

# Security Chaos Engineering: Implementation based on established approaches of Chaos Engineering

- **Security Chaos Engineering (SCE)** ist ein neuartiger Lösungsansatz im Bereich der IT-Security, der auf dem Prinzip des Chaos Engineerings (CE) aufbaut.
- Der Ansatz des Chaos Engineering dient dazu die **Verfügbarkeit** der Infrastruktur, von Daten und Services sicherzustellen. Dabei werden in einer realen Umgebung unter Absicherung des Systems gezielt Fehler injiziert, um das System zum Kollabieren zu bringen. Im besten Fall hält das System jeder Injektion stand.
- Security Chaos Engineering zielt nun auf die **Erweiterung** des Ansatzes um die verbleibenden Security-Bestandteile, also **Integrität und Vertraulichkeit** ab.
- Da es sich hierbei allerdings um eine neue Technologie handelt, existieren keine **konkreten Konzepte** zur Umsetzung von Security Chaos Engineering in Unternehmen. Nachdem allerdings SCE eine Erweiterung des bereits etablierten CE darstellt, können Parallelen erkannt und genutzt werden



<https://cdn.lynda.com/course/5028636/5028636-637491087391090153-16x9.jpg>

## Forschungsfrage

Worin liegen die Unterschiede zwischen CE und SCE?

Welche Konzepte bezüglich der Umsetzung und Implementierung lassen sich auf Basis dieses Vergleichs (weiter-)entwickeln?

## Ansprechpartner



Michael Bitzer

## Vorgehen / Literatur

- Literaturüberblick über bestehende Publikationen zum Thema Security Chaos Engineering, Chaos Engineering und Konzepten bezüglich dessen Umsetzung und Implementierung innerhalb von Unternehmen. Ein zusätzlicher Fokus sollte auf die Unterschiede, sowohl technisch als auch organisatorisch, zwischen den beiden Ansätzen gelegt werden
- Literaturansätze: Jernberg, Runeson, Engström (2020) Getting Started with Chaos Engineering - design of an implementation framework in practice; Torkura et al. (2020) CloudStrike: Chaos Engineering for Security and Resiliency in Cloud Infrastructure; Rinehart, Shortridge (2020) Security Chaos Engineering - Gaining Confidence in Resilience and Safety at Speed and Scale