

Vom Ping zum Panorama: Monitoring mit System

Monitoring ist eines dieser Themen, bei denen fast jedes Team etwas einsetzt, und trotzdem regelmäßig überrascht wird. Server sind “plötzlich” weg, Kunden bemerken Ausfälle vor der IT, Rufbereitschaften brennen aus und am Montagmorgen weiß niemand so genau, was am Wochenende eigentlich passiert ist.

Dieser Vortrag zeigt, warum das kein Tool-Problem ist, und wie ein **strukturierter Monitoring-Ansatz** hilft, vom reaktiven Feuerlöschen zum beherrschbaren Betrieb zu kommen.

Statt einzelne Produkte zu vergleichen, geht es um die **grundlegenden Fragen**, die jedes Monitoring beantworten können muss:

- Was existiert in meiner Umgebung eigentlich gerade?
- Was davon ist *jetzt gerade* kaputt?
- Was war gestern, letzte Nacht oder am Wochenende los?
- Wie schlimm ist/war ein Problem?
- Wen betrifft es, nur uns intern oder auch Kunden?
- Und wer wird wann und wie informiert?

Anhand realer Beispiele aus einem produktiven Rechenzentrums- und Applikationsbetrieb wird gezeigt, wie man diese Fragen systematisch beantwortet, unabhängig davon, ob man Nagios, Zabbix, Prometheus, OpenNMS oder eine Kombination davon einsetzt.

Ein Schwerpunkt liegt auf **Severity-Modellen**, die mehr sind als nur „OK / Warning / Critical“. Der Vortrag zeigt, wie wohldefinierte Schweregrade:

- technische Signale von echtem Kunden-Impact trennen,
- Alert-Flut vermeiden,
- und Alerting-Regeln drastisch vereinfachen.

Darauf aufbauend wird erklärt, warum **Alerting per E-Mail** in der Praxis fast immer scheitert, und weshalb Ticketsysteme, Deduplizierung und bewusst *nicht* automatisch geschlossene Tickets entscheidend für nachhaltige Teamarbeit sind. Abschließend wird auf das Thema **Rufbereitschaft und Alert Fatigue** eingegangen, indem aufbauend auf den vorgestellten Konzepten vorgestellt wird, wie eine Rufbereitschaft ohne Burnout realisiert wird.

Fazit:

Gutes Monitoring ist Selbstfürsorge und entsteht nicht durch mehr Checks, sondern durch bessere Fragen, klare Definitionen und saubere Prozesse.

Der Vortrag liefert dafür ein erprobtes, praxisnahes Denkmodell. Er richtet sich an Admins, SREs, DevOps-Entwickler und Open-Source-Enthusiasten, die weniger Zeit mit Alarmsirenen und mehr Zeit mit stabilen Systemen verbringen wollen (und nachts in Ruhe schlafen).