanisms, boosting both the simulation’s use cases and authenticity.

**Conclusion.** The MiSim approach is a meaningful contribution to our vision of continuous specification and verification of resilience scenarios [Fr22a] in which it can be utilized to get quick feedback on scenario satisfaction at runtime and design-time as well as for what-if-scenarios. Further, MiSim’s extension allows authentic representation of container orchestration mechanisms that is useful for performance prediction of microservice applications with black-box container orchestration⁴.

**Data Availability.** We provide MiSim’s implementation⁵, the evaluation data⁶, and the implementation and evaluation results for our Kubernetes extension⁷.

**References**


---

⁴ https://gepris.dfg.de/gepris/projekt/510552229
⁵ https://github.com/Cambio-Project/MiSim
⁶ https://doi.org/10.24433/CO.9682966.v3
⁷ https://doi.org/10.24433/CO.4913288.v1